



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

DEPARTMENT OF COMPUTER SYSTEMS

**AUTOMATICKÉ ŘÍZENÍ A MONITOROVÁNÍ
ÚSTŘEDNÍHO VYTÁPĚNÍ PRO STARŠÍ BUDOVY**

AUTOMATED CONTROL AND MONITORING OF CENTRAL HEATING FOR OLDER BUILDINGS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MATEJ MUŽILA

VEDOUČÍ PRÁCE

SUPERVISOR

Ing. VÁCLAV ŠIMEK

BRNO 2019

Zadání diplomové práce



22218

Student: **Mužila Matej, Bc.**

Program: Informační technologie Obor: Počítačové a vestavěné systémy

Název: **Automatické řízení a monitorování ústředního vytápění pro starší budovy**

Automated Control and Monitoring of Central Heating for Older Buildings

Kategorie: Vestavěné systémy

Zadání:

1. Seznamte se s principem řízení plynového kotle po stránce ohřevu vody a možností regulace tohoto procesu. Zaměřte se na typ Baxi Prime 24 a zejména pak na komunikační rozhraní s jeho řídicí jednotkou.
2. Prostudujte problematiku bezdrátových a drátových senzorů pro měření teploty. U bezdrátových senzorů se zaměřte na optimalizaci spotřeby energie vzhledem k životnosti baterií.
3. Vytvořte návrh architektury systému, který bude zajišťovat řízení vytápění. Systém bude obsahovat moduly pro ovládání topných těles, senzory teploty, modul pro ovládání plynového kotle a řídicí prvek postavený na některém jednodeskovém počítači (např. Raspberry Pi). K systému bude možný vzdálený přístup.
4. Proveďte fyzickou realizaci zvolených modulů na obvodové úrovni a implementujte obslužný firmware.
5. Demonstrujte funkčnost navrženého systému a zhodnoťte dosažené výsledky. Pokuste se navrhnout případná rozšíření či vylepšení.

Literatura:

- Dle pokynů vedoucího.

Při obhajobě semestrální části projektu je požadováno:

- Splnění bodů 1 až 3 zadání.

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Šimek Václav, Ing.**

Vedoucí ústavu: Sekanina Lukáš, prof. Ing., Ph.D.

Datum zadání: 1. listopadu 2018

Datum odevzdání: 22. května 2019

Datum schválení: 31. října 2018

Abstrakt

Cielom tejto diplomovej práce je vytvorenie systému automatizovanej regulácie vykurovania v starej budove. Súčasťou práce je návrh architektúry tohto systému, jeho implementácia, vrátane výroby prototypov potrebných zariadení a testovania celého systému v praxi, s dôrazom na funkčnosť a hospodárnosť. V rámci tejto práce bolo navrhnutých a vyrobených niekoľko hardvérových zariadení a obslužný softvér pre ne. Navrhnutá a implementovaná softvérová architektúra umožňuje pohodlné ovládanie systému pomocou grafického užívateľského rozhrania a umožňuje jeho jednoduché rozšírenie. Grafické užívateľské rozhranie, dostupné lokálne na dotykovom displeji aj vzdialene cez webovú aplikáciu, ponúka možnosti konfigurácie pripojených zariadení a vykurovacích zón, nastavovanie teplôt v zónach podľa dňa v týždni a dennej doby a prehľad štatistických informácií o vykurovacom systéme. Navrhnutý a implementovaný systém regulácie vykurovania bol úspešne otestovaný v reálnych podmienkach trojizbového bytu v historickej budove. Systém je možné ďalej zdokonaľovať a rozširovať po hardvérovej aj softvérovej stránke.

Abstract

The purpose of this thesis is a design of an automated heating regulation system for an old building. The thesis consists of a system architecture design, an implementation of the system, including prototype manufacturing of the required devices, and testing of the whole system in practice, with focus on functionality and efficiency. Several hardware devices were designed and manufactured, for which appropriate controlling software was implemented. The implemented software architecture enables users to control the system using a graphical user interface, and allows for an easy extension of the system. The graphical user interface, which is accessible both locally through a touchscreen and remotely using a web application, enables users to easily configure connected devices and regulated zones, and to set temperature level in the defined zones variably depending on a week day and time. The user interface also provides statistics about the heating system. The implemented automated heating regulation system was successfully tested in real conditions of a flat in a historical building. The system can be extended both hardware-wise and software-wise.

Klíčové slová

Raspberry Pi, OpenTherm, kotol, regulácia vykurovania, teplotný senzor, bezdrôtové, spotreba energie, dotykový displej

Keywords

Raspberry Pi, OpenTherm, boiler, heating regulation, thermometer, wireless, energy consumption, touchscreen

Citácia

MUŽILA, Matej. *Automatické řízení a monitorování ústředního vytápění pro starší budovy*. Brno, 2019. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Václav Šimek

Automatické řízení a monitorování ústředního vytápění pro starší budovy

Prehlásenie

Prehlasujem, že som túto diplomovú prácu vypracoval samostatne pod vedením Ing. Václava Šimka. Uviedol som všetky pramene a literárne publikácie, z ktorých som čerpal.

.....
Matej Mužila
21. mája 2019

Podakovanie

Ďakujem Ing. Václavovi Šimkovi za cenné pripomienky a súčinnosť pri implementácií zariadení popísaných v tejto práci. Ďalej by som rád poďakoval Jiřímu Jabůrkovi za ochotu a pomoc pri výrobe prototypov a Jaroslavovi Škarvadovi za rady a pripomienky ohľadom bezdrôtových technológií. Ďakujem tiež Janovi (Honzovi) Horákovi a Ondřejovi Vašíkovi za umožnenie skĺbiť prácu so štúdiom a ich priateľstvo. V neposlednom rade by som chcel poďakovať rodičom, starým rodičom a priateľke Lenke za podporu počas môjho štúdia.

Obsah

1	Úvod	11
2	Ovládanie plynového kotla	12
2.1	Rozhranie OpenTherm™	12
2.1.1	Fyzická vrstva	12
2.1.2	Dátová vrstva	14
2.1.3	Aplikačná vrstva	15
3	Stručný prehľad architektúry systému	16
3.1	Hardvérová architektúra systému	16
3.1.1	Káblové prepojenie prvkov	16
3.1.2	Bezdrôtové prepojenie prvkov	18
3.2	Softvérová architektúra	18
3.2.1	Riadiaci počítač	18
3.2.2	Centrálny server	19
4	Rozširujúci modul k <i>Raspberry Pi 2</i>	21
4.1	Výber komunikačných rozhraní	21
4.2	Rozhranie OpenTherm™	22
4.2.1	Hardvér	22
4.2.2	Firmvér	25
4.2.3	Softvér	26
4.3	Rozhranie CAN	26
4.3.1	Hardvér	26
4.3.2	Softvér	27
4.4	Bezdrôtové rozhranie	28
4.4.1	Hardvér	28
4.4.2	Softvér	29
4.5	Celková konštrukcia rozširujúceho modulu	29
5	Servomechanizmus ovládania ventilu radiátora	30
5.1	Mechanická konštrukcia servomechanizmu	30
5.1.1	Pohon	30
5.1.2	Materiál	30
5.1.3	Detekcia koncovej polohy ventilu	31
5.1.4	Rozmiestnenie komponentov	32
5.2	Hardvér	32
5.2.1	Riadenie motora	33

5.2.2	Pripojenie k zvyšku systému	33
5.2.3	Napájanie	33
5.2.4	Riešenie vzniknutých problémov	35
5.3	Firmvér	36
5.3.1	Komunikácia po zbernici CAN	36
5.3.2	Prevenca problémov	37
5.4	Montáž	37
6	Bezdrôtový teplotný senzor	39
6.1	Hardvér	39
6.1.1	Komponenty	39
6.1.2	Napájanie	39
6.1.3	Prototyp	41
6.2	Firmvér	41
6.2.1	Komunikácia	41
6.2.2	Režim spánku	41
6.2.3	Synchronizácia	41
6.2.4	Šifrovanie	42
6.3	Spotreba energie	42
7	Riadiaci počítač	47
7.1	Hardvér	47
7.2	Softvér	47
7.2.1	Systém riadenia bázy dát	48
7.2.2	Kontroléry jednotlivých rozhraní	49
7.2.3	Centrálny kontrolér	49
7.2.4	Webový server pre REST API	50
7.2.5	Grafické užívateľské rozhranie	50
7.2.6	Hlavný webový server	53
7.2.7	Prepojenie s centrálnym serverom	53
7.2.8	Spúšťanie programov	53
8	Centrálny server	54
8.1	Softvér	54
8.1.1	Operačný systém	54
8.1.2	Vzdialený prístup	54
9	Záver	56
	Literatúra	58
A	Zoznam implementovaných správ protokolu OpenTherm™	61
B	Kompletá konštrukcia rozširujúceho modulu k <i>Raspberry Pi 2</i>	63
B.1	Schéma	63
B.2	Plošný spoj	68
B.3	Zoznam komponentov	74
B.4	Obrázky	77

C	Kompletá konštrukcia servomechanizmu ovládania ventilu radiátora	78
C.1	Elektrická konštrukcia	78
C.1.1	Schéma	78
C.1.2	Plošný spoj	80
C.1.3	Zoznam komponentov	86
C.2	Mechanická konštrukcia	87
C.2.1	Zoznam spojovacieho materiálu	87
C.2.2	Technické výkresy	87
C.3	Obrázky	93
D	Kompletá konštrukcia bezdrôtového teplotného senzora	95
D.1	Schéma	95
D.2	Plošný spoj	97
D.3	Zoznam komponentov	103
D.4	Obrázky	104
E	Obsah priloženého dátového nosiča	105

Zoznam obrázkov

2.1	Časový priebeh komunikácie protokolu OpenTherm™	14
2.2	Dátový rámec protokolu OpenTherm™	14
2.3	Časovanie konverzácie protokolu OpenTherm™	14
3.1	Hardvérová architektúra systému	17
3.2	Softvérová architektúra riadiaceho počítača	19
3.3	Architektúra vzdialeného prístupu	20
4.1	Schéma zapojenia OpenTherm™ rozhrania	23
4.2	Vzostupná hrana signálu v doméne napätia na rozhraní OpenTherm™	24
4.3	Zostupná hrana signálu v doméne napätia na rozhraní OpenTherm™	24
4.4	Dátový rámec prechádzajúci po zbernici I ² C™	25
4.5	Napätie na rozhraní OpenTherm™ v priebehu posielania jedného bitu	26
4.6	Schéma zapojenia rozširujúceho obvodu pre rozhranie CAN	27
4.7	Schéma zapojenia bezdrôtového rozhrania	28
5.1	3D model servomechanizmu ventilu	31
5.2	Fotografia servomechanizmu ventilu	33
5.3	Schéma zapojenia servomechanizmu ventilu	34
5.4	Časový priebeh signálu na zbernici CAN	35
5.5	Dátový rámec prechádzajúci po zbernici CAN	37
6.1	Schéma zapojenia bezdrôtového teplotného senzora	40
6.2	Spotreba energie bezdrôtového teplotného senzora	43
6.3	Detail spotreby energie bezdrôtového teplotného senzora pri meraní teploty	43
6.4	Detail spotreby energie bezdrôtového teplotného senzora pri komunikácii	44
6.5	Využitelná energia batérie bezdrôtového teplotného senzora	45
7.1	Schéma navrhutej a implementovanej databázy	48
7.2	Užívateľské rozhranie – sekcia nastavovania teplomerov	50
7.3	Užívateľské rozhranie – sekcia nastavovania vykurovacích zón	51
7.4	Užívateľské rozhranie – sekcia nastavovania teploty vykurovacích zón	52
7.5	Užívateľské rozhranie – prehľad nameraných teplôt podľa zóny	52
B.1	Fotografia rozširujúceho modulu k <i>Raspberry Pi 2</i>	77
C.1	Fotografia plošného spoja servomechanizmu ventilu	93
C.2	Fotografia servomechanizmu ventilu bez nasadeného krytu elektroniky	93
C.3	Rez zjednodušeným modelom servomechanizmu ovládania ventilu a ventilu samotného	94

D.1	Fotografia plošného spoja bezdrôtového teplomera	104
-----	--	-----

Zoznam tabuliek

2.1	Charakteristiky prenosového média rozhrania OpenTherm™	13
2.2	Charakteristiky prenosu po rozhraní OpenTherm™ v doméne prúdu	13
2.3	Charakteristiky prenosu po rozhraní OpenTherm™ v doméne napätia	13
A.1	Zoznam implementovaných správ protokolu OpenTherm™	61
B.1	Zoznam komponentov pre rozširujúci modul k <i>Raspberry Pi 2</i>	74
C.1	Zoznam elektrických komponentov pre servomechanizmus ventilu	86
C.2	Zoznam spojovacieho materiálu pre servomechanizmus ventilu	87
D.1	Zoznam komponentov pre bezdrôtový teplotný senzor	103

Kapitola 1

Úvod

Mnohé dnešné budovy disponujú inteligentnými technológiami, ktoré uľahčujú život ich obyvateľom a užívateľom a v značnej miere tiež prispievajú k hospodárnosti ich prevádzky. Jedná sa o automatické systémy osvetlenia, vykurovania, ventilácie, riadenia prístupu a mnohé ďalšie. Týmito systémami disponujú najmä stredne veľké a veľké budovy, nakoľko prvotná investícia pri ich zavádzaní nie je zanedbateľná. Komplexnejším problémom je nasadzovanie týchto technológií v starších, už stojacich budovách, pretože sú mnohokrát potrebné relatívne náročné a drahé stavebné úpravy.

Autor práce sa v praxi zaoberal problematikou regulácie vykurovania v trojizbovom byte v historickej budove. Napriek tomu, že doteraz využíval neporovnateľne jednoduchší systém, vykázal dlhodobu značné úspory nákladov na vykurovanie. Systém navrhnutý v tejto práci by mal ďalej optimalizovať úspory a užívateľský komfort, najmä umožniť udržiavanie prednastavených teplôt v zónach v požadovaných časoch a poskytnúť jednoduché ovládanie zo strany užívateľov.

Cieľom práce bolo zoznámenie sa s možnosťami regulácie plynového kotla, problematikou bezdrôtových a drôtových senzorov teploty, konštrukcia servomechanizmov ovládania ventilov radiátorov a návrh architektúry systému riadenia vykurovania bytu, prípadne rodinného domu. Systém mal byť jednoducho nasaditeľný v už stojacich budovách bez nutnosti drahých stavebných úprav. Cena tohto systému mala byť dostatočne nízka, aby bolo jeho nasadenie mysliteľné aj v malých budovách. Tento systém je prepojený s centrálnym serverom, ktorého prostredníctvom je možné vzdialene spravovať aj niekoľko jeho inštancií. Na lokálne ovládanie slúži modul s dotykovým displejom.

Kapitola 2

Ovládanie plynového kotla

Kotly, ktoré sa bežne vyskytujú na trhu, sú zvyčajne vybavené možnosťou regulácie ohrevu vykurovacieho média ústredného kúrenia pomocou vzdialeného termostatu. V závislosti od typu kotla môže tento termostat regulovať kotel binárne (teda vypnutý alebo zapnutý) alebo na vyššej úrovni, kedy môže regulovať celú množinu parametrov.

Historicky bývali kotly vybavené len možnosťou binárnej regulácie. Pri tomto type regulácie termostat v prípade potreby vykurovania spojí regulačný obvod a kotel začne zohrievať vykurovacie médium.

Omnoho modernejším a dokonalejším prístupom je digitálne ovládanie kotla, napríklad pomocou protokolu OpenTherm™. Pomocou tohto protokolu je možné zisťovať a nastavovať rôzne parametre súvisiace s prevádzkou kotla, ako napríklad: teplotu vykurovacieho média na vstupe a na výstupe, moduláciu plameňa a podobne [27].

V dnešnej dobe sú na trhu k dispozícii aj takzvané inteligentné kotly, ktoré sú pripojené na internet. Táto práca o nich však nepojednáva, nakoľko stále nie sú príliš rozšírené, a to najmä v starších budovách, kde sa predpokladá nasadenie tohto systému.

2.1 Rozhranie OpenTherm™

Rozhranie OpenTherm™ umožňuje point-to-point komunikáciu medzi kotlom a termostatom (kontrolérom). Protokol OpenTherm™ predpokladá, že termostat je master a kotel je slave. Termostat môže nastavovať a čítať parametre kotla, a to vrátane chybových hlásení [27].

OpenTherm™ protokol je rozdelený do vrstiev založených na OSI modeli. Pozostáva z troch vrstiev: fyzickej, dátovej a aplikačnej.

2.1.1 Fyzická vrstva

Fyzická vrstva definuje elektrické a mechanické charakteristiky prenosového média (viď tabuľku 2.1) a mechanizmus prenosu jednotlivých bitov, vrátane enkódovania na bitovej úrovni. Tiež zabezpečuje kontrolu chýb na bitovej úrovni.

Tabuľka 2.1: Charakteristiky prenosového média rozhrania OpenTherm™

Počet vodičov	2
Typ vedenia	nekrútený pár
Maximálna dĺžka vedenia	50 m
Maximálny elektrický odpor vedenia	10 Ω
Polarita zapojenia	nepodstatná
Maximálne napätie v zbernici	42 V

Komunikácia funguje na základe posielania signálov v doméne prúdu v smere kotol \rightarrow termostat a posielania signálov v doméne napätia v smere termostat \rightarrow kotol. Pri komunikácii v doméne prúdu sú charakteristiky definované v tabuľke 2.2 a pri komunikácii v doméne napätia v tabuľke 2.3.

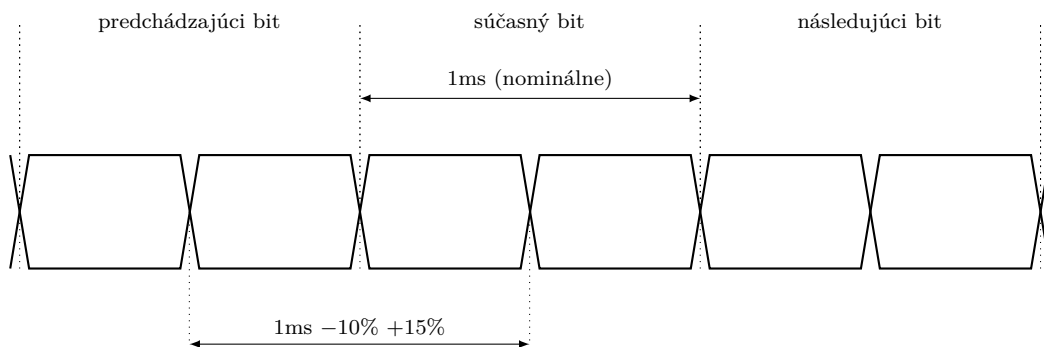
Tabuľka 2.2: Charakteristiky prenosu po rozhraní OpenTherm™ v doméne prúdu

High level (aktívny stav)	17 mA – 23 mA
Low level (pokojový stav)	5 mA – 9 mA
Maximálny čas prechodu Low – High	50 μ s
Maximálny čas prechodu High – Low	50 μ s

Tabuľka 2.3: Charakteristiky prenosu po rozhraní OpenTherm™ v doméne napätia

High level (pokojový stav)	15 V – 18 V
Low level (aktívny stav)	≤ 7 V
Maximálny čas prechodu Low – High	50 μ s
Maximálny čas prechodu High – Low	50 μ s

Pre enkódovanie jednotlivých bitov je použité Manchesterké kódovanie. Toto enkódovanie zabezpečuje synchronizáciu prenosu na bitovej úrovni, keďže na každý prenesený bit pripadá minimálne jeden prechod, a to presne v strede prenosového intervalu. Bit 1 je definovaný ako prechod aktívny stav \rightarrow pokojový stav a bit 0 je definovaný ako prechod pokojový stav \rightarrow aktívny stav. Rýchlosť komunikácie je 1000 bitov/s a čas medzi stredmi prenosových intervalov je 900 μ s – 1150 μ s. Časovanie by malo byť synchronizované pri prenose každého bitu. Časový priebeh komunikácie je znázornený na obrázku 2.1.

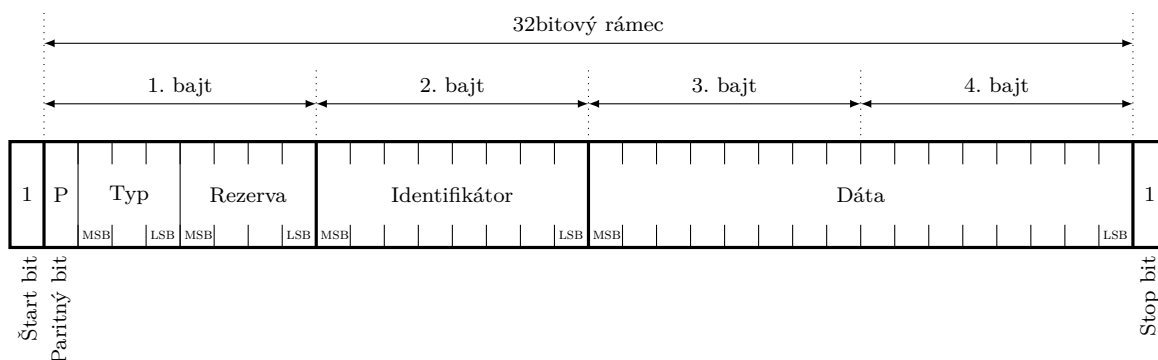


Obr. 2.1: Časový priebeh komunikácie protokolu OpenTherm™

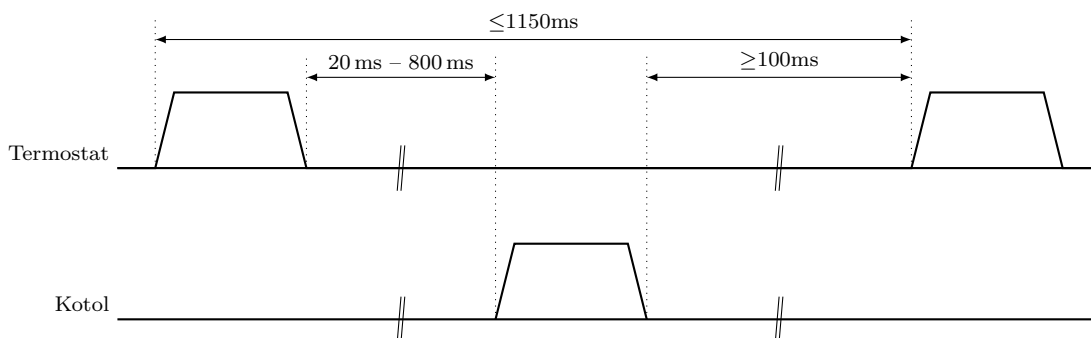
2.1.2 Dátová vrstva

Dátová vrstva je zodpovedná za vytvorenie rámca, zapúzdrujúceho aplikačnú vrstvu protokolu. Definuje typ správ a zabezpečuje ochranu proti chybám prenosu.

Dátový rámec, zobrazený na obrázku 2.2, obsahuje štart bit, paritný bit, typ správy, identifikátor dát, dáta a stop bit.



Obr. 2.2: Dátový rámec protokolu OpenTherm™



Obr. 2.3: Časovanie konverzácie protokolu OpenTherm™

Typ správy definuje, či sa jedná o správu v smere termostat \rightarrow kotol alebo naopak. V prípade komunikácie v smere termostat \rightarrow kotol špecifikuje, či sa jedná o zápis alebo o čítanie, a v smere kotol \rightarrow termostat špecifikuje, či boli zápis alebo čítanie úspešné.

Identifikátor dát špecifikuje dátovú položku, ktorej obsah sa zapisuje, prípadne číta. Dáta obsahujú dátové položky korešpondujúce s identifikátorom dát. Paritný bit je volený tak, aby bol počet bitov 1 v celom 32bitovom rámci párný.

Komunikácia pomocou protokolu OpenTherm™ pozostáva zo série konverzácií medzi zariadeniami. Každá konverzácia je iniciovaná master zariadením, teda termostatom. OpenTherm™ protokol vyžaduje, aby termostat inicioval komunikáciu aspoň raz sa 1150 ms (obrázok 2.3). Konverzácia je limitovaná na jednu výmenu rámcov, teda jeden rámec odoslaný a jeden rámec prijatý.

2.1.3 Aplikačná vrstva

Aplikačná vrstva je zodpovedná za prenos aplikačných správ medzi kotlom a termostatom. Sú definované triedy správ, správy a formát prenášaných hodnôt [27]. Pre návrh adaptéra táto vrstva nie je podstatná a teda ju nebudeme ďalej opisovať.

Kapitola 3

Stručný prehľad architektúry systému

Medzi požiadavky na systém, ktorý je predmetom práce, patrí, aby bol jednoducho nasaditeľný aj v existujúcich budovách, ovládateľný vzdialene a aby bol vhodný pre nasadenie v bytoch, prípadne rodinných domoch. To kladie relatívne veľké požiadavky na nízku cenu systému samotného, ako aj na cenu jeho nasadenia, a tiež to obmedzuje možnosť stavebných úprav potrebných pre jeho nasadenie. Preto bola zvolená architektúra, ktorá je relatívne lacná a ktorá minimalizuje nutnosť stavebných úprav.

3.1 Hardvérová architektúra systému

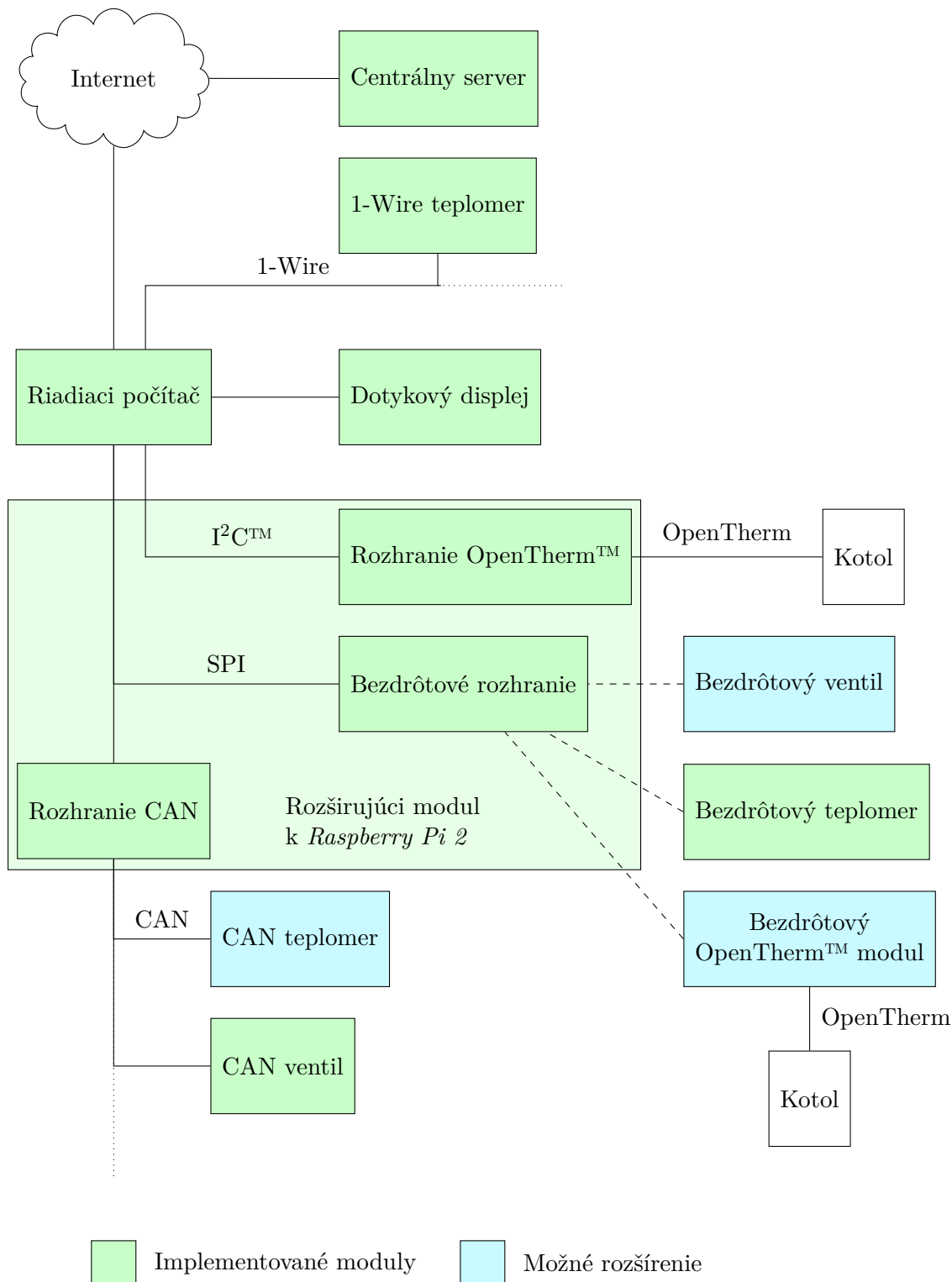
V rámci tejto práce bola navrhnutá hardvérová architektúra systému znázornená na obrázku 3.1. Zvolená architektúra umožňuje relatívne jednoduché rozširovanie systému v budúcnosti.

Požiadavok na možnosť vzdialeného ovládania systému prináša so sebou nutnosť z internetu dostupného centrálného servera, pomocou ktorého bude možné riadiť a monitorovať jednotlivé inštancie tohto systému. Prostredníctvom centrálného servera je možné sa pripojiť na webové užívateľské rozhranie konkrétnej inštancie vykurovacieho systému. Toto pripojenie by sa síce dalo realizovať aj bez centrálného servera, no prinieslo by to celý rad obmedzení. Každá inštancia systému by musela disponovať verejnou, najlepšie statickou IP adresou. Tiež by bolo nutné v domácom routri nastavovať statický NAT.

Ako centrálny riadiaci prvok každej inštancie tohto systému bol zvolený jednodoskový počítač *Raspberry Pi 2*. Hlavným užívateľským rozhraním v domácnosti je dotykový displej pripojený k riadiacemu počítaču. Pomocou tohto displeja je možné systém monitorovať a riadiť ho. Zároveň je možné ovládať systém vzdialene pomocou webovej aplikácie.

3.1.1 Káblové prepojenie prvkov

Systém počíta s tým, že tam, kde je to možné, budú viesť k jednotlivým prvkom káble, respektíve budú tieto prvky spojené zbernicou. Toto riešenie síce nebude aplikovateľné všade, no v istých prípadoch to redukuje cenu systému, znižuje nároky na údržbu (výmena batérií bezdrôtových senzorov) a zvyšuje spoľahlivosť a bezpečnosť. Pretože sa počíta s tým, že káblové vedenia zbernice môžu byť pomerne dlhé a že na ňu bude pripojené množstvo zariadení, bola zvolená zbernica CAN. Táto zbernica vyžaduje len dva dátové vodiče, používa diferenciálny pár vodičov, a teda je relatívne odolná voči rušeniu. Zbernica CAN tiež zabezpečuje arbitráciu [29].



Obr. 3.1: Hardvérová architektúra systému

Pomocou zbernice CAN sú do systému pripojené servomechanizmy ovládania ventilov radiátorov. Výhľadovo sa počíta aj s konštrukciou iných zariadení, napríklad teplotných senzorov a kontrolérov zásuviek, pripojiteľných k tejto zbernici.

K riadiacemu počítaču je možné pripojiť teplotné senzory aj za pomoci zbernice 1-Wire. Táto zbernica síce teoreticky nie je príliš vhodná, najmä pri použití stromovej sieťovej topológie, prípadne hviezdicovej topológie [17], ale prax ukázala, že pre účely tohto systému je dostatočne spoľahlivá. Veľkým plusom tejto technológie je existencia teplotného senzoru *Dallas Semiconductor DS18B20*, ktorý je pomerne lacný a presný, a tiež jednoduchosť zapojenia.

Pretože zbernica CAN potrebuje 2 signálové vodiče a zbernica 1-Wire len jediný signálny vodič [18], je možné obe tieto zbernice viesť jediným UTP káblom spolu so zemniacim vodičom a napájaním pre pripojené zariadenia s menším odberom.

3.1.2 Bezdrôtové prepojenie prvkov

System musí byť nasaditeľný aj bez nutnosti stavebných úprav, a teda nie je možné všade viesť kábel. K počítaču je preto pripojený bezdrôtový modul umožňujúci komunikáciu s ďalšími zariadeniami bez nutnosti vedenia káblov. Zatiaľ boli implementované bezdrôtové senzory teploty, neskôr bude možné doplniť bezdrôtové servomechanizmy ventilov radiátorov, prípadne bezdrôtové ovládače kotla a iné.

Počítač *Raspberry Pi 2* samotný nie je vybavený žiadnym rozhraním vhodným pre komunikáciu s plynovým kotlom. Je teda potrebné rozšíriť počítač *Raspberry Pi 2* o modul disponujúci týmto rozhraním.

3.2 Softvérová architektúra

System pre svoju správnu funkciu potrebuje značné množstvo softvéru na rôznych úrovniach. Táto kapitola sa zaoberá popisom softvérovej architektúry.

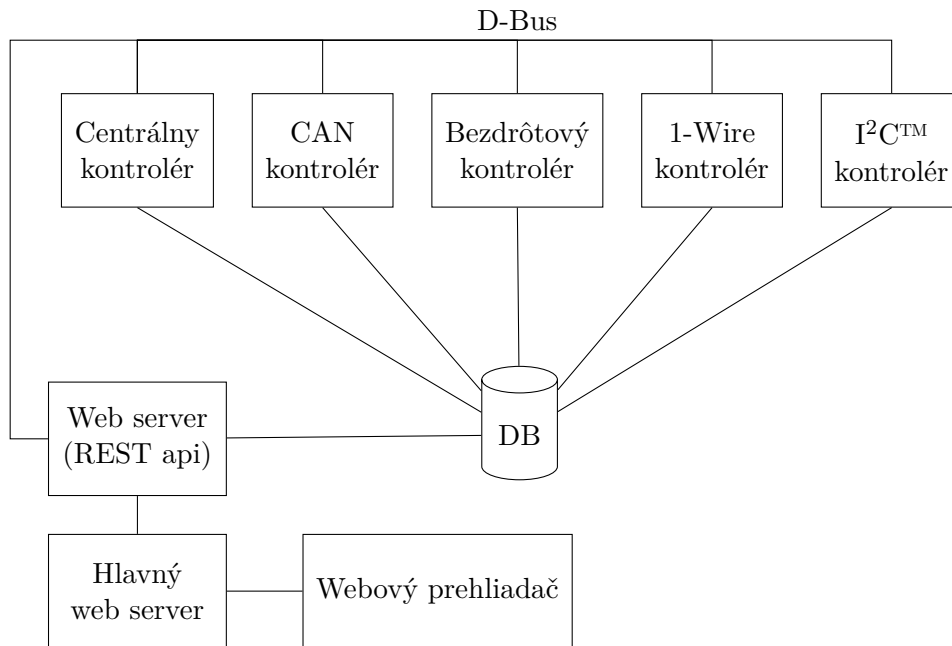
3.2.1 Riadiaci počítač

Softvér riadiaceho počítača (*Raspberry Pi 2*) musí zabezpečovať viacero funkcií. V prvom rade musí byť schopný komunikovať s jednotlivými senzormi a aktuátormi (servomechanizmy ventilov, kotol), vyhodnocovať získané údaje a rozhodovať o potrebných zásahoch. Musí poskytovať prívetivé užívateľské rozhranie pre svoje ovládanie a zabezpečiť možnosť vzdialeného ovládania. Tiež musí zaznamenávať namerané hodnoty pre účely zobrazenia štatistík a prípadného inteligentného ovládania. Preto bola navrhnutá architektúra zobrazená na obrázku 3.2.

Základom riadiaceho počítača je systém riadenia bázy dát. Tento databázový systém uchováva stav systému, a namerané hodnoty. Pripájajú sa k nemu ďalšie subsystemy, umožňujúce interakciu s užívateľom a samotné riadenie vykurovania.

Pre každé z implementovaných komunikačných rozhraní bol navrhnutý oddelený kontrolér, špecifický pre toto rozhranie. Vďaka tomu je možné vytvoriť abstrakčnú vrstvu medzi centrálnym kontrolérom a jednotlivými komunikačnými rozhraniami. Centrálny kontrolér teda môže pristupovať ku všetkým rozhraniam jednotným spôsobom. Do budúcnosti to prináša možnosť jednoducho rozšíriť systém o ďalšie komunikačné rozhrania – stačí pre dané rozhranie implementovať kontrolér.

Centrálny kontrolér má za úlohu riadenie vykurovania a zber dát. Aktuálny stav a topológiu systému získava z databázového systému a zo sensorov. Na základe týchto informácií rozhoduje o tom, akú je potrebné vykonať akciu.



Obr. 3.2: Softvérová architektúra riadiaceho počítača

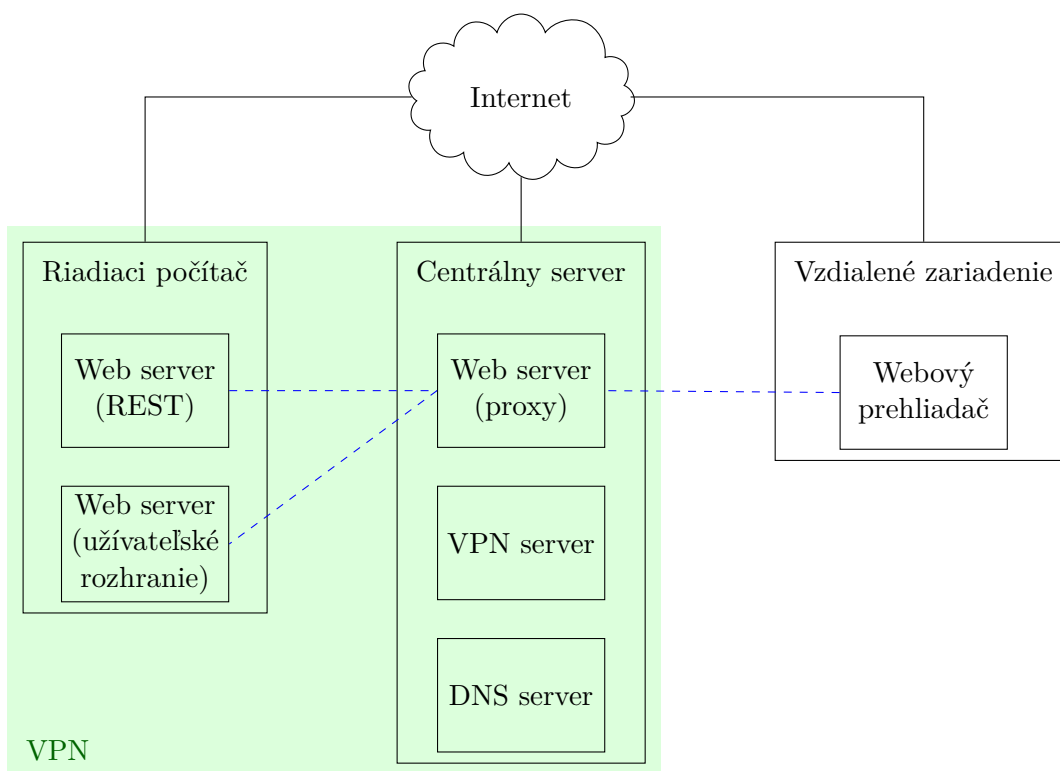
Pre komunikáciu medzi jednotlivými subsystémami (kontrolérmi) bola použitá softvérová zbernica D-Bus. D-Bus je mechanizmus, ktorý umožňuje medziprocesovú komunikáciu a vzdialené volanie procedúr medzi viacerými programami bežiacim na jednom počítači. Použitie tohto riešenia zjednodušuje implementáciu jednotlivých častí systému [14].

Pre účely nastavovania a zobrazovania údajov o systéme bol implementovaný webový server ponúkajúci REST API k danému systému. REST API je použité ako na vzdialené ovládanie systému, tak aj pre účely lokálneho užívateľského rozhrania na dotykovom displeji. Na riadiacom počítači beží aj druhý (hlavný) webový server, ktorý sprístupňuje webovú aplikáciu užívateľského rozhrania, dostupnú na dotykovom displeji.

3.2.2 Centrálny server

Centrálny server umožňuje riadenie a monitorovanie mnohých inštancií tohto systému. Ako jediný prvok musí disponovať verejnou IP adresou. Funguje ako tenký klient medzi inštanciou systému a vzdialeným zariadením pripájaným sa k nej. Aby mal server prístup k jednotlivým inštanciám systému, sú tieto inštancie zahrnuté do virtuálnej súkromnej siete (VPN). Riadiacim prvkom tejto siete je práve centrálny server. Architektúra je znázornená na obrázku 3.3.

Na serveri beží HTTP proxy server, ktorý sprístupní vzdialenému zariadeniu užívateľské rozhranie danej inštancie, umožňujúce plnú kontrolu. Server samozrejme musí spravovať adresy všetkých inštancií na virtuálnej sieti. Na tento účel bol použitý dynamický DNS server [12].



Obr. 3.3: Architektúra vzdialeného prístupu

Kapitola 4

Rozširujúci modul k *Raspberry Pi 2*

4.1 Výber komunikačných rozhraní

Jednodoskový počítač *Raspberry Pi 2* nedisponuje komunikačnými rozhraniami vhodnými pre potreby tohto systému. Neobsahuje žiadne nízkoenergetické bezdrôtové komunikačné rozhranie ani OpenTherm™ rozhranie [25]. Obsiahnuté komunikačné rozhrania (FastEthernet, USART, I²C™, SPI) nie sú príliš vhodné na káblovú komunikáciu medzi viacerými zariadeniami na veľkú vzdialenosť.

Rozhranie FastEthernet umožňuje (bez ďalších smerovačov a prepínačov) len point-to-point komunikáciu. Použitie tohto rozhrania na komunikáciu medzi jednotlivými zariadeniami by prinieslo nutnosť viesť z centrálného prepínača ku každému zariadeniu oddelený kábel, prípadne nutnosť inštalácie ďalších prepínačov. V neposlednom rade by použitie tohto rozhrania viedlo k nezanedbateľnému predraženiu jednotlivých modulov (pripojených zariadení).

Rozhranie USART, prípadne *RS232*, *RS485* alebo *RS422*, je síce možné použiť pre komunikáciu medzi viacerými zariadeniami, a to aj na pomerne veľkú vzdialenosť, no nerieši problémy spojené s adresovaním zariadení, prípadne kolíziami na zbernici. Toto rozhranie je tiež vhodné ponechať voľné pre účely ladenia.

Zbernica I²C™ nie je príliš vhodná pre komunikáciu na potrebnú vzdialenosť [23]. Je relatívne náchylná na rušenie. Tiež je možné na tejto zbernici adresovať len 127 zariadení. Zbernica je arbitrovaná master zariadením a bolo by teda nutné riešiť možnosť iniciácie komunikácie slave zariadením pomocou nejakého externého kanálu a riešiť kolízie na zbernici.

Zbernica SPI pri bežnom zapojení vyžaduje, aby ku každému zariadeniu viedol oddelený signálny vodič aktivujúci toto zariadenie [22]. Bolo by teda potrebných $4 + n$ vodičov. Za bežných okolností je táto zbernica tiež riadená jediným master zariadením. Je síce možné zapojiť zariadenia do kruhu, no toto riešenie by prinášalo komplikácie s inicializáciou komunikácie slave zariadením a tiež zvyšovalo počet možných miest zlyhania [15].

Pre implementáciu bola použitá zbernica CAN. Jedná sa o robustnú multi-master sériovú zbernicu používanú najmä v automobilovom priemysle. Zbernica CAN používa pre komunikáciu diferenciálny pár vodičov a obsahuje arbitračné mechanizmy. Komunikačný protokol používaný touto zbernicou zahŕňa mechanizmy pre kontrolu integrity prenášaných správ a pre prenos informácie o doručení [29]. Problémom je, že počítač *Raspberry Pi 2* neobsahuje toto rozhranie, a je teda nutné ho vhodným spôsobom doplniť.

Počítač *Raspberry Pi 2* tiež neobsahuje žiadne rozhranie pre komunikáciu s kotlom. Pri použití historického spôsobu ovládania kotla (zapnutý/vypnutý) by bolo nutné doplniť počítač *Raspberry Pi 2* o relé, prípadne podobné zariadenie, ktoré by galvanicky oddelilo regulačnú slučku kotla a eventuálne by ju spájalo. Nakoľko však dnes už aj kotly nízkej triedy obsahujú OpenTherm™ rozhranie, bolo rozhodnuté o jeho doplnení do *Raspberry Pi 2*.

Verzia 3 jednodoskového počítača *Raspberry Pi* síce obsahuje WiFi a Bluetooth rozhrania, no tieto neboli vyhodnotené ako vhodné pre použitie v tomto systéme, a to najmä kvôli ich vysokým energetickým nárokom a malému dosahu.

Na trhu sú rôzne rozširujúce moduly k *Raspberry Pi 2* obsahujúce spomínané chýbajúce rozhrania. Pravdepodobne však neexistuje komerčne dostupný modul, ktorý by počítač *Raspberry Pi 2* rozširoval o všetky potrebné rozhrania pri zachovaní kompaktných rozmerov. Použitie viacerých samostatných rozširujúcich modulov by prinášalo problémy s mechanickou konštrukciou a tiež by nepôsobilo elegantne. Z týchto dôvodov bolo rozhodnuté o stavbe rozširujúceho modulu, ktorý bude obsahovať všetky potrebné rozhrania. Jeho kompletná dokumentácia je uvedená v prílohe B.

4.2 Rozhranie OpenTherm™

4.2.1 Hardvér

Oddeľovací obvod

Ako je vysvetlené v kapitole 2.1, napätie na rozhraní OpenTherm™ môže dosahovať až 42 V a jeho polarita nie je definovaná. Toto napätie je do rozhrania dodávané slave zariadením, teda kotlom [27]. Komunikácia jedným smerom prebieha v doméne prúdu a v opačnom smere v doméne napätia na rozhraní. Je zrejmé, že nie je možné k tomuto rozhraniu priamo pripojiť žiaden bežný logický obvod a už vôbec nie komunikovať po tomto rozhraní. Preto bol skonštruovaný oddeľovací obvod znázornený na obrázku 4.1, ktorý umožňuje pripojenie mikrokontroléra k OpenTherm™ rozhraniu.

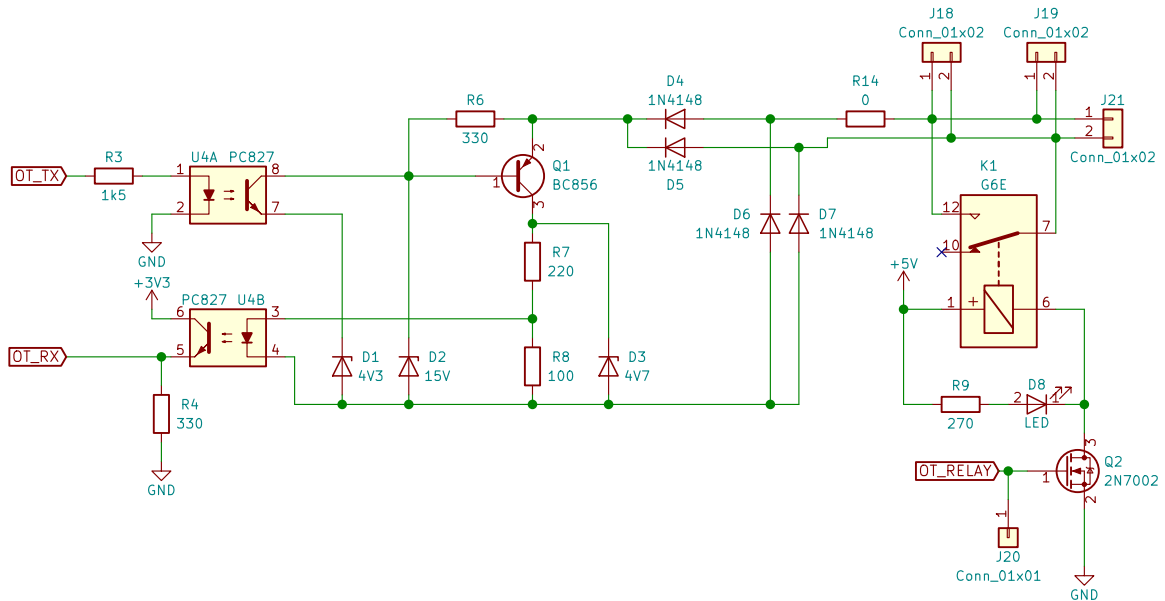
Schéma vychádza z typického zapojenia [19] oddeľovacieho obvodu pre rozhranie OpenTherm™, bola však rozšírená o relé K1, doplnené unipolárnym tranzistorom Q2, ktoré je použiteľné pre ovládanie kotla historickým spôsobom. Pomocou tohto relé je možné ovládať aj kotly, ktoré nie je možné ovládať pomocou OpenTherm™ rozhrania ani v režime kompatibility. Toto relé prípadne umožňuje pri výrobe vôbec neosadiť obvody spojené s komunikáciou pomocou OpenTherm™ protokolu a modul bude stále schopný ovládať kotol historickým spôsobom.

Terminály OT_TX a OT_RX sú logickým vstupom a výstupom, kotol je pripojený ku konektoru J21. Terminál OT_RELAY je riadiacim vstupom pre relé K1. Konektory J18, J19 a J20 sú určené pre účely ladenia. Prepojka R14 umožňuje v prípade potreby vyradiť z činnosti OpenTherm™ rozhranie a ponechať v činnosti len relé K1.

Obvod galvanicky oddeľuje OpenTherm™ rozhranie od zvyšku zariadenia. Pre tento účel bol použitý optočlen U4 a relé K1. Ich izolačná pevnosť značne prevyšuje kladené nároky [27, 24, 7].

Pripojenie k *Raspberry Pi 2*

Pretože OpenTherm™ štandard vyžaduje pomerne striktné časovanie a na *Raspberry Pi 2* nebeží real-time operačný systém, takéto striktné časovanie by mohlo spôsobovať problémy. Preto bol zvolený konzervatívny prístup: digitálny logický vstup a výstup (OT_RX a OT_TX)



Obr. 4.1: Schéma zapojenia OpenTherm™ rozhrania

OpenTherm™ rozhrania nie sú pripojené priamo k *Raspberry Pi 2*, ale k mikrokontroléru *Atmel ATmega328*, ktorý zabezpečí potrebné časovanie. Tento mikrokontrolér je následne pripojený pomocou zbernice I²C™ k *Raspberry Pi 2* (viď prílohu B.1).

Zbernica I²C™ bola zvolená preto, lebo *Raspberry Pi 2* natívne podporuje len dve zariadenia na zbernici SPI, ktorá má len dva dedikované slave select signály [25]. Ostatné slave select signály by bolo potrebné generovať softvérovo. Ako je uvedené nižšie, k zbernici SPI sú pomocou natívnych slave select signálov pripojené ďalšie dve zariadenia. *Raspberry Pi 2* tiež obsahuje len jedinú prístupnú sériovú linku. Tá sa môže zísť pri ladení zariadenia. Najjednoduchším riešením sa teda javilo použitie zbernice I²C™.

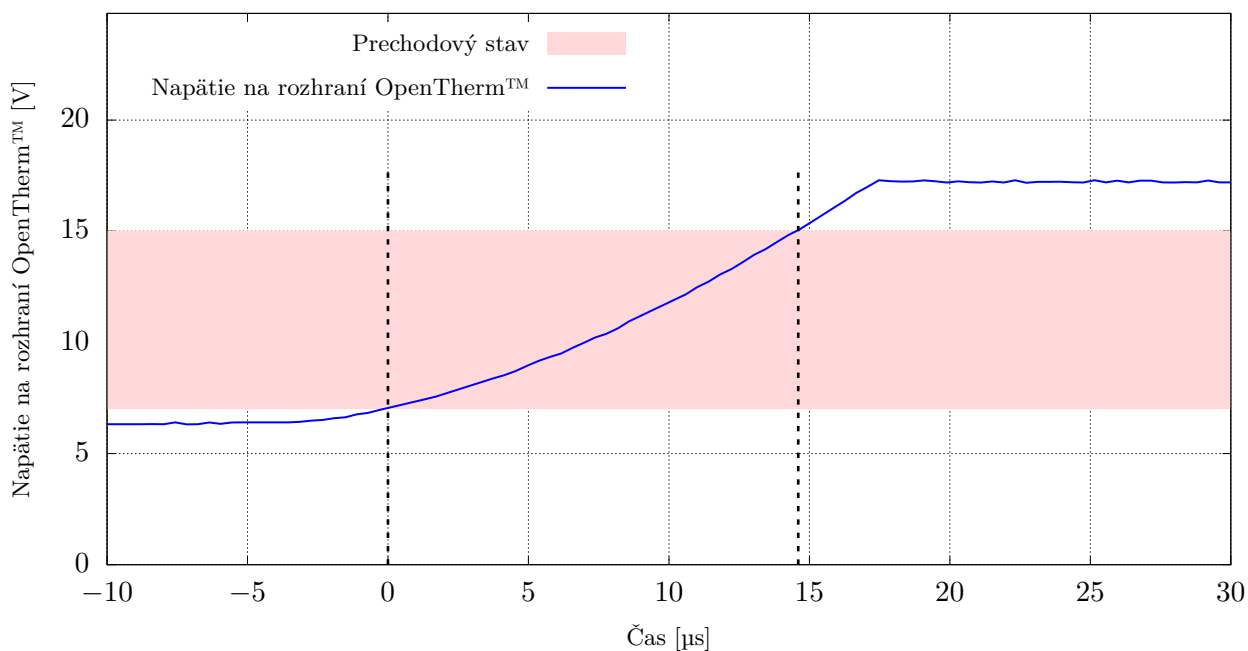
Obvod *Atmel ATmega328* je pre tieto účely pomerne predimenzovaný, je ho však bez ďalších hardvérových zásahov možné nahradiť obvodom *Atmel ATmega8* [1, 2]. Prípadne by sa dal použiť aj nejaký menší, jednoduchší a lacnejší obvod, tam by však už boli nutné hardvérové zmeny. K výberu tohto čipu viedla aj skutočnosť, že je súčasťou vývojovej dosky *Arduino Nano* a teda bolo pomerne jednoduché zhotoviť funkčný prototyp OpenTherm™ prevodníka ešte pred návrhom kompletného rozširujúceho modulu k *Raspberry Pi 2*. Rozdiel ceny v rádoch desiatok korún oproti najlacnejšej novej alternatíve je v prípade malosériovej výroby zanedbateľný. Výhodou je, že značné množstvo pinov tohto mikrokontroléra zostalo nevyužitých a je ich teda možné použiť pri prípadnom rozširovaní zariadenia. Nevyužitie piny boli vyvedené na konektory.

Aby mohol mikrokontrolér priamo komunikovať s *Raspberry Pi 2*, jeho napájacie napätie bolo redukované na výrobcom podporovaných 3.3 V. Toto umožňuje jeho priame pripojenie k TTL netolerantnému *Raspberry Pi 2* bez použitia prevodníka napätových úrovní.

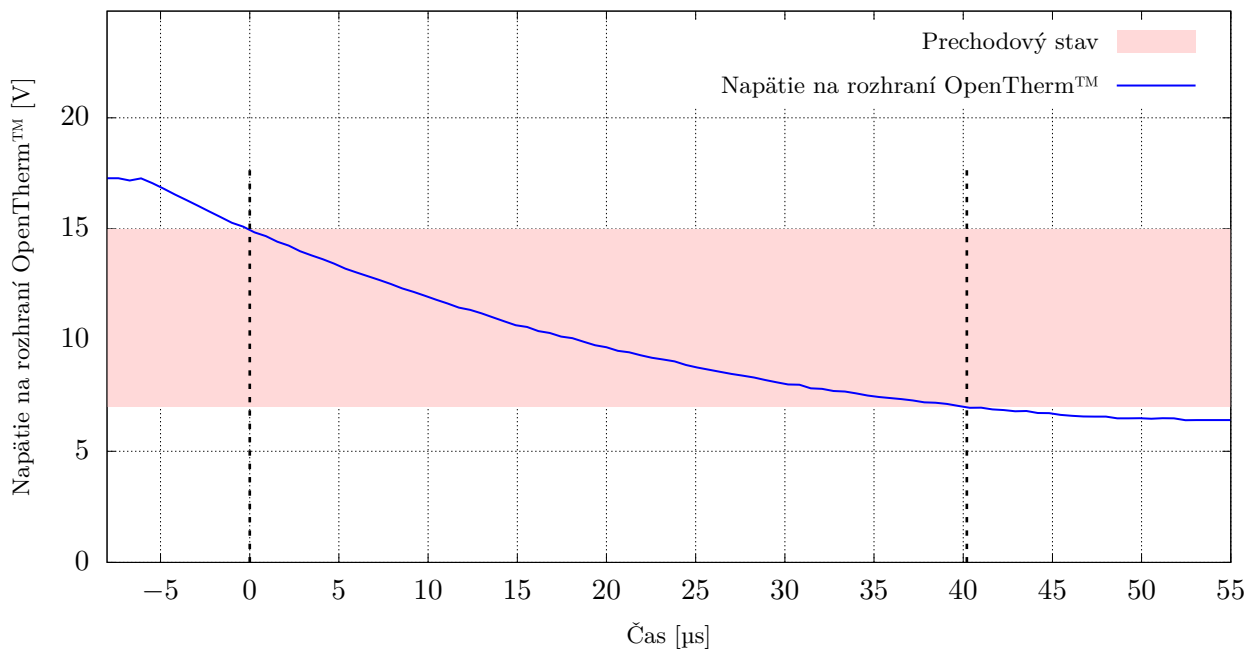
Testovanie

Pri návrhu zapojenia rozširujúceho modulu došlo k chybe. Tranzistor Q2 nemá pripojený na gate kanál pull down rezistor, a teda môže vplyvom elektromagnetického rušenia dôjsť k jeho samovoľnému zopnutiu, zopnutiu relé a teda aj kotla. Tento problém by bolo možné

riešiť softvérovou na počítači *Raspberry Pi 2*, a to nastavením pinu ovládajúceho gate kanál tohto tranzistora na 0. Toto riešenie by však nebolo optimálne: počas štartu počítača by bol kotol zapnutý. Preto bolo zvolené riešenie, kde je pomocou ladiaceho konektora J20 pripojený medzi gate kanál tranzistora Q2 a zem externý pull down rezistor.



Obr. 4.2: Vzostupná hrana signálu v doméne napätia na rozhraní OpenTherm™

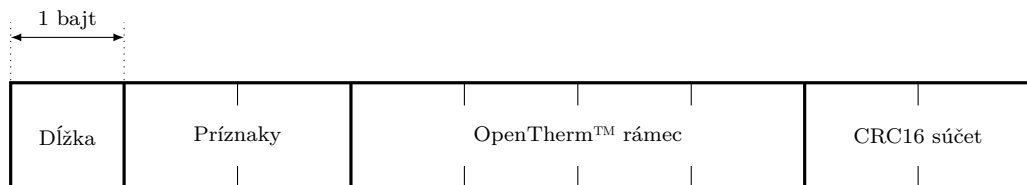


Obr. 4.3: Zostupná hrana signálu v doméne napätia na rozhraní OpenTherm™

Elektrické charakteristiky prevodníka boli otestované na prototype. Nakoľko prenosové charakteristiky v doméne prúdu závisia najmä na slave zariadení (kotol), boli testované len charakteristiky v doméne napätia. Testovanie prebiehalo pri komunikácií s reálnym zariadením (kotol *Baxi Prime 24*). Komunikačná linka bola vedená telefónnou štvorlinkou o dĺžke približne 25 m. Ako je zobrazené na obrázku 4.5, boli dodržané napätové úrovne požadované štandardom OpenTherm™ pre high level (15 V – 18 V) aj pre low level (≤ 7 V). Štandard OpenTherm™ ďalej definuje maximálne časy prechodu low \rightarrow high a high \rightarrow low. Tieto nesmú presiahnuť 50 μ s [27]. Merania ukázali, že čas prechodu zo stavu low do stavu high je kratší ako 15 μ s a čas prechodu zo stavu high do stavu low je približne 40 μ s. Napätové priebehy na rozhraní OpenTherm™ pri prechodoch medzi stavmi low a high sú zobrazené na obrázkoch 4.2 a 4.3. Dá sa teda konštatovať, že aj požiadavky na čas prechodu medzi logickými úrovňami kladené štandardom OpenTherm™ [27] boli splnené.

4.2.2 Firmvér

Firmvér mikrokontroléra zabezpečuje komunikáciu po zbernici I²C™ s počítačom *Raspberry Pi 2* a emuláciu rozhrania OpenTherm™ pre ovládanie kotla, a to ako v plnom režime, tak aj v režime spätnej kompatibility (OT/–) [27].

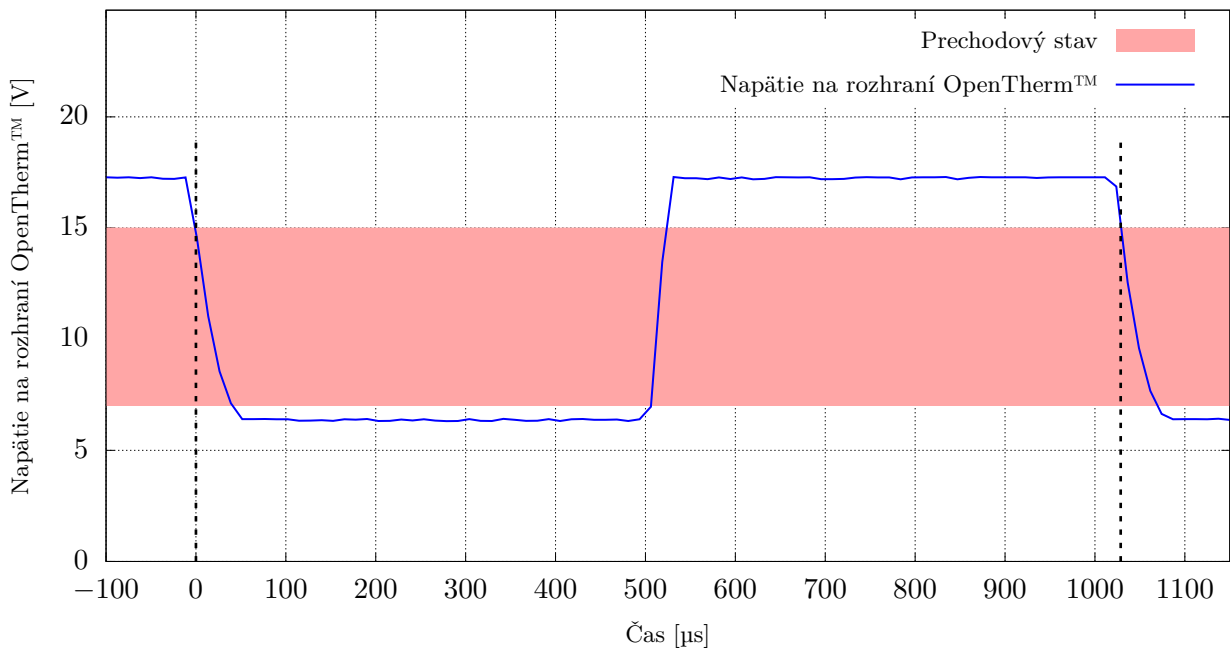


Obr. 4.4: Dátový rámec prechádzajúci po zbernici I²C™

Po zbernici I²C™ putujú medzi mikrokontrolérom a počítačom *Raspberry Pi 2* rámce definované nasledovne: prvý bajt každého rámca udáva počet nasledujúcich bajtov. V súčasnej dobe sú sice všetky rámce rovanko dlhé, no v budúcnosti to umožňuje jednoduchšie rozšírenie. Nasledujú 2 bajty príznakov, samotný 4 bajty dlhý OpenTherm™ rámec a 2 bajty dlhý kontrolný súčet Cyclic Redundancy Check (CRC). Dĺžka tohto súčtu je značne predimenzovaná, no v tomto prípade to ničomu neškodí. Príznačky slúžia na prenos chybových stavov a na ovládanie kotla v režime spätnej kompatibility (OT/–). Dátový rámec je zobrazený na obrázku 4.4.

Pre komunikáciu s kotlom mikrokontrolér emuluje rozhranie OpenTherm™. Na tento účel boli použité hardvérové časovače obsiahnuté v mikrokontroléri a tiež vonkajšie prerušenia. Štandard OpenTherm™ požaduje, aby bol čas odosielania jedného bitu v intervale 900 μ s – 1150 μ s. Časovanie generovaného signálu bolo overené meraním. Na obrázku 4.5 je zobrazený napätový priebeh na rozhraní OpenTherm™ pri posielaní jedného bitu. Je zrejmé, že čas odosielania jedného bitu, približne 1025 μ s, spadá do intervalu požadovaného štandardom OpenTherm™ [27].

Prevodník pre rozhranie OpenTherm™ pracuje na fyzickej a dátovej vrstve protokolu OpenTherm™, nezaobrá sa teda obsahom prenášaných správ. Prevodník správu na jednom rozhraní preberie a na druhom rozhraní odošle.



Obr. 4.5: Napätie na rozhraní OpenTherm™ v priebehu posielania jedného bitu

4.2.3 Softvér

V rámci tejto práce boli implementované dve knižnice pre *Raspberry Pi 2*, a to *otherm* a *mi2c*. Knižnica *mi2c* zabezpečuje komunikáciu medzi mikrokontrolérom a *Raspberry Pi 2* pomocou zbernice I²C™. Knižnica *otherm* abstrahuje komunikáciu s kotlom, vrátane iniciácie periodických konverzácií vyžadovaných protokolom OpenTherm™. Toto riešenie umožňuje jednoducho nahradiť zbernicu I²C™ inou, prípadne použiť bezdrôtový prenos.

Knižnica *mi2c* používa pre prístup ku GPIO pinom na *Raspberry Pi 2* existujúcu knižnicu *pigpio*. Knižnica *pigpio* [13] umožňuje vzdialený prístup po počítačovej sieti, čo je vhodné najmä pri vývoji a pri ladení, keď nie je nutné spúšťať program komunikujúci s kotlom priamo na *Raspberry Pi 2*.

Knižnica *otherm* zabezpečuje jednoduché aplikačné rozhranie pre komunikáciu s kotlom. V rámci tejto diplomovej práce bolo implementovaných niekoľko desiatok správ z aplikačnej vrstvy protokolu OpenTherm™ (viď prílohu A). Na binárnej úrovni je však možné kotlu poslať akúkoľvek správu.

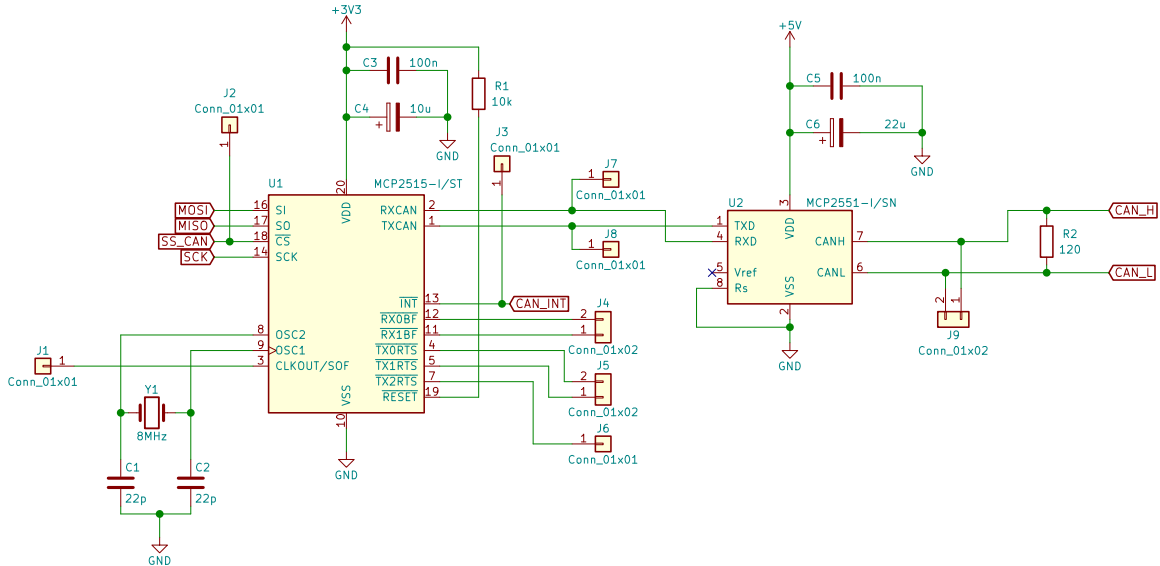
4.3 Rozhranie CAN

4.3.1 Hardvér

Kvôli potrebe rozšíriť *Raspberry Pi 2* o rozhranie CAN bol uskutočnený prieskum trhu s cieľom nájsť vhodný prevodník. Požadované parametre boli: jednoduchosť zapojenia, možnosť pripojenia pomocou GPIO konektora (ktorý neobsahuje USB) k *Raspberry Pi 2*, možnosť napájania napätím 3.3 V alebo 5 V a v neposlednom rade softvérová podpora.

Ako najlepším riešením sa zdala byť konštrukcia postavená na integrovaných obvodoch *MCP2515* a *MCP2551*. Obvod *MCP2515* je CAN kontrolér komunikujúci pomocou zber-

nice SPI [21]; obvod *MCP2551* je CAN transceiver [20]. Táto kombinácia nevyžaduje veľké množstvo externých komponentov. Obvod *MCP2515* je možné, za predpokladu, že jeho napájacie napätie dosahuje 3.3V, pripojiť priamo k *Raspberry Pi 2*. Transceiver vyžaduje dostupné napätie 5 V. Zmieňaná kombinácia obvodov sa používa aj v komerčne dostupných rozširujúcich moduloch pre *Raspberry Pi 2*.



Obr. 4.6: Schéma zapojenia rozširujúceho obvodu pre rozhranie CAN

V rámci tejto práce bolo navrhnuté zapojenie obvodu rozširujúceho počítač *Raspberry Pi 2* o rozhranie CAN, zobrazené na obrázku 4.6. Terminály SCK, MISO, MOSI a SS_CAN sú pripojené priamo k SPI zbernici *Raspberry Pi 2*. Terminál CAN_INT slúži na prenos prerušenia z *MCP2515* na GPIO pin počítača. Prakticky všetky zakreslené konektory sú určené najmä pre potreby ladenia. V prípade potreby je však možné pomocou nich pripojiť signály rôznych prerušení z *MCP2515* k GPIO portu počítača. Terminály CAN_H a CAN_L predstavujú diferenciálny pár zbernice CAN.

Zbernica CAN by mala byť na oboch koncoch ukončená rezistorom. V našom prípade to však nebolo možné, nakoľko sa predpokladá stromová topológia zbernice. Zbernica je teda zakončená len na jednej strane, a to za pomoci rezistora R2. V praxi sa ukázalo, že pri použitej frekvencii (100 kHz) a dĺžke zbernice to nie je problém. Pri predpokladanom objeme prenášaných dát je prenosová rýchlosť aj napriek nízkej frekvencii naddimenzovaná o niekoľko rádov.

4.3.2 Softvér

Medzi výhody obvodu *MCP2515* patrí najmä to, že má podporu v Linuxe, respektíve linuxovom jadre. Po správnom nastavení je teda možné k tomuto zariadeniu pristupovať ako ku ktorémukolvek inému sieťovému zariadeniu [6]. Pre prístup k tejto zbernici sa dobre osvedčila knižnica *python-can*, implementujúca vysokoúrovňový prístup k zbernici pomocou unixových socketov [28].

jednoduchšej káblovej antény o vhodnej dĺžke k tomuto konektoru je plne dostačujúce, naopak pri použití cesty na plošnom spoji boli výsledky mimoriadne neuspokojivé. Namiesto konektoru J25 by bolo možné pripojiť na dosku priamo koaxiálny kábel vedúci k externej anténe. Na plošný spoj by bolo možné doplniť aj skutočný koaxiálny konektor pre pripojenie externej antény.

4.4.2 Softvér

Pre komunikáciu s modulom *RFM69HW* existuje knižnica *rpi-RFM69*, vhodná pre počítač *Raspberry Pi 2*. Jej použitie sa však ukázalo byť problematické. Knižnica je značne nedokonalá, obsahuje veľké množstvo chýb. Niektoré z týchto chýb boli pre účel tejto práce opravené, no rozumnejším riešením by bola pravdepodobne kompletná reimplementácia tejto knižnice. Ideálnym riešením by bola implementácia ovládača pre modul *RFM69HW* priamo v jadre operačného systému, rozsah tejto práce to však nedovoľoval.

4.5 Celková konštrukcia rozširujúceho modulu

Vzhľadom na potrebu napájacieho napätia 3.3 V bol modul doplnený o lineárny regulátor *AZ1117-3.3* [8]. Nakoľko je celé zariadenie napájané napätím 5 V a špičkový odber na vetve 3.3 V nepresiahne 200 mA, je maximálna tepelná strata na tomto regulátore (nepresahujúca 0.4 W) akceptovateľná.

Na pripojenie zbernice pre komunikáciu s ostatnými zariadeniami v systéme (teplomery, servomechanizmy ovládania ventilov), kombinujúcej zbernicu 1-Wire, CAN a napájacie napätia 3.3 V a 5 V, je použitý konektor *RJ45*. Na plošnom spoji sú však vyvedené aj konektory J34 a J36 (viď prílohu B.1) pre pripojenie zbernice iným spôsobom.

Pre pripojenie k počítaču *Raspberry Pi 2* bol použitý štandardný 40kolíkový dvojradý konektor a plochý kábel, aký poznáme z pevných diskov pripojených pomocou zbernice *ATA*.

Celkovo bol rozširujúci modul konštruovaný s veľkým dôrazom na možnosti ladenia a eventuálneho rozšírenia. Prakticky každý podstatný signál je vyvedený na niektorý z konektorov (kolíkové lišty). Toto sa ukázalo byť správnym rozhodnutím. Vďaka tomuto rozhodnutiu bolo možné jednoducho doplniť chýbajúci pull down rezistor na gate elektródu tranzistora Q2, ovládajúceho relé K1 pre historické riadenie kotla. Vôbec prítomnosť tohto relé umožňuje ovládať kotol v prípade nefunkčnosti alebo neosadení obvodov súvisiacich s *OpenTherm™* rozhraním.

Modul bol skonštruovaný na dvojvrstvovom plošnom spoji o veľkosti 140 mm × 56 mm. Pri oživovaní modulu neboli zaznamenané žiadne ďalšie chyby v návrhu. Hneď prvý vyrobený prototyp bol plne funkčný. Pre účely minimalizácie ceny, prípadne sériovej výroby, by bolo možné vypustiť značné množstvo konektorov, prípadne použiť súčiastky menších rozmerov.

Kapitola 5

Servomechanizmus ovládania ventilu radiátora

Aby bolo možné nezávisle riadiť vykurovanie v jednotlivých miestnostiach, je v prípade použitia bežných radiátorov ústredného kúrenia nutné ovládať ventily týchto radiátorov. Dnes už je na trhu nepreberné množstvo hlavíc pre radiátorové ventily, či už termostatických alebo diaľkovo ovládaných. Existujú v bezdrôtových aj káblových prevedeniach. Na trhu sa však nenachádza, ako ukázal prieskum, radiátorový ventil s príslušnou hlavicom, ktorý by mohol poslúžiť ako priama náhrada starých radiátorových ventilov typu *Myjava 1 1/2*". Dostupné ventily majú oproti pôvodným ventilom znížený prietok, a teda ich nie je možné bez ďalších úprav vykurovacieho systému použiť. Preto bolo rozhodnuté o návrhu a výrobe servomechanizmov, ktoré budú ovládať tieto ventily starého typu. Ich kompletná dokumentácia je uvedená v prílohe C.

5.1 Mechanická konštrukcia servomechanizmu

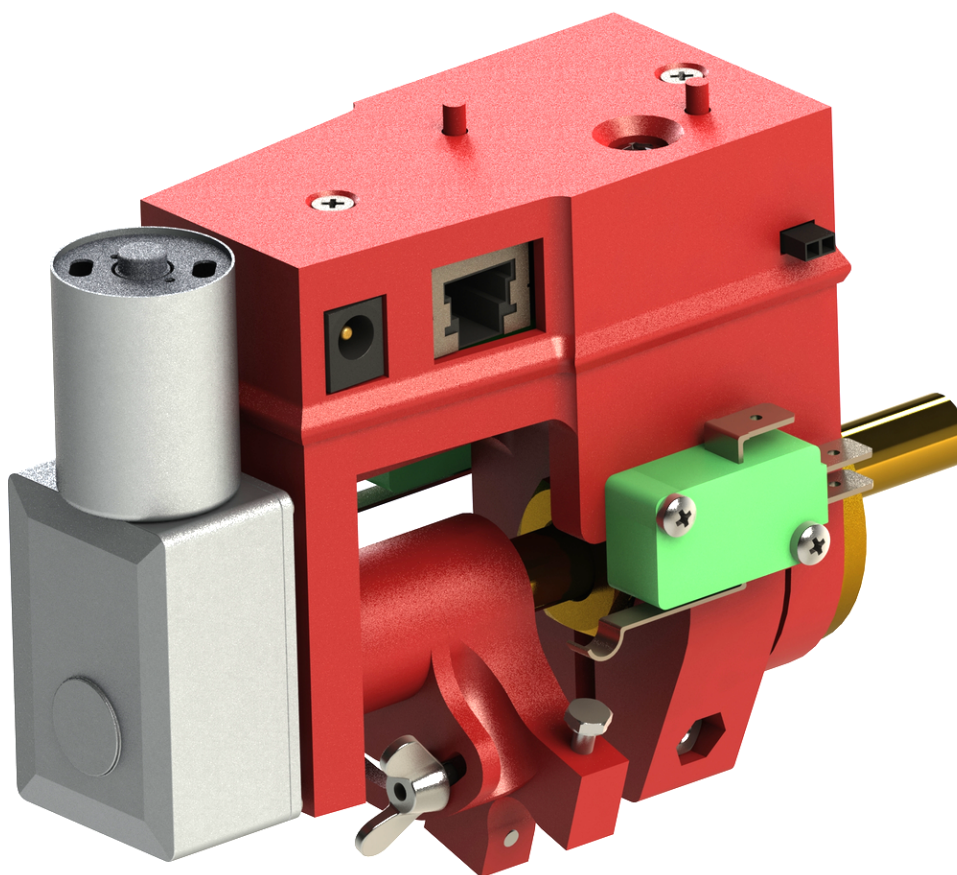
V rámci merania bolo zistené, že pre otočenie ventilu je potrebný krútiaci moment približne 0.5 N m. Nakoľko majú niekedy ventily tendenciu zatuhnúť, najmä keď sa s nimi dlhšie nehýbe, bolo rozhodnuté o skonštruovaní servomechanizmu, ktorý poskytne krútiaci moment aspoň 2 N m.

5.1.1 Pohon

Pri prieskume trhu bolo zistené, že hotové servopohony disponujúce požadovaným krútiacim momentom sú pomerne drahé, a to aj za predpokladu, že by neboli pripojené na ventil priamo, ale cez dodatočný prevod. Ukázalo sa, že aj elektromotory vybavené prevodovkou a so zodpovedajúcimi parametrami od renomovaných dodávateľov sú pomerne drahé. Bolo teda rozhodnuté o použití neznačkových jednosmerných elektromotorov s prevodovkou. Ich krútiaci moment dosahuje 2.2 N m, voľnobežné otáčky sú 30 ot/min, napájacie napätie 12 V a maximálny prúd (pri zablokovaní motora) 2 A.

5.1.2 Materiál

Od konštrukcie motora s prevodovkou a ventilu sa odvíja konštrukcia samotného servomechanizmu. Pri konštrukcii bol kladený dôraz na vyrobiteľnosť pomocou bežnej 3D tlačiarne a tiež na to, aby bol použitý len štandardný spojovací materiál (viď prílohu C.2.1). Ako



Obr. 5.1: 3D model servomechanizmu ventilu

materiál pre výrobu plastových dielov bol zvolený termoplast PETG. Tento materiál je pomerne húževnatý a pevný. Oproti plastu ABS má menšiu tepelnú rozťažnosť a teda pri výrobe na 3D tlačiarňi nedochádza k tak veľkým deformáciám a je možné dosiahnuť vyššiu presnosť. Voľba tohto materiálu však prináša aj isté obmedzenia. Nakoľko sa jedná o termoplast, jeho pevnosť so zvyšujúcou sa teplotou klesá. Servomechanizmus vyrobený z tohto materiálu teda pravdepodobne nebude použiteľný pri vysokej teplote vykurovacieho média (približne nad 80 °C). Takto vysoké teploty vykurovacieho média však zvyčajne nie sú potrebné a iba zvyšujú energetickú náročnosť. Samozrejme by bolo možné použiť aj iný materiál, prípadne inú technológiu výroby, no zhotovenie prototypu by bolo náročnejšie a drahšie.

5.1.3 Detekcia koncovej polohy ventilu

V rámci servomechanizmu bolo potrebné zabezpečiť detekciu koncovej polohy ventilu. Bolo uvažované o dvoch možných spôsoboch takejto detekcie. Jedným riešením by mohla byť inštalácia koncových spínačov signalizujúcich koncovú polohu ventilu. Tieto spínače by však bolo nutné kalibrovať pre každý ventil zvlášť. Jednoduchším a elegantnejším riešením by bolo vyhodnocovanie dosiahnutia koncovej polohy pomocou odberu elektromotora. Neboli by potrebné žiadne koncové spínače, odpadla by potreba kalibrácie týchto spínačov, znížil by sa počet dielov a zjednodušila konštrukcia. Problémom by mohlo byť chybné vyhodnotenie

dosiahnutia koncovej polohy v prípade, že by bol ventil zaseknutý. Testy ukázali, že slabým miestom konštrukcie je (za predpokladu, že je vyrobená z PETG) spojka medzi hriadeľom motora prierezu "D" (respektíve k nemu pripojenej prevodovky) a osou ventilu štvorcového prierezu (viď prílohu C.2.2). Pri vyhodnocovaní krajnej polohy ventilu pomocou veľkosti prúdu pretekajúceho motorom dochádzalo k nadmernému opotrebeniu tejto spojky. Nepodarilo sa skonštruovať a vyrobiť z materiálu PETG tak pevnú spojku, aby bola použiteľná pri metóde používajúcej veľkosť prúdu. Dostatočne pevnú spojku by bolo samozrejme možné vyrobiť z nejakého vhodnejšieho materiálu (mosadz, dural, oceľ a podobne), no bolo by to po výrobnej stránke omnoho náročnejšie. S ohľadom na túto skutočnosť bola zvolená varianta obsahujúca koncové spínače.

Pri výbere koncových spínačov bolo rozhodované medzi použitím optozávor a mechanických mikrospínačov. Vzhľadom na robustnosť a jednoduchosť konštrukcie a jej odolnosť voči prachu a nečistotám bola zvolená varianta s mikrospínačmi. Boli použité mimoriadne naddimenzované mikrospínače, minimálne čo sa týka možného spínaného prúdu (až 16 A). K ich výberu viedla najmä ich mechanická konštrukcia a možnosť ich pripevniť pomocou skrutiek M3. Menšie a elegantnejšie mikrospínače by vyžadovali použitie menších, relatívne ťažko dostupných skrutiek, a pri výrobe použitie drahých a relatívne ťažko dostupných závitníkov. Prípadne by bolo nutné konštruovať nejaké iné mechanizmy pre pripevnenie týchto spínačov.

Koncové spínače je možné kalibrovať pomocou nastavovacích skrutiek. Aby sa tieto skrutky nemohli samovelne otáčať, sú umiestnené v dierach s neúplne dorezaným závitom (boli použité len závitníky 1 a 2 z trojdielnej sady). Bolo by možné aj použitie kontramatíc, no ich nastavovanie by sa stalo menej pohodlným.

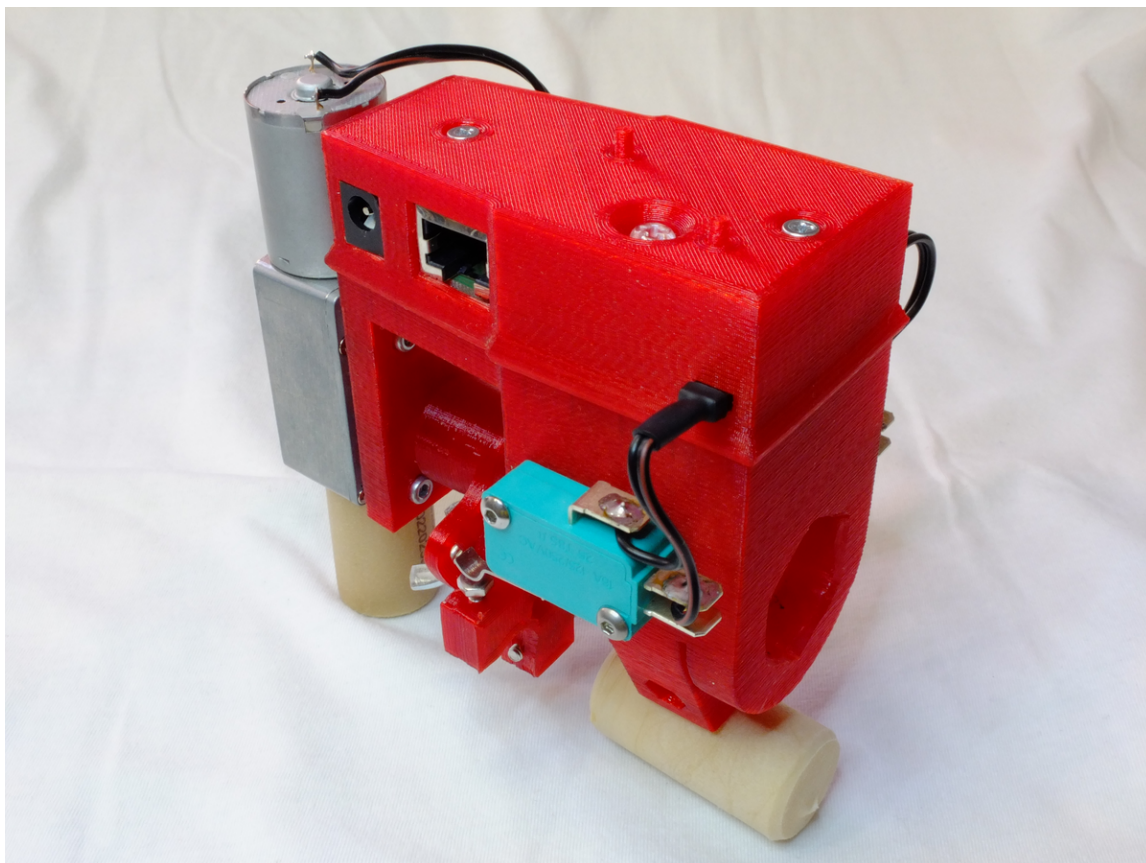
5.1.4 Rozmiestnenie komponentov

Pri konštrukcii mechanizmu bolo myslené aj na umiestnenie riadiacej elektroniky a otvorov pre konektory na pripojenie koncových vypínačov, motora, zbernice a napájania. Tiež sa počítalo s umiestnením tlačidiel pre ovládanie mechanizmu a svetlovodom pre signalizačnú LED diódu. Ako svetlovod bola použitá polykarbonátová skrutka M4×14 so šošovkovitou hlavou zapustená do veka elektroniky.

Pri návrhu mechanizmu bolo vyrobených a otestovaných niekoľko rôznych prototypov. Vývoj došiel až k mechanizmu znázornenému na obrázkoch 5.1 (3D model) a 5.2 (fotografia výsledného prototypu).

5.2 Hardvér

Pre účely riadenia servomechanizmu bol navrhnutý elektrický obvod znázornený na obrázku 5.3. Nakoľko je žiadané, aby servomechanizmus komunikoval s riadiacim počítačom pomocou zbernice CAN, bol zvolený mikrokontrolér *Atmel ATmega32m1*, ktorý toto rozhranie obsahuje [4]. Je veľmi podobnej architektúry ako mikrokontrolér použitý pre OpenTherm™ rozhranie, no obsahuje iné periférne zariadenia. Pre pripojenie k zbernici CAN je nutné použiť CAN transceiver. Tak isto ako pri rozširujúcom module pre počítač *Raspberry Pi 2*, bol použitý obvod *MCP2551* [4, 20].



Obr. 5.2: Fotografia servomechanizmu ventilu

5.2.1 Riadenie motora

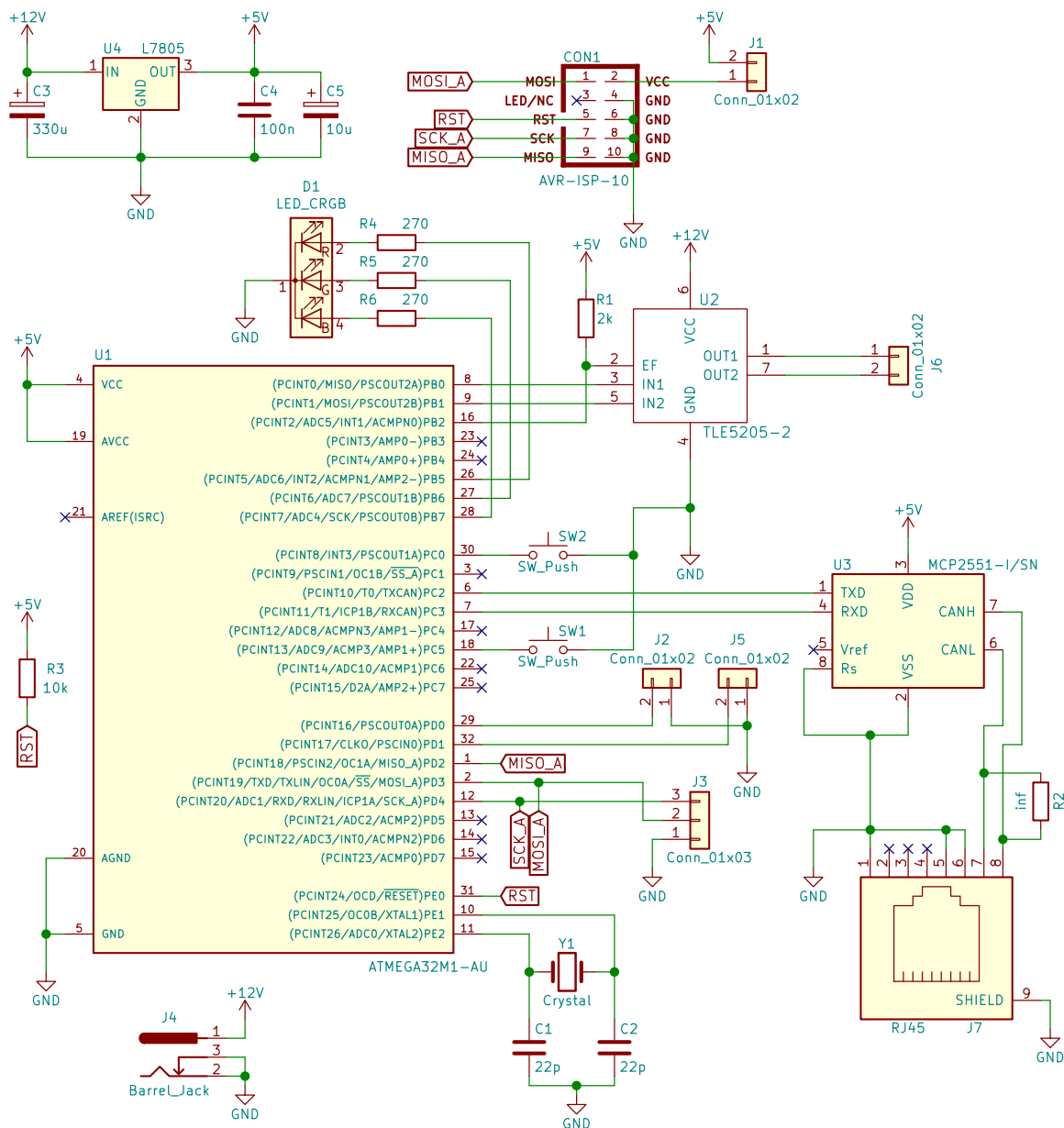
Pre účely spínania motora bol použitý H mostík *TLE2502-2G* [11]. Tento H mostík je schopný spínať trvalé prúdy až do 5 A. Nakoľko motor odoberá maximálne 2 A a je zopnutý maximálne jednotky sekúnd, a to aj v prípade, že je ventil zablokovaný, prípadne je vadný koncový vypínač, nie je potrebné k tomuto obvodu navrhovať chladič. Vystačí si s medenou plochou na plošnom spoji [11].

5.2.2 Pripojenie k zvyšku systému

Pre účely pripojenia ku zbernici CAN je obvod vybavený konektorom *RJ45*. Koncové spínače sú pripojené pomocou konektorov *J2* a *J5*, motor je pripojený pomocou konektora *J6*. Spínače *SW1* a *SW2* slúžia pre manuálne nastavenie polohy mechanizmu. Pre účely programovania mikrokontroléra bol osadený konektor *CON1* a pre účely ladenia konektor *J3*, na ktorý je z mikrokontroléra vyvedená sériová linka.

5.2.3 Napájanie

Nakoľko má použitý elektromotor pomerne veľký odber (až do 2 A), bolo pristúpené k riešeniu, kde má každý takýto servomechanizmus vlastný adaptér 230 V AC/12 V DC. Kvôli vysokému odberu nebolo možné napájať zariadenie priamo po komunikačnej zbernici vedenej UTP (Cat 5e) káblom za predpokladu, že by bolo po tejto zbernici šírené napätie



Obr. 5.3: Schéma zapojenia servomechanizmu ventilu

12 V. Kábel danej kategórie totiž umožňuje viesť prúd len 0.577 A, a to pri nezanedbateľnom odpore vedenia. Bolo by síce možné použiť DC/DC prevodníky a po zbernici distribuovať vyššie napätie, no toto zariadenie by už nespadlo do kategórie SELV / PELV, bolo by nutné riešiť izolačné medzery, zapojenie by sa stalo zložitejším a s najväčšou pravdepodobnosťou by to viedlo k vyššej cene. Tiež by bolo možné distribuovať po zbernici 12 V a potrebnú energiu uchovávať v akumulátore, a to v každom servomechanizme. Toto riešenie bolo vyhodnotené ako zbytočne zložité. Bolo by totiž potrebné implementovať nabíjacie obvody a ochrany proti preťaženiu akumulátora. Životnosť zariadenia by tiež bola obmedzená životnosťou akumulátora.

Napätie 5 V pre logické obvody je získavané pomocou regulátora *L7805*. Pretože je odber logických obvodov relatívne nízky, ani množstvo tepla produkovaného týmto regulátorom

mie interval, kedy vysielateľ vysielá na zbernicu recesívny signál (1) a čaká na potvrdenie prijatia rámca prijímačom, ktorý v prípade úspešného príjmu vyšle na zbernicu dominantný signál (0). V ACK slotě je prenášaný posledný podstatný bit rámca. Začiatok tohto intervalu (za predpokladu, že by bola zbernica časovaná presne) je zobrazený prvou fialovou bodkočiarkovanou vertikálou z ľavej strany. Z obrázku je zrejme, že rámec končí až v čase 988 μ s, a teda o 78 μ s neskôr, ako bolo očakávané (druhá fialová bodkočiarkovaná vertikála z ľavej strany). Práve priebeh signálu medzi týmito dvomi vertikálami (teda na konci chybného rámca) je zaujímavý, nakoľko pri zvolenom rámci by mal mať formu "...010101". Tiež sú v signále pozorovateľné podozrivé medzistavy. Nakoľko dĺžka vysielania rámca nezodpovedala vypočítanej dĺžke (bola o približne 8.6% dlhšia), pravdepodobným zdrojom tohto problému bol interný RC oscilátor mikrokontroléra, ktorý bol použitý ako zdroj hodinového signálu pre tento mikrokontrolér.

V elektrickom zapojení figuruje aj kryštál Y1 (viď obrázok 5.3) s frekvenciou 8 MHz. S jeho osadením sa však nepočítalo. Do zapojenia bol umiestnený len ako poistka pri problémoch s časovaním, prípadne ako zdroj hodín pre mikrokontrolér v prípade, že bude tento omylom naprogramovaný pre použitie externého kryštálu. Bez hodinovej frekvencie totiž nie je možné mikrokontrolér tohto typu preprogramovať. Ako pokus pri riešení vzniknutej situácie bol kryštál Y1 osadený a použitý ako zdroj hodinového signálu. Táto zmena problémy s časovaním odstránila. Signál nameraný na zbernici CAN pri použití kryštálu Y1 je zobrazený modrou farbou.

Ďalším problémom v obvode je chybné zapojenie RGB LED diódy. Schéma počíta s tým, že RGB LED dióda je v prevedení so spoločnou katódou. Objednaná dióda však bola v prevedení so spoločnou anódou. Bolo zistené, že žiadna kompatibilná dióda (prevedenie púzdra, rozmiestnenie pinov) nie je jednoducho dostupná. Ako riešenie bolo zvolené umiestnenie tejto diódy na plošný spoj tak, že je otočená o 180°. Pomocou vhodnej voľby rezistorov R4 – R6 a vhodného softvéru je možné rozsvietiť každú z troch integrovaných diód osobitne, nie však naraz. Prispieva k tomu rôzny pokles napätia na jednotlivých diódach.

5.3 Firmvér

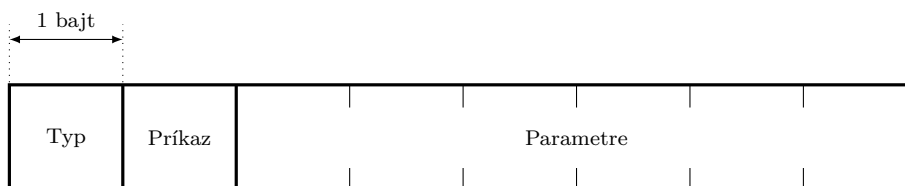
Pre dosiahnutie cieľov práce bolo potrebné implementovať firmvér pre mikrokontrolér *Atmel ATmega32m1* v servomechanizme. Pretože obsluha integrovaného rozhrania CAN nie je triviálna, bola pre účely ovládania tohto rozhrania použitá už existujúca knižnica *avr_can*, dostupná pod licenciou LGPL-2.1. Tento prístup sa ukázal byť dobrou voľbou, nakoľko je daná knižnica napísaná relatívne dobre a nebolo potrebné opravovať príliš veľa chýb.

5.3.1 Komunikácia po zbernici CAN

Každé vyrobené zariadenie má pridelený unikátny identifikátor, obdobným spôsobom ako MAC adresy pri Ethernete. Tento identifikátor je uložený v pamäti EEPROM v mikrokontroléri. Tiež bolo rozhodnuté, že budú povolené len dva smery komunikácie, a to z riadiaceho počítača na konkrétny servomechanizmus (prípadne nejaké iné zariadenie) a naopak. Priama komunikácia medzi servomechanizmami nie je podporovaná. Je známe, že zbernica CAN nepodporuje adresovanie, no podporuje arbitráciu [29]. Každá správa je vybavená identifikátorom, zvaným arbitration ID, ktorý je pri použití normy CAN 2.0B dlhý 29 bitov. Čím nižšie je toto číslo, tým vyššiu prioritu z pohľadu arbitrácie zbernice má daná správa. Arbitration ID CAN zbernice bolo použité na adresovanie zariadení. Každé zariadenie (servomechanizmus) má z výroby pridelený 28 bitov dlhý identifikátor, ktorý sa pri komunikácii

v smere počítač → servomechanizmus použije ako arbitration ID. V opačnom smere sa ako arbitration ID použije identifikátor zariadenia zvýšený o 2^{28} (teda na mieste 29. bitu bude 1). To zaručuje, že riadiaci počítač (*Raspberry Pi 2*) vie, odkiaľ prišla správa, a že správy v smere počítač → servomechanizmus majú vyššiu prioritu.

V rámci tejto diplomovej práce bol navrhnutý protokol na prenos správ po zbernici CAN. Dátový rámec navrhnutého protokolu je znázornený na obrázku 5.5. Prvý bajt špecifikuje typ správy, druhý bajt špecifikuje príkaz a následovných 0 – 6 bajtov sú parametre. Protokol je typu výzva-odpoveď. Jedno zariadenie teda pošle výzvu a čaká na odpoveď. Ak odpoveď nepríde do daného časového limitu, považuje akciu za nevykonanú. Za predpokladu, že bola akcia vykonaná správne, má odpoveď bajty *typ* a *príkaz* nastavené zhodne ako výzva. V prípade zlyhania je na ich mieste použitý vhodný chybový kód.



Obr. 5.5: Dátový rámec prechádzajúci po zbernici CAN

Aby bolo možné pridávať zariadenia do systému dynamicky, bol navrhnutý a implementovaný mechanizmus na vyhľadávanie zariadení pripojených na zbernicu. Pre tento účel bola vyhradená adresa (arbitration ID) 0, ktorá nesmie byť pridelená žiadnemu zariadeniu. Počítač *Raspberry Pi 2* na túto adresu vyšle výzvu, aby sa všetky pripojené zariadenia ozvali. Zariadenia následne odpovedajú počítaču. Nakoľko však pri odpovediach už používajú ako arbitration ID svoj identifikátor zvýšený o 2^{28} , počítač zistí ktoré zariadenia sú pripojené na zbernicu. Ak dôjde ku kolízii na zbernici, zariadenie automaticky opakuje pokus o odpoveď.

Zatiaľ boli implementované len metódy pre otvorenie a zatvorenie ventilu a vyhľadanie zariadení na zbernici. V budúcnosti by však bolo možné implementovať metódy ďalšie.

5.3.2 Prevencia problémov

Firmvér počíta s tým, že na zariadení môže nastať porucha, prípadne samotný ventil môže byť zablokovaný. V prípade poruchy na koncovom vypínači by sa mohlo ľahko stať, že elektromotor pohybujúci ventilom by zostal dlhodobo zopnutý z zablokovanej pozícií. To by pravdepodobne viedlo k deštrukcii motora, prípadne H mostíka ktorý ho spína. Tomuto stavu bolo predídene tým, že bol stanovený časový limit, do ktorého musí byť akcia spojená s pohybom ventilu dokončená. Ak je tento časový limit prekročený, akcia je prerušená a je odoslaná odpoveď definujúca tento chybový stav.

5.4 Montáž

Montáž prebieha tak, že sa servomechanizmus narazí na šesťhrannú časť samotného ventilu. Válcová časť je zovretá v klieštine pomocou skrutky. Vďaka nastaviteľnému uhlu medzi spojkou hriadeľa motora s osou ventilu je možné mechanizmus namontovať na ventil v akejkoľvek pozícií na šesťhrane ventilu, teda šiestimi rôznymi spôsobmi.

Pri montáži servomechanizmu na ventil radiátora a kalibrácií koncových vypínačov je potrebné sústrojím pootáčať. To však inak ako pomocou elektromotora nie je možné, pretože prevodovka k nemu pripojená je samosvorná. Pre pootáčanie mechanizmom pri montáži slúžia tlačidlá umožňujúce obojstranný pohyb. Pri testovaní bolo zistené, že pri montáži nie je vhodné, aby sa mechanizmus otáčal plnou silou, lebo to viedlo k deštrukcii už spomínanej spojky (viď prílohu C.2.2). Výkon motora je teda pri ručnom ovládaní obmedzený pomocou PWM.

Kapitola 6

Bezdrôtový teplotný senzor

Nakoľko musí byť systém nasaditeľný v už stojacich budovách bez nutných náročných stavebných úprav, je potrebné, aby niektoré prvky komunikovali bezdrôtovo. Na trhu sú síce dostupné rôzne bezdrôtové senzory teploty, no tie sú zvyčajne obmedzené voľbou bezdrôtového rozhrania. Vzhľadom na voľbu modulu *RFM69HW* pre bezdrôtový prenos, opísanú v kapitole 4.4, bol navrhnutý a vyrobený teplotný senzor používajúci zodpovedajúcu technológiu, viď prílohu D.

6.1 Hardvér

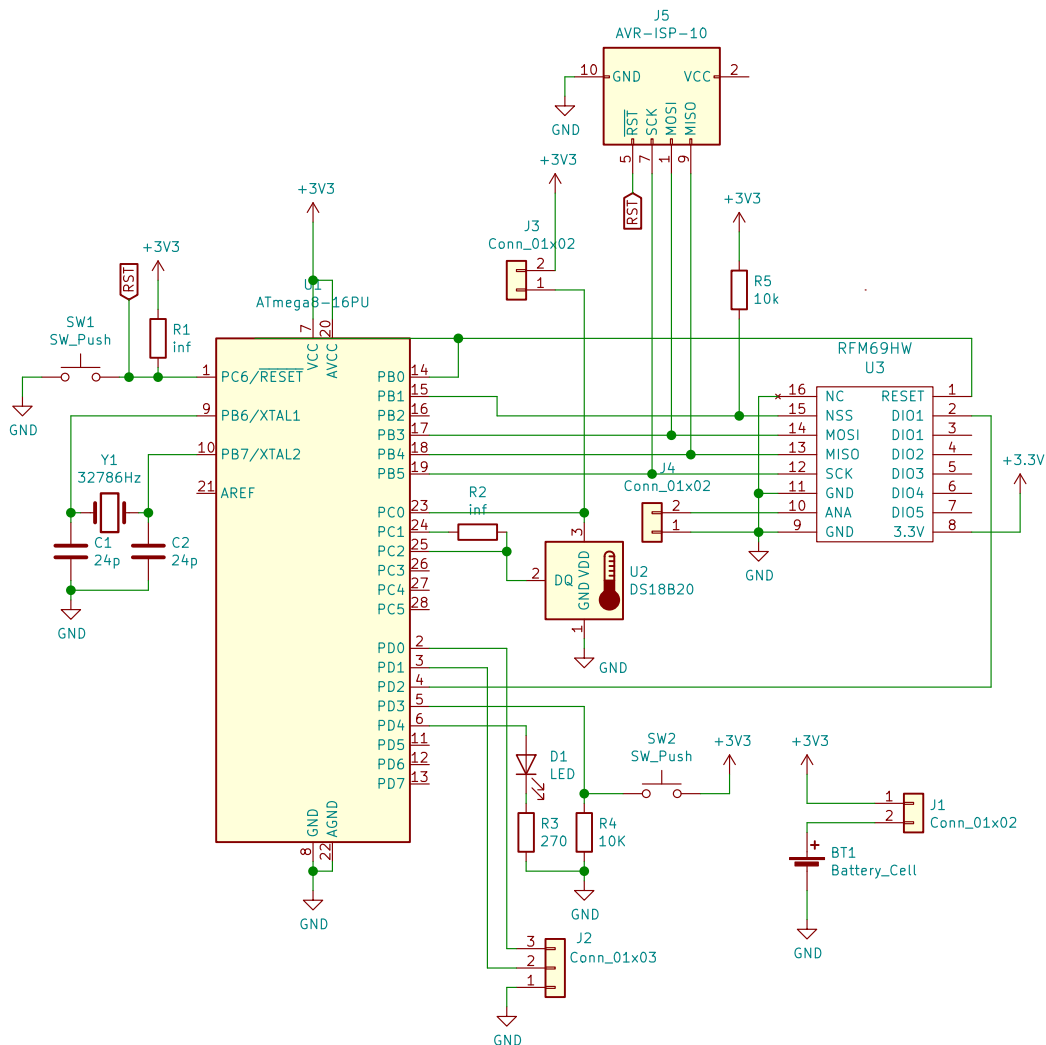
6.1.1 Komponenty

Pre účely tejto práce bolo navrhnuté zapojenie teplotného senzora znázornené na obrázku 6.1. V implementácii bol použitý mikrokontrolér *Atmel ATmega328*, komunikačný modul *RFM69HW* a teplotný senzor *DS18B20*. Pôvodne sa počítalo s využitím teplotného senzora obsiahnutého v module *RFM69HW*, no toto sa neukázalo byť správnou voľbou. Teplotný senzor v tomto module nie je z výroby nakalibrovaný, a ani po kalibrácii nie je senzor použiteľný kvôli nedostatočnej presnosti a produkovanému šumu. Z tohto dôvodu bola zvolená varianta používajúca teplotný senzor *DS18B20*. Senzor *DS18B20* používa napájacie napätie kompatibilné s použitým mikrokontrolérom a komunikačným modulom [16, 1, 10]. Senzor komunikuje s mikrokontrolérom pomocou zbernice 1-Wire. Zapojenie ďalej obsahuje dve tlačidlá pre reset a inicializáciu zariadenia.

6.1.2 Napájanie

Ako zdroj energie bola použitá primárna batéria *Saft LS14500CNA*. Jedná sa o batériu typu Li-SOCl₂ s menovitým napätím 3.6 V a kapacitou 2.7 A h [26]. Použitie tohto typu batérie umožňuje konštrukciu zariadenia bez napäťového regulátora, čo znižuje spotrebu energie.

Pri návrhu zariadenia bol kladený veľký dôraz na to, aby mohlo byť dlhodobo napájané batériou. Pretože nie je nutné, aby teplotný senzor bežal stále, je napájaný priamo pomocou I/O pinu mikrokontroléra. Toto riešenie však prináša aj nevýhodu spojenú s nutnosťou vyššieho napájacieho napätia potrebného pre činnosť teplotného senzora. Časť napätia sa totiž stratí na tranzistoroch ovládajúcich daný pin mikrokontroléra. Obvod je navrhnutý tak, aby bolo možné napájanie teplotného senzora pripojiť trvalo na napájacie napätie, a to pomocou prepójky na konektore J3. Tiež bol osadený krištál presného času s nízkou



Obr. 6.1: Schéma zapojenia bezdrôtového teplotného senzora

frekvenciou (32 768 Hz), ktorý je vhodný ako zdroj hodinového signálu pre mikrokontrolér v režime spánku.

Zbernica 1-Wire vyžaduje použitie pull up rezistora. Bol použitý interný pull up rezistor obsiahnutý v mikrokontroléri. Nakoľko je zbernica krátka, nepredstavuje to problém. Zapojenie je navrhnuté tak, že je možné použiť aj externý pull up rezistor. Pretože vďaka odpájaniu teplotného senzora od napájania nie je nutné, aby bola udržiavaná 1-Wire zbernica pod prúdom, je pri nepoužívaní zbernice odpojený aj tento pull up rezistor.

Zariadenie nedisponuje meraním a signalizáciou nízkeho stavu batérie. Je však navrhnuté tak, že keď napätie batérie klesne pod kritickú hranicu, začne merať vyššiu teplotu ako je skutočná teplota. Nehrozí teda stav, že vplyvom nízkeho napätia na batérii zbytočne aktivuje kúrenie. V budúcnosti je možné zariadenie o funkcionalitu merania stavu batérie doplniť.

6.1.3 Prototyp

Pre demonštráciu funkčnosti celého zariadenia bol zhotovený a oživený prototyp bezdrôtového teplotného senzora. Vzhľadom na jednoduchosť výroby bol tento prototyp zhotovený ako jednovrstvový plošný spoj, používajúci technológiu THT. Takýto plošný spoj bolo možné vyrobiť na CNC frézke, ktorá bola autrovi dostupná. Neskôr bol spoj objednaný a vyrobený profesionálne.

6.2 Firmvér

6.2.1 Komunikácia

Na ovládanie bezdrôtového modulu bola použitá existujúca knižnica *RFM69-Library-AVR*. Táto knižnica však (podobne ako knižnica pre *Raspberry Pi 2*) obsahuje veľa nedostatkov a množstvo chýb. Niekoľko takýchto nedostatkov a chýb muselo byť v rámci práce odstránených. Pravdepodobne by bolo lepším riešením túto knižnicu reimplementovať. Vzhľadom na rozsah práce to však nebolo možné.

Pretože mikrokontrolér *Atmel ATmega328* [1] neobsahuje hardvérovú implementáciu rozhrania 1-Wire, je toto rozhranie softvérovo emulované. K tomuto účelu bola použitá existujúca knižnica *avr-ds18b20*, ktorá sa ukázala byť plne dostačujúca.

6.2.2 Režim spánku

Vzhľadom na šetrenie energie je mikrokontrolér aj bezdrôtový modul uvádzaný do režimu spánku. Teplotný senzor je odpájaný od napájania úplne. Mikrokontrolér je taktovaný hodinovým signálom o frekvencii 8 MHz, získavaným z interného oscilátora. Počas spánku sú v prevádzke len nutné periférne zariadenia, samotné jadro procesora stojí. Časovač je taktovaný externým křišťálom o frekvencii 32 768 Hz. Počíta sa s tým, že toto zariadenie bude komunikovať s riadiacim počítačom raz za 5 minút, prípadne raz za iný časový interval v rádoch minút. Zvyšný čas bude v režime spánku.

6.2.3 Synchronizácia

Pre účely synchronizácie s riadiacim počítačom *Raspberry Pi 2* malo pôvodne slúžiť tlačidlo SW2, po ktorého stlačení mal byť mikrokontrolér zobudený a mal konať kroky súvisiace so synchronizáciou. Vzhľadom na chybu v návrhu zapojenia tlačidla SW2 to však nie je možné. Z režimu spánku síce je možné mikrokontrolér prebrať externým prerušením, no toto prerušenie je možné v režime spánku generovať len nastavením napätovej hladiny na danom pine na logickú 1. Prerušenie nie je možné generovať hranou signálu, prípadne logickou hladinou 0. Tlačidlo SW2 je však chybné zapojené tak, že pri jeho stlačení je na danom pine logická 0.

Pre započatie synchronizácie je teda potrebné stlačiť tlačidlá SW1 a SW2 naraz (tlačidlo SW1 resetuje mikrokontrolér) a následne uvoľniť tlačidlo SW1. Mikrokontrolér vstúpi do synchronizačného režimu, čo je indikované LED diódou D1. V synchronizačnom režime sa zariadenie snaží po dobu približne 65s pomocou broadcastu kontaktovať riadiaci počítač. V prípade úspešného kontaktu riadiaci počítač zariadeniu pošle jeho node ID a AES-128 šifrovací kľúč. Po úspešnom prijatí, indikovanom LED diódou D1, sú tieto informácie uložené v pamäti EEPROM mikrokontroléra a použité pre ďalšiu komunikáciu. Samotné šifrovanie

prebieha priamo v bezdrôtovom module *RFM69HW* [10]. Všetka komunikácia od tohto momentu je šifrovaná a zariadenie ďalej pokračuje v normálnej činnosti.

6.2.4 Šifrovanie

Prenos šifrovacieho kľúča v nešifrovanej podobe verejným kanálom nie je správnym riešením. Pôvodne bolo uvažované o použití asymetrickej kryptografie pre prenos symetrického kľúča. Komunikujúce zariadenia by si najprv vymenili verejné kľúče a ďalej by mohli preniesť symetrický kľúč bezpečne šifrovaný. Vzhľadom na to, že nebola nájdená použiteľná šifrovacia knižnica pre použitý mikrokontrolér a rozsah práce implementáciu takejto knižnice neumožňoval, nebolo asymetrické šifrovanie použité. Možnosť odchytiť šifrovací kľúč je však malá, nakoľko sa prenáša jednorázovo. Iným riešením by mohlo byť prenesenie šifrovacieho kľúča iným kanálom, napríklad pomocou infračerveného svetla. Túto technológiu poznáme napríklad z diaľkového ovládania domácich elektrospotrebičov.

Počítanie kľúčového páru RSA (prípadne iného algoritmu) o bezpečnej dĺžke by na tak nevykonnom zariadení, akým je použitý mikrokontrolér *Atmel ATmega328*, trvalo príliš dlho. Tiež by bolo potrebné pre tieto účely pripojiť externú pamäť. Počíta sa teda s tým, že každé zariadenie by obsahovalo v pamäti EEPROM uložený unikátny pár kľúčov, predpočítaný z výroby. Samotné šifrovanie a dešifrovanie už aj na takto pomalom a malom mikrokontroléri možné je.

6.3 Spotreba energie

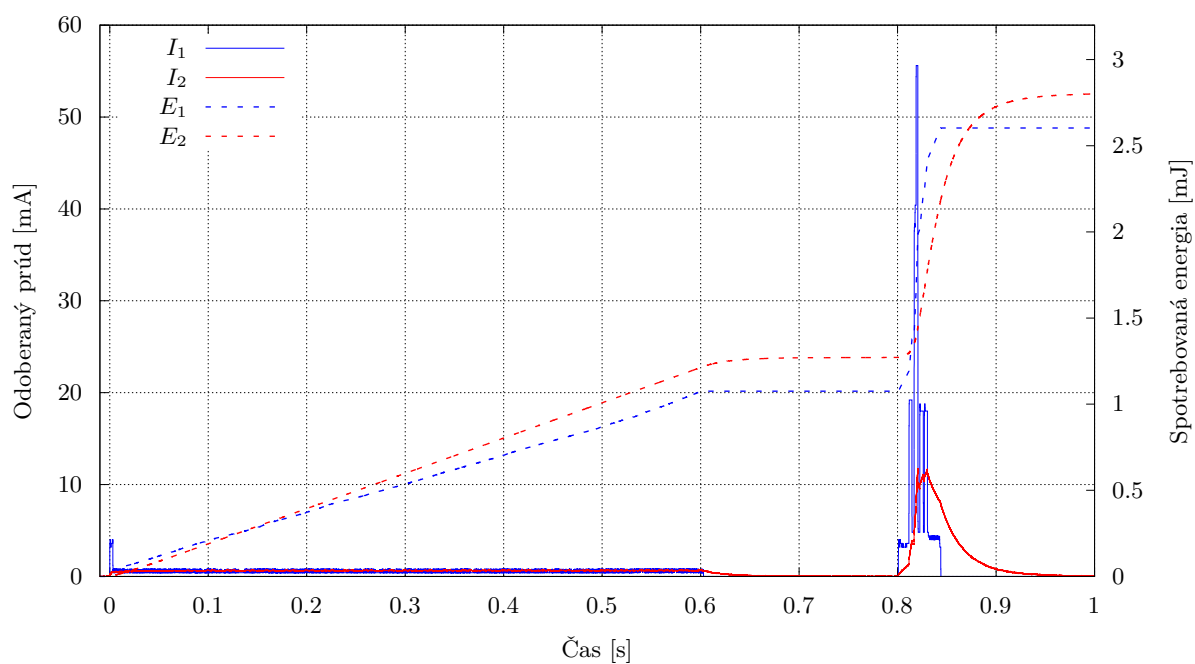
Pretože zariadenie musí byť schopné dlhodobej prevádzky na batériu, bolo nutné overiť jeho spotrebu meraním. Nakoľko zariadenie vydrží pracovať na batériu pomerne dlhú dobu, nebolo možné merať výdrž batérie priamo. Preto bolo pristúpené k výpočtu predpokladanej doby prevádzky zariadenia na základe nameraných údajov o spotrebe a údajov špecifikovaných výrobcou batérie [26].

Pre účely výpočtu výdrže napájacej batérie bol testovaný odber zariadenia a minimálne napätie, pri ktorom je zariadenie schopné pracovať. Bolo zistené, že zariadenie je schopné plnohodnotne pracovať aj pri napájacom napätí 2.4 V. Výrobca teplotného senzoru *DS18B20* však špecifikuje, že minimálne napájacie napätie tohto senzoru musí byť aspoň 3 V. Pri výpočte výdrže batérie bola teda použitá táto hodnota. Meranie odoberaného prúdu prebehlo pri napájacom napätí 3.2 V. Nameraná hodnota odberu prúdu v režime spánku je 1.9 μ A.

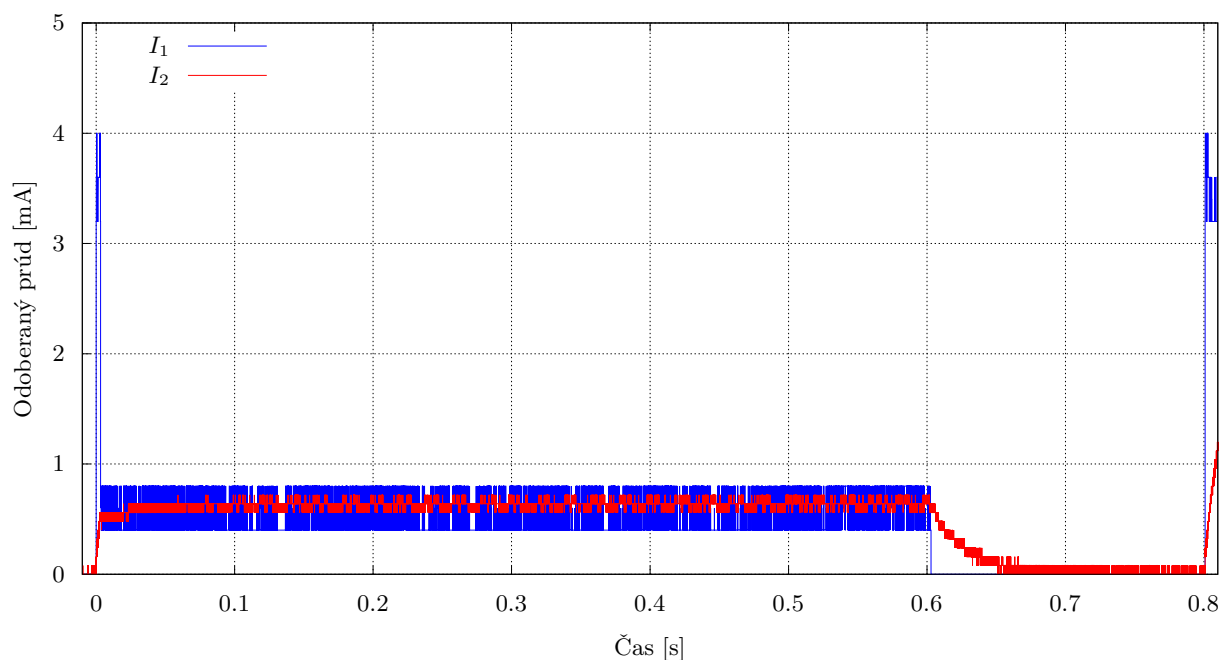
Odber prúdu počas jedného merania teploty a komunikácie s riadiacim počítačom je znázornený modrou farbou na obrázku 6.2 (I_1). Energia spotrebovaná pri tejto operácii je zobrazená ako E_1 . Na jedno meranie je spotrebovaných približne 2.6 mJ energie.

Detail odberu prúdu pri meraní teploty je znázornený modrou farbou na obrázku 6.3 (I_1). Tu môžeme vidieť v čase 0 s nárast prúdu pri zobudení mikrokontroléra. Mikrokontrolér následne prikáže teplotnému senzoru začať meranie teploty a uspeje sa. Z obrázku je zrejmé, že meranie pokračuje ďalších 0.6 s, potom sa odber znižuje. Nakoľko však výrobca teplotného senzoru požaduje, aby bol čas na meranie teploty aspoň 0.75 s, mikrokontrolér spí až do času 0.8 s, kde môžeme vidieť nárast prúdu po jeho zobudení.

Detail odberu prúdu pri komunikácii je znázornený modrou farbou na obrázku 6.4 (I_1). Podľa veľkosti odoberaného prúdu sa dá konštatovať, že komunikačný modul bol niekoľkokrát prepnutý z režimu spánku do režimu príjmu, prípadne do režimu vysielania. Prúd pri vysielaní dosahuje maxima až približne 56 mA. To je v súvislosti s použi-

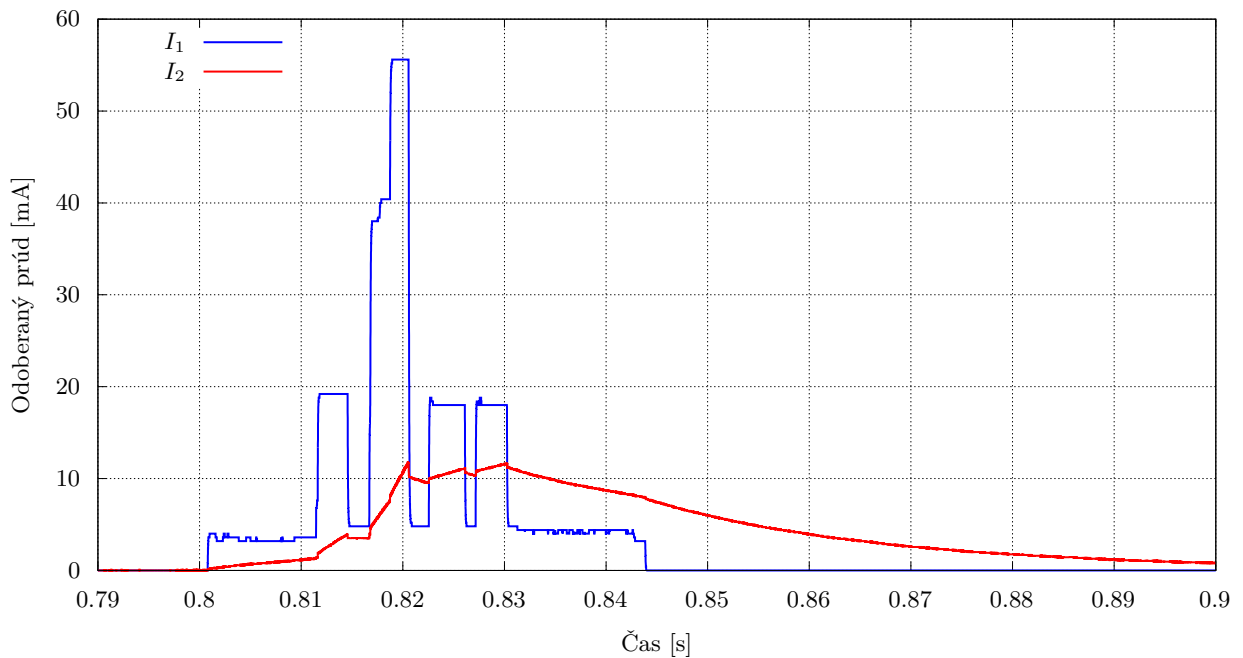


Obr. 6.2: Spotreba energie bezdrôtového teplotného senzora



Obr. 6.3: Detail spotreby energie bezdrôtového teplotného senzora pri meraní teploty

tou batériou nezanedbateľná hodnota. Výrobca odporúča zafažovať batériu stálym prúdom nepresahujúcim 25 mA, ale pripúšťa pulzné zafažovanie do 150 mA po dobu 100 ms v časových intervaloch o dĺžke minimálne 2 min. Ak by zariadenie vysielalo teplotu v intervale 2 min alebo dlhšom, nedošlo by k prekročeniu hodnôt odporúčaných výrobcom [26].



Obr. 6.4: Detail spotreby energie bezdrôtového teplotného senzora pri komunikácii

Pretože však vnútorný odpor batérie použitého typu dosahuje už pri polovičnom vybití približne $30\ \Omega$, pri odbere $56\ \text{mA}$ by mohlo klesnúť napájacie napätie až o približne $1.68\ \text{V}$, čo by v konečnom dôsledku drasticky redukovalo množstvo využiteľnej energie z batérie. Už pri polovičnom vybití by pri zaťažení dosahovalo napätie batérie len približne $1.92\ \text{V}$, čo na napájanie zariadenia nestačí. Výpočet je znázornený rovnicou 6.1, kde U_b je napájacie napätie, U_{bn} je nominálne napätie batérie a R_b je vnútorný odpor batérie pri polovičnom vybití.

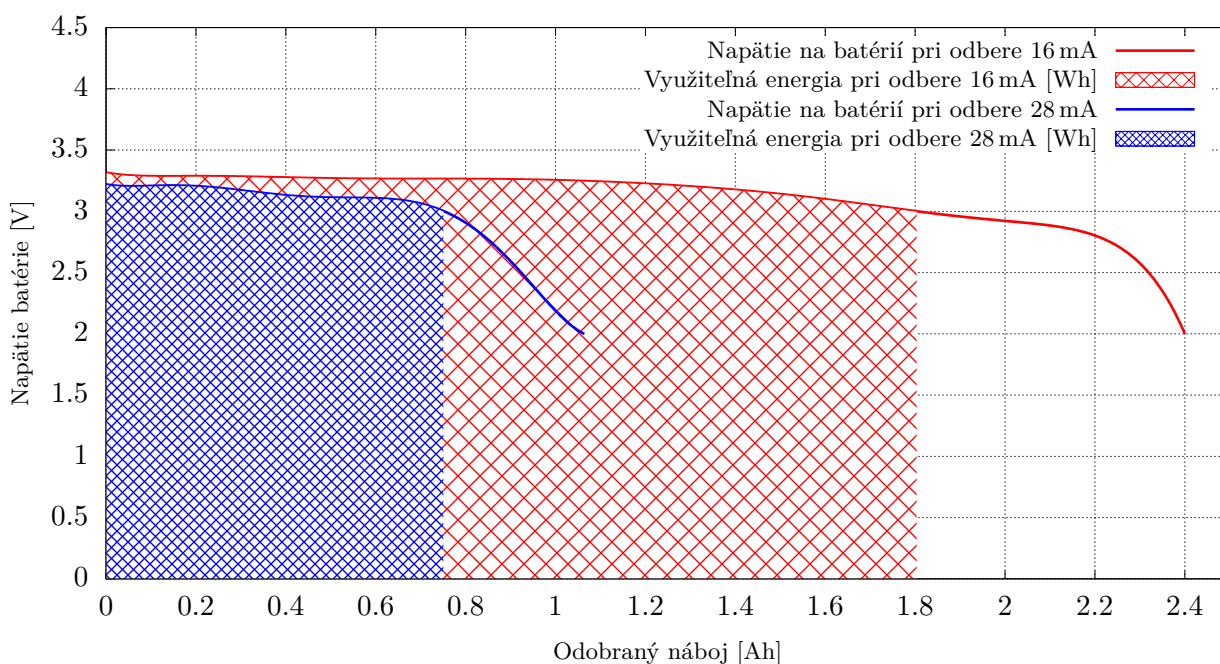
$$\begin{aligned}
 U_b &= U_{bn} - I_{1max} \cdot R_b [\text{V}] \\
 U_b &\doteq 3.6 - 0.056 \cdot 30\text{V} \\
 U_b &\doteq 1.92\text{V}
 \end{aligned}
 \tag{6.1}$$

Z obrázkov 6.2 a 6.4 je však zrejmé, že špičkový odber netrvá dlhšie ako $5\ \text{ms}$. Pre účely zmenšenia tohto špičkového odberu bolo zariadenie doplnené o kondenzátor s kapacitou $2\ \text{mF}$, pripojený paralelne k batérii. Po pripojení tohto kondenzátora klesol špičkový prúd na približne $12\ \text{mA}$. Priebeh prúdu (I_2) a spotrebovaná energia (E_2) po pripojení kondenzátora sú znázornené na obrázkoch 6.2, 6.3 a 6.4 červenou farbou. Množstvo spotrebovanej energie je v tomto prípade asi o 8% vyššie, až $2.82\ \text{mJ}$, ako bez použitia kondenzátora. Toto sa dá prisúdiť stratám na kondenzátore. Pri polovičnom vybití batérie by jej napätie pri zaťažení dosahovalo približne $3.24\ \text{V}$, ako je ukázané rovnicou 6.2.

Kapacita pripojeného kondenzátora bola zvolená empiricky. Použitím kondenzátora inej kapacity a lepších vlastností by bolo pravdepodobne možné množstvo spotrebovanej energie a veľkosť odoberaného prúdu ešte znížiť.

$$\begin{aligned}
 U_b &= U_{bn} - I_{2max} \cdot R_b [\text{V}] \\
 U_b &\doteq 3.6 - 0.012 \cdot 30\text{V} \\
 U_b &\doteq 3.24\text{V}
 \end{aligned}
 \tag{6.2}$$

Výrobca batérie neudáva vybijaciu charakteristiku článku pri prúde vyššom ako 28 mA [26]. Z dostupných údajov však je možné konštatovať, že pri nižšom prúdovom zaťažení je možné z batérie odobrať vyššie množstvo energie. Na obrázku 6.5 sú zobrazené vybijacie charakteristiky batérie pri odbere 28 mA a 16 mA udávané výrobcom. Pre znázornenie boli vybrané charakteristiky najbližšie ku v prototype nameraným hodnotám prúdu (56 mA a 12 mA). Tieto charakteristiky boli doplnené o množstvo zariadením využiteľnej energie, zobrazenej ako šrafovaná plocha pod krivkou. Počíta sa s vybitím na úroveň 3 V. Pri zaťažení prúdom 28 mA by teda zariadenie mohlo využiť len približne 11.4 kJ (3.167 Wh) energie, no pri zaťažení prúdom 16 mA až približne 26.9 kJ (7.47 Wh) energie. Dá sa predpokladať, že pri prúdovom zaťažení 56 mA by množstvo využiteľnej energie ešte drasticky kleslo (oproti znázornenému odberu 28 mA), zatiaľ čo pri prúdovom zaťažení 12 mA bude toto množstvo vyššie ako pri výrobcom špecifikovanom odbere 16 mA.



Obr. 6.5: Využitelná energia batérie bezdrôtového teplotného senzora

Predpokladajme teda, že zariadenie bude merať teplotu každých 5 min a bude vybavené spomínaným kondenzátorom. Za hodinu bude zariadenie merať teplotu 12 krát a teda v aktívnom režime (pri meraní a komunikácii) sa spotrebuje 33.84 mJ energie. Výpočet je znázornený rovnicou 6.3, kde E_2 je energia potrebná na jedno meranie a komunikáciu.

$$\begin{aligned}
 E_{rh} &= E_2 \cdot 12 [J] \\
 E_{rh} &\doteq 2.82 \times 10^{-3} \cdot 12 J \\
 E_{rh} &\doteq 3.384 \times 10^{-2} J
 \end{aligned} \tag{6.3}$$

Každé meranie a komunikácia trvá približne 0.845 s, v aktívnom režime teda strávi každú hodinu 10.14 s. Na chod v režime spánku, ktorý bude z hodiny trvať 3589.86 s sa spotrebuje 218.26 mJ energie. Výpočet je znázornený rovnicou 6.4, kde I_s je odber v režime spánku a t_r je čas strávený v aktívnom režime.

$$\begin{aligned}
 E_{sh} &= U \cdot I_s \cdot (3600 - 12 \cdot t_r) [J] \\
 E_{sh} &\doteq 3.2 \cdot 1.9 \times 10^{-6} \cdot (3600 - 12 \cdot 8.45 \times 10^{-1}) J \\
 E_{sh} &\doteq 2.1826 \times 10^{-1} J
 \end{aligned} \tag{6.4}$$

Ako je znázornené na obrázku 6.5, z batérie je možné využiť približne 7.47 W h, čo je rovné 26.9 kJ, energie. Do vybitia batérie na úroveň 3 V teda zariadenie bude bežať približne 106 700 h alebo 4446 d, ako je demonštrované rovnicou 6.5, kde t_{tot} je čas do dosiahnutia napätia 3 V, E_b využiteľné množstvo energie z batérie, E_{sh} energia spotrebovaná za hodinu prevádzky pripadajúca na spánok a E_{rh} energia spotrebovaná za hodinu prevádzky pripadajúca na aktívny režim.

$$\begin{aligned}
 t_{tot} &= \frac{E_b}{E_{sh} + E_{rh}} [h] \\
 t_{tot} &\doteq \frac{2.69 \times 10^4}{2.1826 \times 10^{-1} + 3.384 \times 10^{-2}} h \\
 t_{tot} &\doteq \frac{2.69 \times 10^4}{2.521 \times 10^{-1}} h \\
 t_{tot} &\doteq 1.067 \times 10^5 h \\
 t_{tot} &\doteq 4446 d
 \end{aligned} \tag{6.5}$$

Tento výpočet je však len orientačný, neberie do úvahy meniaci sa prúd so znižujúcim sa napätím batérie. Tiež neberie do úvahy to, že teplotný senzor, ktorý je na hladinu napájacieho napätia najcitlivejší, nikdy nebeží súčasne s vysielateľom. Odber prúdu pri samotnom meraní teploty je o viac ako rád nižší ako pri vysielaní a teda zariadenie pravdepodobne bude fungovať dlhšie ako bolo vypočítané. Aj pri zanedbaní tejto skutočnosti je však vypočítaná výdrž batérie, dosahujúca viac ako 12 rokov, dostatočná.

Kapitola 7

Riadiaci počítač

Ako už bolo spomenuté v kapitole 3, pre účely riadenia systému bol zvolený jednodoskový počítač *Raspberry Pi 2*. K jeho výberu viedla jeho dobrá dostupnosť, veľká rozšírenosť v komunite, a teda aj veľké množstvo otvoreného softvéru pre tento počítač. V neposlednom rade prispela k jeho výberu tohto počítača existencia už odladených doplnkových modulov, ako napríklad dotykový displej, jeho hardvérové vybavenie a dokumentácia.

Medzi hlavné nevýhody tohto počítača, oproti iným jednodoskovým počítačom, patrí absencia vstavaného perzistentného pamäťového média, prípadne rozhrania SATA. Toto prináša nutnosť použiť ako perzistentné pamäťové médium SD kartu. Dnes sa však už vyrábajú aj odolné (čo sa týka počtu zápisov) SD karty, použiteľné ako systémový disk. V prípade potreby by bolo možné počítač rozšíriť o externý disk pripojený pomocou rozhrania USB. *Raspberry Pi 2* však podporuje len boot z SD karty, teda minimálne pre účely štartovania systému musí byť karta prítomná.

7.1 Hardvér

K *Raspberry Pi 2* je na trhu dostupný ako prídavný modul aj kompatibilný dotykový displej s uhlopriečkou 7 palcov a rozlíšením 800×480 bodov. Tento prídavný modul obsahuje aj riadiacu elektroniku pre displej samotný a dotykový kontrolér. Dátové spojenie medzi riadiacim počítačom a dotykovým displejom je zabezpečené pomocou jediného plochého flexibilného kábla. Modul displeja je konštruovaný tak, aby ho bolo možné s počítačom mechanicky spojiť. Obe zariadenia zdieľajú jednotné napájacie napätie 5 V.

Pre účely doplnenia počítača o potrebné rozhrania je k nemu pripojený pomocou 40pólového plochého kábla rozširujúci modul opísaný v kapitole 4. Rozširujúci modul zdieľa spolu s počítačom a displejom jednotné napájacie napätie 5 V, poskytované spínaným meničom 230 V AC/5 V DC.

7.2 Softvér

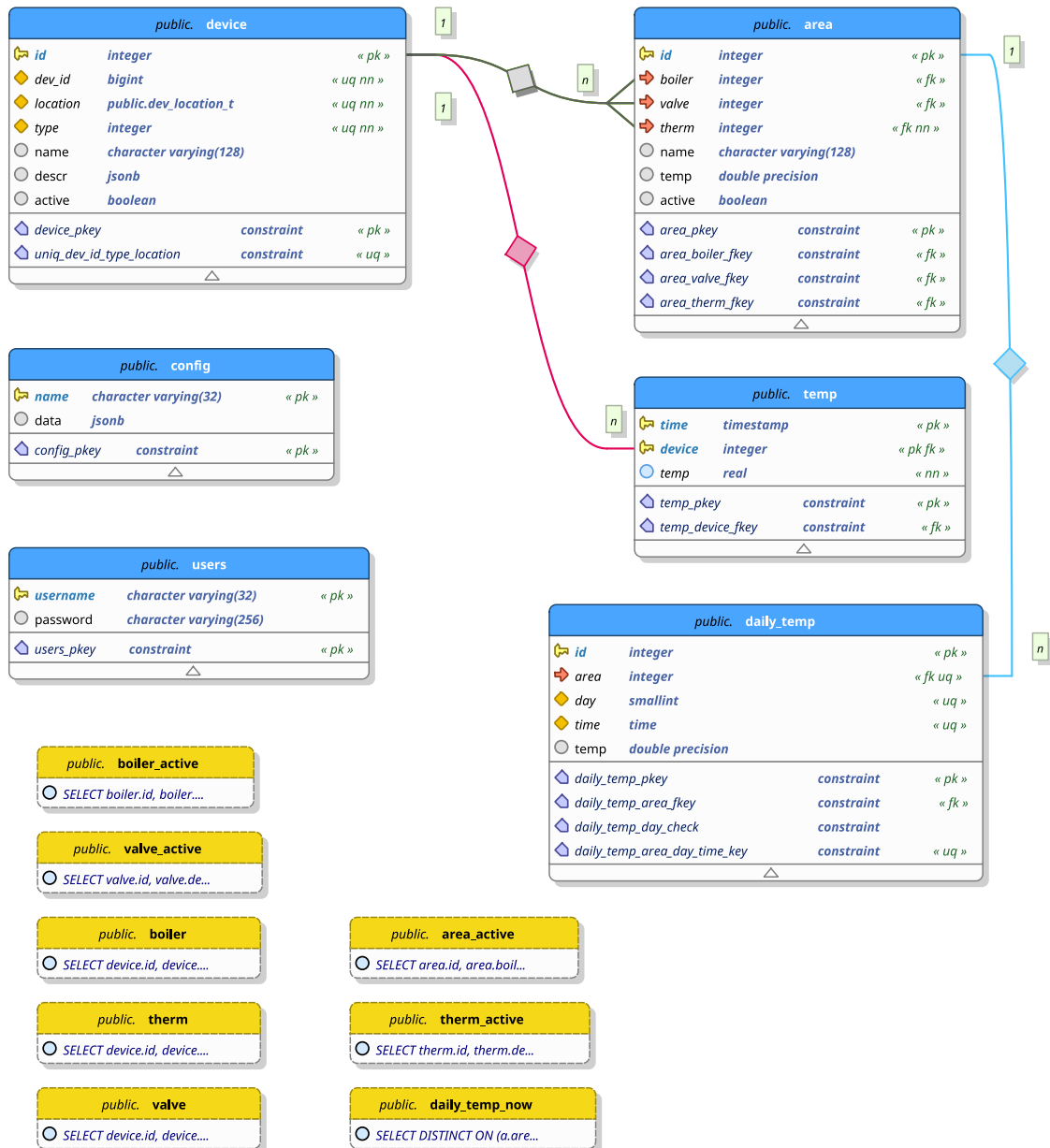
Riadiaci počítač používa operačný systém *Raspbian Stretch*, ktorý je optimalizovaný pre počítač *Raspberry Pi 2*. Prípadné použitie iného operačného systému by prinášalo nutnosť zostavovania vlastného jadra, a dá sa predpokladať, že by bolo potrebné riešiť značné množstvo problémov na rôznych úrovniach.

V kapitole 3 je popísaná zvolená softvérová architektúra, znázornená na obrázku 3.2, ktorá zahŕňa: systém riadenia bázy dát, centrálny kontrolér, CAN kontrolér, bezdrôtový

kontrolér, 1-Wire kontrolér, I²C™ kontrolér, web server pre REST API, hlavný web server a webový prehliadač.

7.2.1 Systém riadenia bázy dát

Pre účely uchovávanía stavových informácií systému a nameraných dát bol použitý databázový systém *PostgreSQL*. *PostgreSQL* je robustným systémom riadenia bázy dát. Toto riešenie umožňuje zdieľať dáta medzi komponentami systému, zabezpečuje perzistenciu dát a umožňuje rýchle vyhľadávanie.



Obr. 7.1: Schéma navrhutej a implementovanej databázy

Navrhnutá schéma databázy je znázornená na obrázku 7.1. Všetky pripojené zariadenia, teda teplomery, ventily a kotly sú uložené v tabuľke *device*. V danej tabuľke sú uložené aj tieto informácie: typ zariadenia, názov zariadenia, lokalita (na ktorej zbernici sa nachádza), identifikátor zariadenia na danej zbernici a informácia o tom, či je zariadenie aktivované. Tabuľka tiež obsahuje stĺpec *descr*, v ktorom môžu byť vo formáte JSON uložené doplnkové informácie špecifické pre zariadenie. Pre účely jednoduchého výberu zariadení príslušného typu (teplomer, ventil, ...) boli vytvorené databázové pohľady ako *boiler*, *valve* a podobne.

Tabuľka *area* obsahuje informácie o zónach, v ktorých je možné riadiť vykurovanie. Každá zóna musí obsahovať teplomer a môže obsahovať kotol alebo ventil. Zóna môže byť aktívna alebo neaktívna. Databáza obsahuje také integritné obmedzenia, aby nebolo možné ako teplomer zvoliť napríklad kotol a podobne. Každá zóna môže mať prednastavenú teplotu.

Tabuľka *temp* obsahuje namerané teploty. Každý záznam je pri vytvorení zaopatrený časovou pečiatkou a identifikátorom teplomera, ktorý danú hodnotu namerá.

Pre účely regulácie vykurovania na základe dennej doby bola vytvorená tabuľka *daily_temp*. Do tejto tabuľky je možné uložiť pre každú zónu požadovanú teplotu pre daný deň v týždni a dennú dobu. Pomocou databázového pohľadu *daily_temp_now* je potom možné zobrazíť informácie o požadovanej teplote v daných zónach v čase databázového dotazu.

Rôzne konfiguračné údaje spojené so systémom vykurovania sú uložené v tabuľke *config*. Prístupové údaje sú uložené v tabuľke *users*.

7.2.2 Kontroléry jednotlivých rozhraní

Ku každému používanému rozhraniu bol vytvorený kontrolér (viď obrázok 3.2), teda oddelený program, ktorý zapúzdruje implementačné detaily špecifické pre dané rozhranie. S pripojenými zariadeniami je preto možné komunikovať jednotným spôsobom. V rámci tejto práce boli implementované metódy pre vyhľadanie zariadení na zberniciach, čítanie teploty, nastavovanie ventilov a komunikáciu s kotlom.

Kontroléry jednotlivých zariadení boli implementované v jazyku *Python 3*, no môžu byť implementované v prakticky akomkoľvek jazyku. Programy boli implementované ako viacvláknové, kde je jedno vlákno vyhradené pre D-Bus a ostatné vlákna sa zaoberajú samotným riadením zariadení na zbernici.

7.2.3 Centrálny kontrolér

Centrálny kontrolér je program, ktorý je zodpovedný za zhromažďovanie informácií o teplote v jednotlivých zónach a za rozhodovanie o vykonaných akciách. Centrálny kontrolér získava konfiguračné informácie potrebné k svojej činnosti z už spomínaného systému riadenia bázy dát.

Medzi týmito informáciami sa nachádza aj časový interval regulácie. Raz za špecifikovaný interval sa kontrolér zobudí, získa aktuálne hodnoty, rozhodne o potrebných akciách a vykoná ich. Po vykonaní všetkých akcií sa kontrolér na daný časový interval uspí.

V prípade, že nie je možné nadviazať spojenie s niektorým zariadením potrebným pre vyhodnotenie situácie, prípadne nie je možné riadenie vykurovania v niektorej zo zón, vykurovanie v tejto zóne neprebíha. Nemôže sa teda stať, že by pri poruche ventilu, ktorý zostal v uzavretej polohe, bol spustený kotol, zatiaľ čo by vykurovacie teleso v danej zóne zostalo vypnuté, čo by malo za následok zbytočné zvýšenie nákladov.

V súčasnej dobe nebolo implementované žiadne inteligentné riadenie vykurovania. Kontrolér teda zatiaľ nie je schopný učiť sa, nepodporuje automatické riadenie teploty vykurovacieho média v závislosti na vonkajšej teplote a podobne. O tieto schopnosti je však možné systém rozšíriť v budúcnosti.

7.2.4 Webový server pre REST API

Pre účely ovládania systému užívateľom bol implementovaný web server poskytujúci REST API. Pomocou tohto API je možné nastavovať rôzne parametre systému, pridávať a odberať zariadenia a zóny. Tiež umožňuje získať zo systému štatistické informácie.

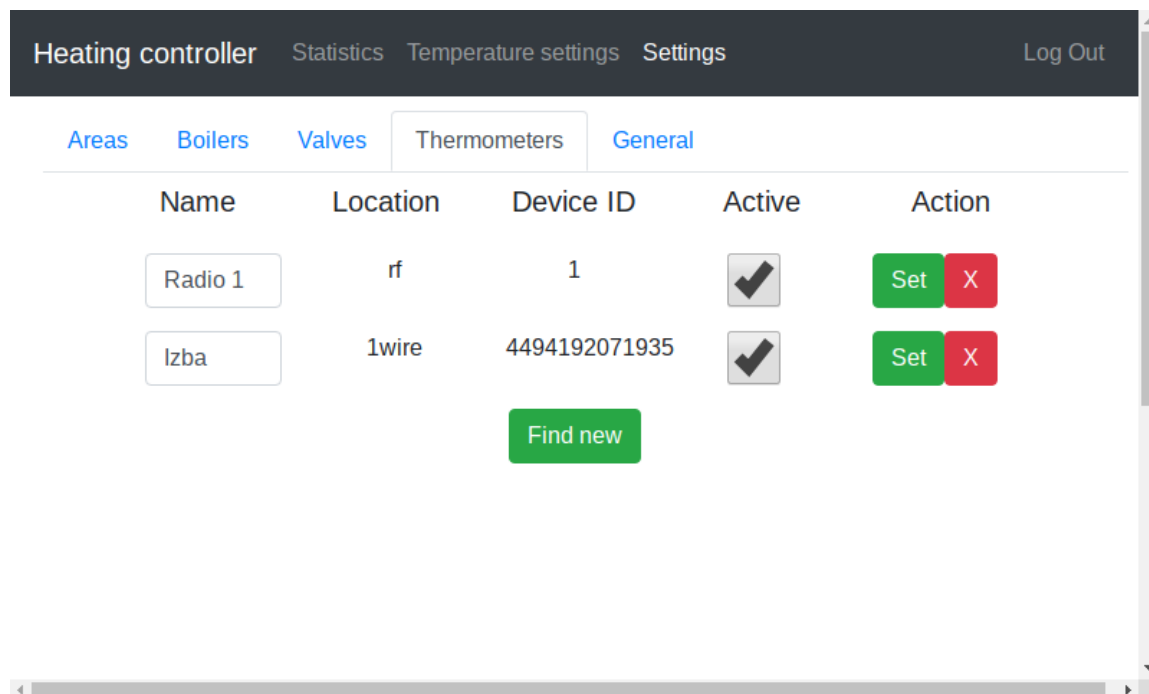
Nakoľko sa počíta s tým, že toto API bude vystavené na internete, bola implementovaná autentizácia. Pre autentizáciu rozhrania boli použité tokeny typu JWT. To znižuje počet nutných databázových dotazov, pretože nie je nutné token overovať oproti tokenu uloženému v databáze.

Web server bol implementovaný v jazyku *Python 3* s použitím existujúcej knižnice *tornado*.

7.2.5 Grafické užívateľské rozhranie

Grafické užívateľské rozhranie je tvorené webovou aplikáciou bežiacou v bežnom internetovom prehliadači. Priamo na riadiacom počítači beží webový prehliadač v kiosk režime, ktorý umožňuje pohodlné ovládanie zariadenia. Tento prístup bol zvolený aj kvôli tomu, že umožňuje použiť jednu aplikáciu pre vzdialené aj lokálne ovládanie zariadenia.

Grafické užívateľské rozhranie bolo implementované v jazyku *Typescript* za pomoci frameworku *Angular 7* a štýlov *Bootstrap 4*. Bola vytvorená moderná single-page webová aplikácia využívajúca REST API.



Obr. 7.2: Užívateľské rozhranie – sekcia nastavovania teplomerov

Užívateľské rozhranie ponúka základné možnosti konfigurácie systému. Ak chce napríklad užívateľ pripojiť k systému nový teplomer, fyzicky ho pripojí na danú zbernicu, prípadne aktivuje bezdrôtové vysielanie, a následne klikne na tlačidlo *Find new* v zodpovedajúcej sekcii užívateľského rozhrania (viď obrázok 7.2). Ak sa podarí nový teplomer identifikovať, bude pridaný do databázy a zobrazený v užívateľskom rozhraní. Každý teplomer si môže užívateľ ľubovoľne pomenovať. Obdobne sa postupuje aj pri pridávaní iných zariadení.

Podobným spôsobom, ako je možné pridávať zariadenia, je možné pridávať aj vykurovacie zóny. Užívateľ si môže vytvoriť vykurovaciu zónu, ktorá musí obsahovať teplomer a môže obsahovať ventil a kotol. Ak si užívateľ zariadenia pomenoval, sú v ponuke zobrazené tieto názvy, inak je zobrazené logické meno zariadenia, jeho lokalita (rozhranie) a identifikátor. Snímok obrazovky zariadenia pri pridávaní zón je zobrazený na obrázku 7.3.

Name	Boiler	Valve	Thermometer	Active	Action
Chodba			Radio 1	<input checked="" type="checkbox"/>	Set X
Izba	Boiler 1	can:26843	Izba	<input checked="" type="checkbox"/>	Set X
Name				<input type="checkbox"/>	Create

Obr. 7.3: Užívateľské rozhranie – sekcia nastavovania vykurovacích zón

Zatiaľ sú k dispozícii dva režimy vykurovania: režim, ktorý udržiava v jednotlivých zónach prednastavenú konštantnú teplotu, a režim, ktorý riadi teplotu v zónach v závislosti od dňa v týždni a času. Ak nie je pre nejakú zónu nastavená žiadna teplota, použije sa teplota predvolená vo všeobecných nastaveniach. Užívateľské rozhranie, zobrazené na obrázku 7.4, umožňuje nastavenie teplotného priebehu pre každú zónu, deň v týždni a dennú dobu zvlášť.

Grafické užívateľské rozhranie ponúka tiež prehľad teplôt zaznamenaných za posledných 24 h v nastavených zónach. Ukážka tohto prehľadu je na obrázku 7.5. Do budúcnosti sa počíta s rozšírením možností zobrazovania štatistík o ďalšie parametre. Bolo by možné pridať informácie o tom, v ktorom čase bolo vykurovanie aktívne (kotol bol v prevádzke). Tiež by bolo možné pridať informácie spojené s prevádzkou plynového kotla. Pomocou protokolu OpenTherm™ je možné z kotla získať napríklad informácie o aktuálnej teplote vykurovacieho média, a to ako na výstupe tak aj na vstupe do kotla, a množstvo iných zaujímavých informácií.

Heating controller [Statistics](#) [Temperature settings](#) [Settings](#) [Log Out](#)

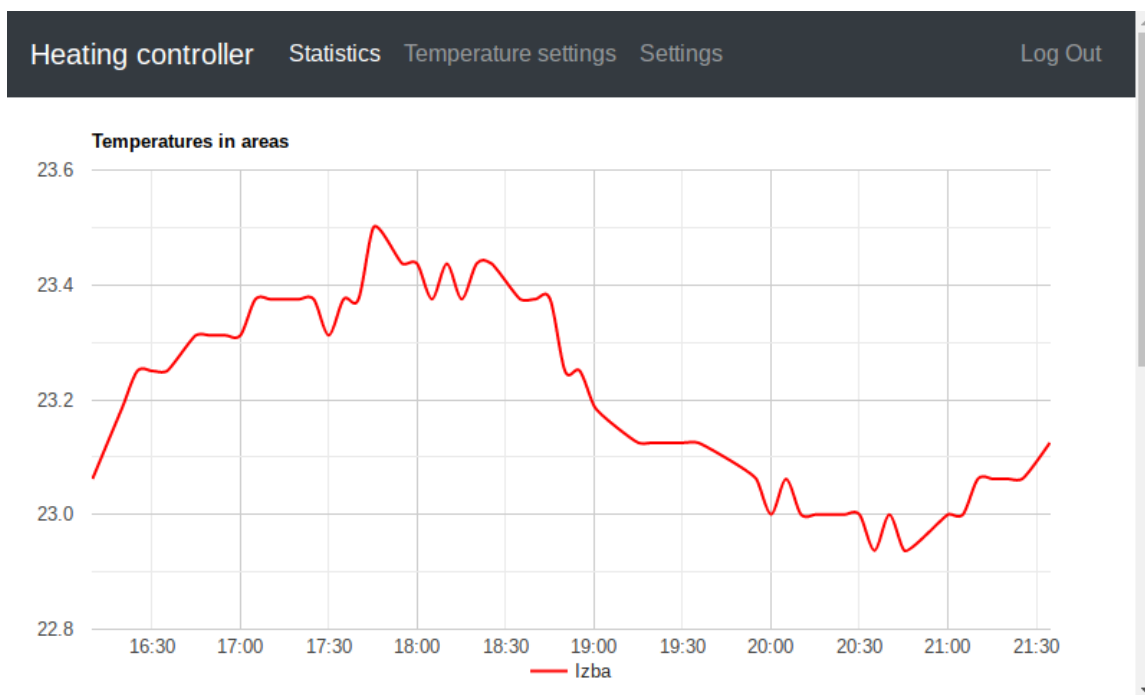
Mode [Constant temp](#) [Daily preset](#)

Izba [Monday](#)

Time	Temperature	Action
07:00:00	21	Delete
22:00:00	17	Delete

:

Obr. 7.4: Užívateľské rozhranie – sekcia nastavovania teploty vykurovacích zón



Obr. 7.5: Užívateľské rozhranie – prehľad nameraných teplôt podľa zóny

Implementácia

Užívateľské rozhranie na dotykovom displeji je zobrazené prostredníctvom webového prehliadača, v tomto prípade štandardného prehliadača *Chromium*. Pretože zariadenie nie je vy-

bavené hardvérovou klávesnicou, bolo potrebné nainštalovať vhodnú virtuálnu klávesnicu, ktorá sa zobrazí na obrazovke v prípade, že je potrebné zadať text. Medzi doplnkami pre prehliadač *Chromium* je k dispozícii niekoľko rôznych implementácií takýchto klávesníc, avšak ukázalo sa, že väčšina z nich nie je schopná správne pracovať s webovými stránkami implementovanými pomocou frameworku *Angular 7*. Bola zvolená vizuálne nie najlepšia virtuálna klávesnica *Browser Keyboard (Perfect for kiosk)*, ktorá je však funkčná. Bolo zvažované aj použitie virtuálnej klávesnice na úrovni operačného systému, no žiadna dostupná nebola vyhovujúca pre účely tejto práce.

Ako správca okien bol použitý program *Openbox*. *Openbox* je vysoko konfigurovateľný a rýchly. Práve jeho vysoká konfigurovateľnosť umožnila jednoducho odstrániť rôzne nepotrebné lišty a doplnky, čo umožnilo efektívne využiť relatívne malú plochu dotykového displeja.

7.2.6 Hlavný webový server

Na účely sprístupnenia webového užívateľského rozhrania a REST API bol použitý robustný webový server *nginx*. Tento server na základe URL adresy poskytuje buď statický obsah alebo funguje ako proxy server a presmeruje dotaz na webový server pre REST API.

Toto riešenie znižuje šance útočníka na preniknutie do systému, nakoľko *nginx* je široko používaný a záplatovaný softvér a je menej náchylný na prienik ako server implementovaný pomocou knižnice *tornado*.

7.2.7 Prepojenie s centrálnym serverom

Na pripojenie sa do siete VPN pre účely vzdialeného ovládania bol nakonfigurovaný klient pre *OpenVPN*. Tiež bolo zabezpečené oznamovanie aktuálnej IP adresy DDNS serveru bežiacemu na centrálnom serveri.

7.2.8 Spúšťanie programov

O spúšťanie programov, spomenutých v predchádzajúcich častiach, a ich beh na pozadí sa stará *systemd*. Všetky implementované programy sú spúšťané s právami bežného užívateľa. Tým je redukované riziko napadnutia systému.

Kapitola 8

Centrálny server

Ako centrálny server, ktorý umožňuje vzdialené pripojenie k vykurovaciemu systému, bol použitý komerčne dostupný virtuálny server.

8.1 Softvér

8.1.1 Operačný systém

Na serveri je nainštalovaný operačný systém *CentOS 7* [5]. Jedná sa o robustný operačný systém s dlhodobou podporou, vychádzajúci z komerčného operačného systému *Red Hat Enterprise Linux 7*.

8.1.2 Vzdialený prístup

Na serveri bol nainštalovaný *OpenVPN* server, ku ktorému sa môžu pripájať jednotlivé inštancie vykurovacích systémov. Tým bol zabezpečený prístup k systémom, ktoré nemajú verejnú IP adresu, prípadne sú ukryté za NATom. VPN bola nakonfigurovaná tak, že neumožňuje klientom (jednotlivým inštanciam vykurovacieho systému) komunikáciu iní ako s týmto serverom. Je zakázaná aj komunikácia medzi jednotlivými klientami. S každým klientom je teda vytvorená akási virtuálna point-to-point linka.

Na centrálnom serveri bol nakonfigurovaný aj dynamický DNS server. DNS server obsluhuje doménu, pomocou ktorej sa dá z internetu pripojiť na jednotlivé vykurovacie systémy. Pre každého pripojeného klienta (inštanciu vykurovacieho systému) sú vytvorené dve subdomény: jedna je určená pre použitie z internetu a ukazuje na centrálny server, druhá subdoména je určená pre interné potreby tohto servera a ukazuje na adresu klienta v sieti VPN. Pre účely dynamického obnovovania doménových záznamov bol v rámci tejto práce implementovaný program *ddnsUpdater*.

Pre potreby pripojenia k jednotlivým inštanciam vykurovacieho systému bol nakonfigurovaný webový server *nginx*. Server *nginx* funguje ako proxy server a dotazy, ktoré prišli na verejnú doménu, presmerováva na internú doménu, teda na špecifickú inštanciu vykurovacieho systému v rámci siete VPN.

Čo sa týka zabazpečenia, v dnešnej dobe je už možné zadarmo vystaviť wildcard certifikát na všetky subdomény tretieho rádu. Je teda možné komunikáciu medzi vzdialeným zariadením a týmto serverom aj bez problémov šifrovať. Prenos dát v sieti VPN medzi centrálnym serverom a inštanciami vykurovacieho systému je šifrovaný.

V súčasnej dobe je možné pridávať, respektíve generovať prihlasovacie údaje, do siete VPN len ručne. Tak isto je nutné ručne konfigurovať povolené doménové mená pri DDNS a virtuálne webové servery v konfigurácii *nginx*. Do budúcnosti sa počíta s automatizáciou týchto úkonov.

Kapitola 9

Záver

V rámci diplomovej práce bola navrhnutá architektúra automatického systému regulácie vykurovania, nasaditeľného v starších budovách s možnosťou vzdialeného prístupu.

Po hardvérovej stránke systém pozostáva z riadiacieho počítača *Raspberry Pi 2*, ktorý je pripojený cez internet k centrálnemu serveru, servomechanizmov ovládania radiátorových ventilov, plynového kotla *Baxi Prime 24*, káblových a bezkáblových teplotných senzorov. Riadiaci počítač je doplnený o rozširujúci modul, ktorý umožňuje komunikáciu s pripojenými zariadeniami. Súčasťou systému je aj dotykový displej pre pohodlné užívateľské ovládanie.

Pre účely práce boli navrhnuté a vyrobené funkčné prototypy rozširujúceho modulu k *Raspberry Pi 2*, servomechanizmu ovládania ventilu radiátora a bezdrôtového teplotného senzora.

Rozširujúci modul k počítaču *Raspberry Pi 2* sa skladá z OpenTherm™ modulu, umožňujúceho komunikáciu s plynovým kotlom, bezdrôtového modulu pre komunikáciu s bezdrôtovými teplotnými senzormi a CAN modulu pre komunikáciu so zariadeniami pripojenými pomocou zbernice CAN, napríklad servomechanizmom ovládania ventilov radiátorov. Pre komunikáciu s kotlom boli v rámci tejto práce implementované knižnice *otherm* a *mi2c*.

Servomechanizmus ovládania ventilu pozostáva z dielov vyrobených na 3D tlačiarňi z termoplastu PETG, elektromotoru s prevodovkou, spínačov detekcie koncovej polohy, riadiacej elektroniky, pre ktorú bol navrhnutý a vyrobený plošný spoj, a ovládacích tlačidiel. Servomechanizmus je napájaný prostredníctvom vlastného adaptéra a komunikuje s riadiacim počítačom prostredníctvom zbernice CAN. Riadený je mikrokontrolérom *Atmel ATmega32m1*. Pre účely spínania motora bol použitý H mostík *TLE2502-2G*. Počítačové problémy s časovaním zbernice CAN boli vyriešené zapojením kryštálu, použitého ako zdroja presného hodinového signálu pre mikrokontrolér. Firmvér servomechanizmu vyžadoval mimo iné implementáciu protokolu na prenos správ po zbernici a mechanizmu na vyhľadávanie zariadení pripojených na zbernicu CAN.

V bezdrôtovom teplotnom senzore bol použitý mikrokontrolér *Atmel ATmega328*, komunikačný modul *RFM69HW* a teplotný senzor *DS18B20*. Ako zdroj energie bola použitá primárna batéria typu Li-SOCl₂ s menovitým napätím 3.6 V a kapacitou 2.7 A h. V prototypy bol použitý jednovrstvový plošný spoj, používajúci technológiu THT. Použitá existujúca knižnica pre ovládanie bezdrôtového modulu vyžadovala odstránenie radu chýb. Spotreba energie bola optimalizovaná a pri predpokladanej frekvencii meraní (raz za 5 minút) by napájacia batéria v bezdrôtovom module mala vydržať viac ako 12 rokov.

Softvérová architektúra riadiacieho počítača zahŕňa databázový server *PostgreSQL*, centrálny kontrolér a kontroléry pre každé z implementovaných komunikačných rozhraní, pre-

pojené pomocou softvérovej zbernice D-Bus, a webové servere. Pre účely nastavovania a zobrazovania údajov o systéme bol implementovaný webový server ponúkajúci REST API, a pre prístupnosť webovej aplikácie užívateľského rozhrania bol použitý webový server *nginx*. O spúšťanie aplikácií sa stará *systemd*.

Centrálny server zabezpečuje vzdialený prístup a umožňuje riadenie a monitorovanie mnohých inštancií tohto systému. Ako jediný prvok musí disponovať verejnou IP adresou. Na centrálnom serveri, kde je nainštalovaný operačný systém *CentOS 7*, beží webový proxy server, *OpenVPN* server, program *ddnsUpdater* a DNS server.

Pre každé z použitých hardvérových rozhraní bol implementovaný kontrolér, ktorý zapúzdruje špecifické vlastnosti daného rozhrania. Ku každému rozhraniu je teda možné prostredníctvom jeho kontroléra pristupovať jednotným spôsobom. Boli implementované kontroléry jednotlivých hardvérových rozhraní, najmä metódy pre vyhľadanie zariadení na zberniciach, čítanie teploty, nastavovanie ventilov a komunikáciu s kotlom.

Centrálny kontrolér je zodpovedný za zhromažďovanie informácií o teplote v jednotlivých zónach a za rozhodovanie o vykonaných akciách. Konfiguračné informácie potrebné k svojej činnosti získava z databázového systému *PostgreSQL*.

Grafické užívateľské rozhranie ponúka možnosti konfigurácie systému, ako je správa pripojených zariadení a vykurovacích zón, nastavovanie teplôt v zónach podľa dňa v týždni a dennej doby a prehľad štatistických informácií o vykurovanom systéme. Užívateľské rozhranie je dostupné lokálne na dotykovom displeji a vzdialene pomocou webovej aplikácie na internete.

Celý systém regulácie vykurovania bol úspešne otestovaný v reálnom prostredí trojizbového bytu v historickej budove a ukázal sa byť v daných podmienkach vyhovujúci. Vzhľadom na použitie systému iba počas krátkej doby vykurovacej sezóny nebolo možné vyhodnotiť ekonomický prínos nasadeného zariadenia.

Systém je možné ďalej zdokonaľovať a rozširovať. Po hardvérovej stránke je možné systém relatívne jednoducho rozšíriť o ďalšie moduly, ako napríklad bezdrôtový modul pre ovládanie kotla, bezdrôtové servomechanizmy ovládania ventilov radiátorov, prípadne moduly ovládania osvetlenia či zásuviek. Po softvérovej stránke je možné systém rozšíriť napríklad o automatické riadenie teploty vykurovacieho média v závislosti na vonkajšej teplote, prípadne o schopnosť učiť sa a predvídať priebeh teplôt vo vykurovacích zónach, a na základe toho upravovať charakteristiky vykurovania, čo by mohlo viesť k ďalšiemu zvýšeniu efektivity.

Literatúra

- [1] Atmel Co.: *ATmega328P*. [Online; navštívené 21. mája 2019].
URL http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf
- [2] Atmel Co.: *ATmega8*. [Online; navštívené 21. mája 2019].
URL https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2486-8-bit-AVR-microcontroller-ATmega8_L_datasheet.pdf
- [3] Atmel Co.: *ATtiny24/44/82*. [Online; navštívené 21. mája 2019].
URL http://ww1.microchip.com/downloads/en/devicedoc/Atmel-7701_Automotive-Microcontrollers-ATtiny24-44-84_Datasheet.pdf
- [4] Atmel Co.: *ATmega16M1/ATmega32M1/ATmega64M1/ATmega32C1/ATmega64C1 Automotive*. 2015, [Online; navštívené 21. mája 2019].
URL http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7647-Automotive-Microcontrollers-ATmega16M1-32M1-64M1-32C1-64C1_datasheet.pdf
- [5] centos.org: *CentOS Product Specifications: End of Lifetime (EOL) Dates*. [Online; navštívené 21. mája 2019].
URL <https://wiki.centos.org/About/Product>
- [6] Chu, C.: *CAN-Bus with Raspberry Pi: HowTo/Quickstart MCP2515 Kernel 4.4.x+*. Január 2017, [Online; navštívené 21. mája 2019].
URL <https://vamtut0r.com/2017/01/17/can-bus-with-raspberry-pi-howtoquickstart-mcp2515-kernel-4-4-x/>
- [7] Cosmo Electronics Co.: *High Reliability Photo Coupler KP1020*. [Online; navštívené 21. mája 2019].
URL <https://www.gme.cz/data/attachments/dsh.523-055.1.pdf>
- [8] Diodes, Inc.: *AZ1117 – 1A Low Dropout Linear Regulator*. [Online; navštívené 21. mája 2019].
URL <http://www.mouser.com/ds/2/115/AZ1117-1139752.pdf>
- [9] Dobeš, J.; Žalud, V.: *Moderní radiotechnika*. BEN — technická literatura, 2006, ISBN 80-7300-132-2.
- [10] Hope Microelectronics Co., Ltd: *RFM69HW ISM Transceiver Module v 1.0*. [Online; navštívené 21. mája 2019].
URL <http://www.anarduino.com/docs/RFM69HW.pdf>

- [11] Infineon Technologies: *5A H-Bridge for DC-Motor Applications TLE5205-2*. 2007, [Online; navštívené 21. mája 2019].
URL <https://www.tme.eu/Document/4455e5e54ed2b6df320b566f3d093dd3/TLE5205-2G-DTE.pdf>
- [12] Internet Systems Consortium: *BIND9 Documentation*. [Online; navštívené 21. mája 2019].
URL <https://www.isc.org/downloads/bind/doc/>
- [13] Joan: *pigpio library*. [Online; navštívené 21. mája 2019].
URL <http://abyz.me.uk/rpi/pigpio/index.html>
- [14] Love, R.: *Get on the D-BUS*. *Linux Journal*, Január 2005, [Online; navštívené 21. mája 2019].
URL <https://www.linuxjournal.com/article/7744>
- [15] Maxim Integrated: *Daisy-Chaining SPI Devices*. [Online; navštívené 21. mája 2019].
URL <https://www.maximintegrated.com/en/app-notes/index.mvp/id/3947>
- [16] Maxim Integrated: *DS18B20 – Programmable Resolution 1-Wire Digital Thermometer*. [Online; navštívené 21. mája 2019].
URL <https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>
- [17] Maxim Integrated: *Guidelines for Reliable Long Line 1-Wire Networks*. [Online; navštívené 21. mája 2019].
URL <https://www.maximintegrated.com/en/app-notes/index.mvp/id/148>
- [18] Maxim Integrated: *Reading and Writing 1-Wire Devices Through Serial Interfaces*. [Online; navštívené 21. mája 2019].
URL <https://www.maximintegrated.com/en/app-notes/index.mvp/id/74>
- [19] Melnyk, I.: *Arduino OpenTherm Controller*. Máj 2017, [Online; navštívené 21. mája 2019].
URL http://ihormelnyk.com/arduino_opentherm_controller
- [20] Microchip Technology: *MCP2551 High-Speed CAN Transceiver*. 2007, [Online; navštívené 21. mája 2019].
URL <http://ww1.microchip.com/downloads/en/devicedoc/21667e.pdf>
- [21] Microchip Technology: *MCP2515 Stand-Alone CAN Controller with SPI Interface*. 2019, [Online; navštívené 21. mája 2019].
URL <http://ww1.microchip.com/downloads/en/DeviceDoc/MCP2515-Stand-Alone-CAN-Controller-with-SPI-20001801J.pdf>
- [22] Motorola, Inc.: *SPI Block Guide*. [Online; navštívené 21. mája 2019].
URL <https://www.nxp.com/docs/en/user-guide/UM10204.pdf>
- [23] NXP Semiconductors N.V.: *I2C-bus specification and user manual*. [Online; navštívené 21. mája 2019].
URL <https://www.nxp.com/docs/en/user-guide/UM10204.pdf>
- [24] Omron Co.: *G6E Low Signal Relay*. [Online; navštívené 21. mája 2019].
URL <https://www.tme.eu/Document/e36d216ae9e4d7e778281e7112a26c1e/G6E.PDF>

- [25] Raspberry Pi Foundation: *Raspberry Pi Documentation*. [Online; navštívené 21. mája 2019].
URL <https://www.raspberrypi.org/documentation/hardware/raspberrypi/>
- [26] Saft: *Primary lithium battery LS 14500C*. [Online; navštívené 21. mája 2019].
URL <https://www.gme.cz/data/attachments/dsh.542-028.1.pdf>
- [27] The OpenTherm Association: *OpenTherm™ Protocol Specification v2.2*. [Online; navštívené 21. mája 2019].
URL <https://www.domoticaforum.eu/uploaded/Ard%20M/OpenTherm%20Protocol%20v2-2.pdf>
- [28] Thorne, B.; Sandberg, C.: *python-can 2.1.0 documentation*. Február 2018, [Online; navštívené 21. mája 2019].
URL <https://python-can.readthedocs.io/en/2.1.0/>
- [29] Watterson, C.: *Controller Area Network (CAN) Implementation Guide*. Analog Devices, 2017, [Online; navštívené 21. mája 2019].
URL <https://www.analog.com/media/en/technical-documentation/application-notes/AN-1123.pdf>

Príloha A

Zoznam implementovaných správ protokolu OpenTherm™

Tabuľka A.1: Zoznam implementovaných správ protokolu OpenTherm™

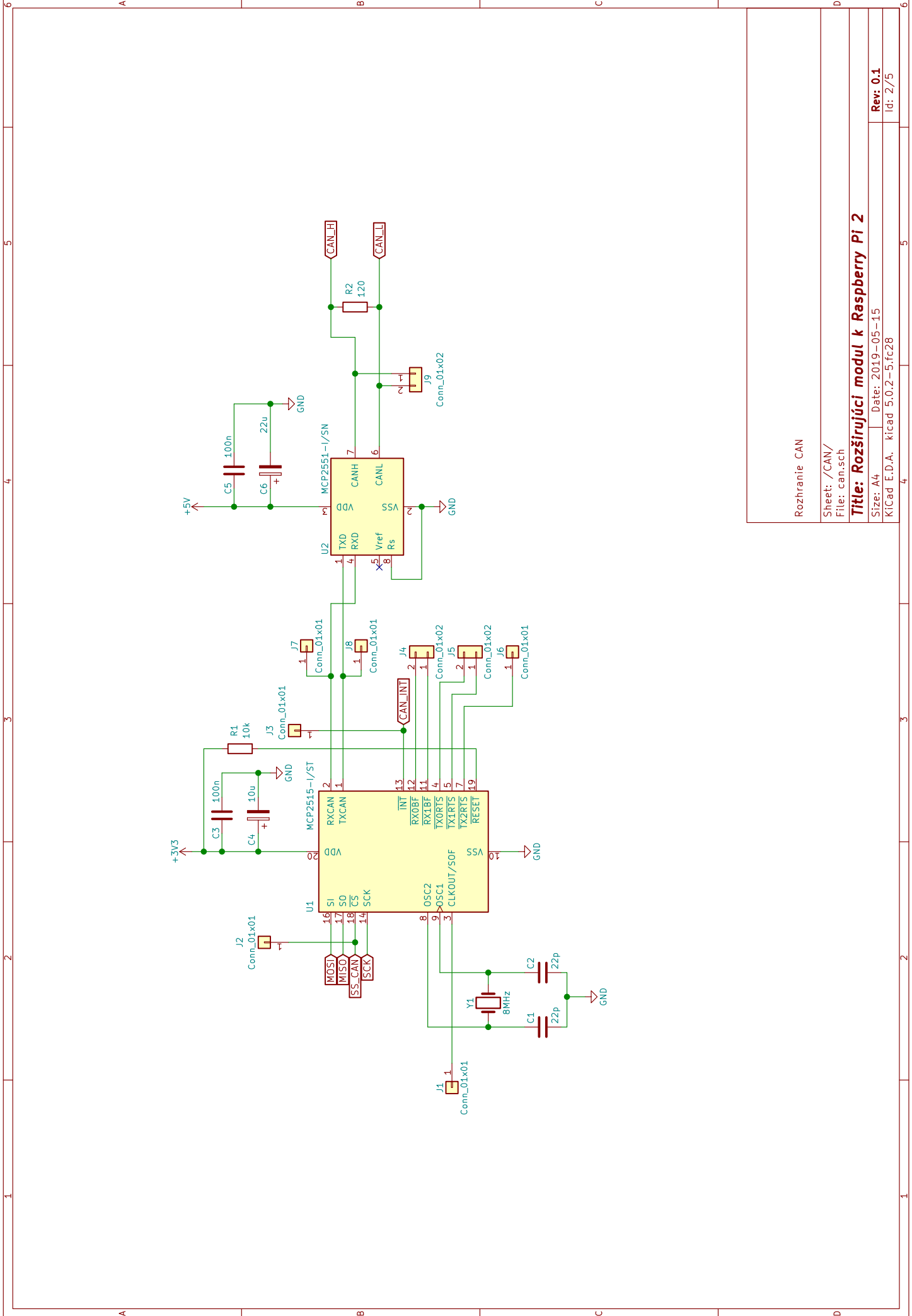
Data-ID	Trieda	Popis
0	MsgStatus	Master and Slave Status flags
1	MsgTSet	Control Setpoint ie CH water temperature setpoint
2	MsgMConfig	Master Configuration Flags / Master MemberID Code
3	MsgSConfig	Slave Configuration Flags / Slave MemberID Code
4	MsgCommand	Remote Command
5	MsgASFFlags	Application-specific fault flags and OEM-fault-code
6	MsgRBPFlags	Remote boiler parameter transfer-enable & read/write flags
17	MsgRelModLevel	Relative Modulation Level
25	MsgTboiler	Boiler flow water temperature
26	MsgTdhw	DHW temperature
28	MsgTret	Return water temperature
33	MsgTexhaust	Boiler exhaust temperature

48	MsgTdhwSetBounds	DHW setpoint upper & lower bounds for adjustment
49	MsgMaxTSetBounds	Max CH water setpoint upper & lower bounds for adjustment
56	MsgTdhwSet	DHW setpoint
57	MsgMaxTSet	Max CH water setpoint
116	MsgBurnerStarts	Number of burner starts
117	MsgCHPumpStarts	Number of CH pump starts
118	MsgDHWPumpStarts	Number of DHW pump starts
119	MsgDHWBurnerStarts	Number of DHW burner starts
120	MsgBurnerHours	Number of burner operation hours
121	MsgCHPumpHours	Number of CH pump operation hours
122	MsgDHWPumpHours	Number of DHW pump operation hours
123	MsgDHWBurnerHours	Number of DHW burner operation hours

Príloha B

Kompletá konštrukcia rozširujúceho modulu k *Raspberry Pi 2*

B.1 Schéma



Rozhranie CAN

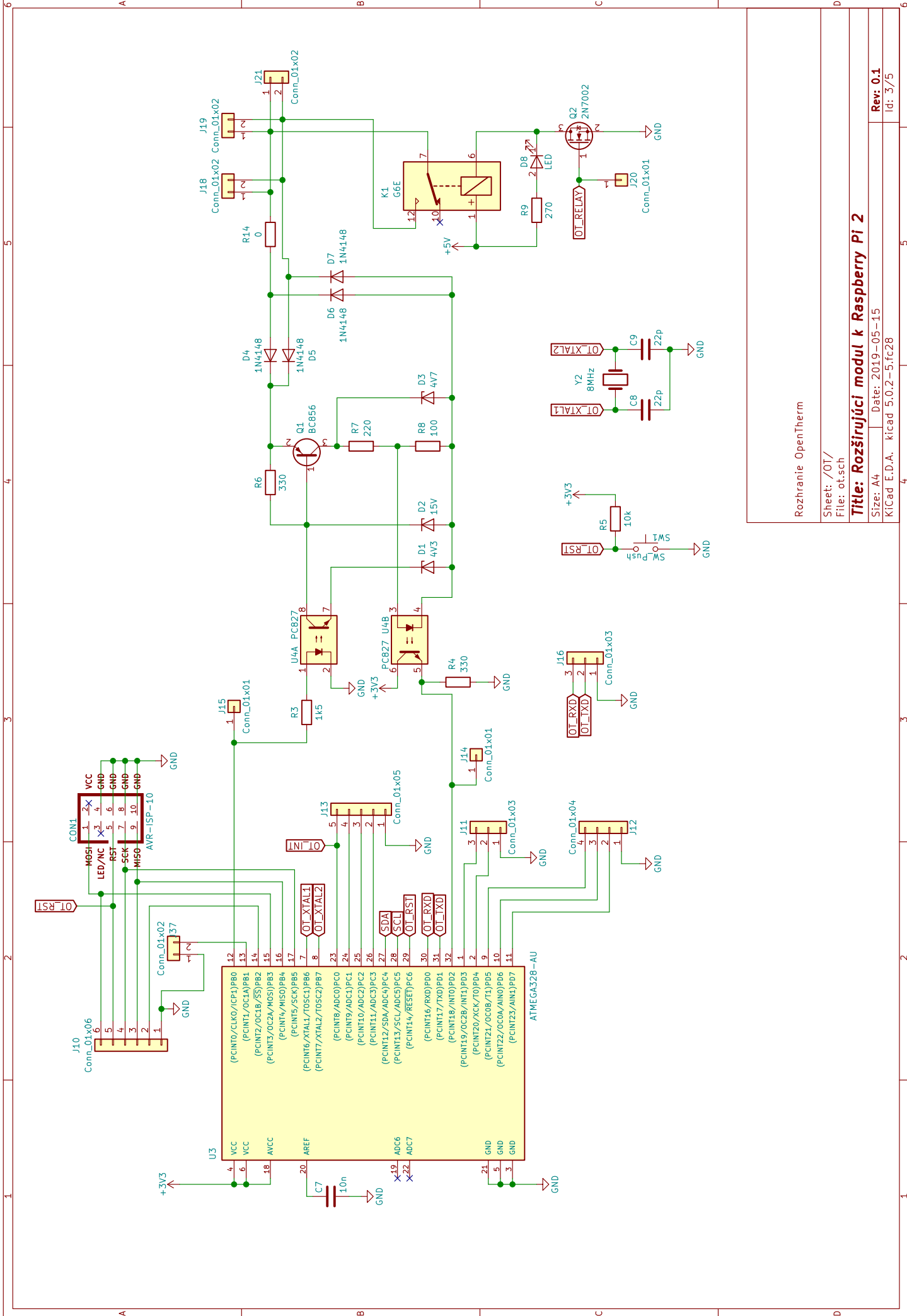
Sheet: /CAN/
File: can.sch

Title: Rozširujúci modul k Raspberry Pi 2

Size: A4
KiCad E.D.A. kicad 5.0.2-5.fc28

Date: 2019-05-15

Rev: 0.1
Id: 2/5



Rozhranie OpenTherm

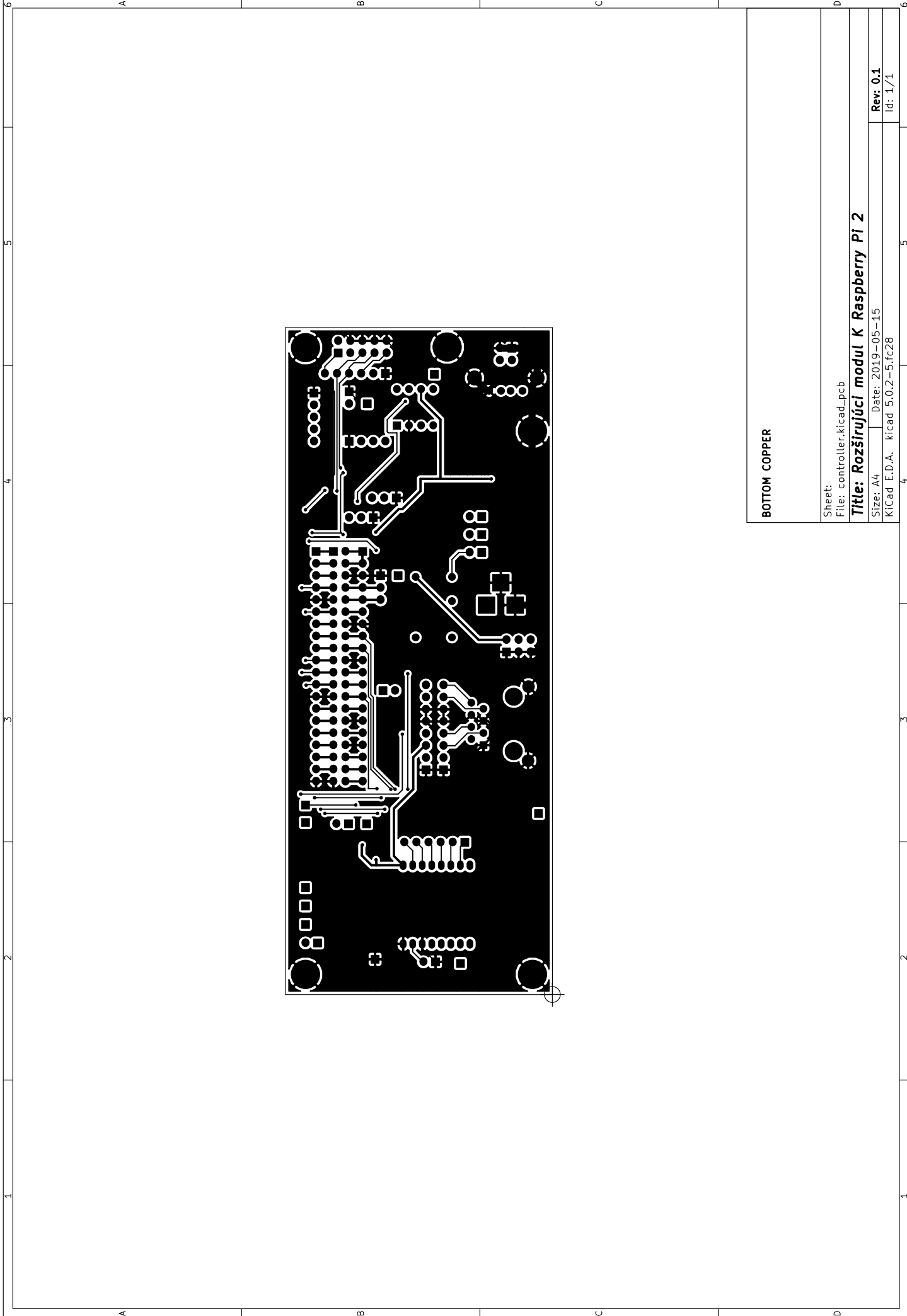
Sheet: /OT/
File: ot.sch

Title: Rozšiřujúci modul k Raspberry Pi 2

Size: A4 Date: 2019-05-15
KiCad E.D.A. kicad 5.0.2-5.fc28

Rev: 0.1
Id: 3/5

B.2 Plošný spoj



BOTTOM COPPER

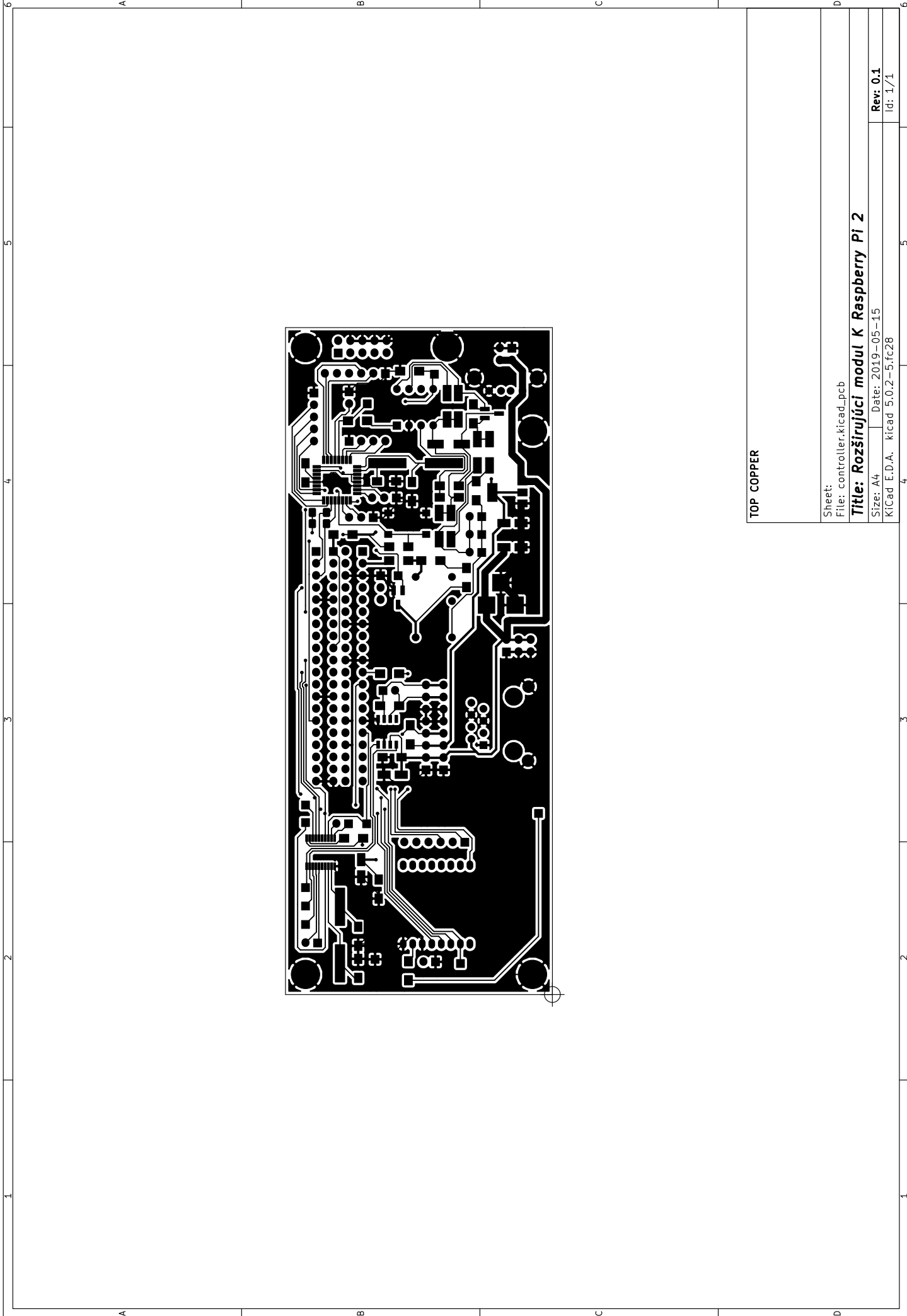
Sheet:
File: controller.kicad_pcb

Title: Rozšířující modul K Raspberry Pi 2

Size: A4 Date: 2019-05-15

Rev: 0.1

Id: 1/1

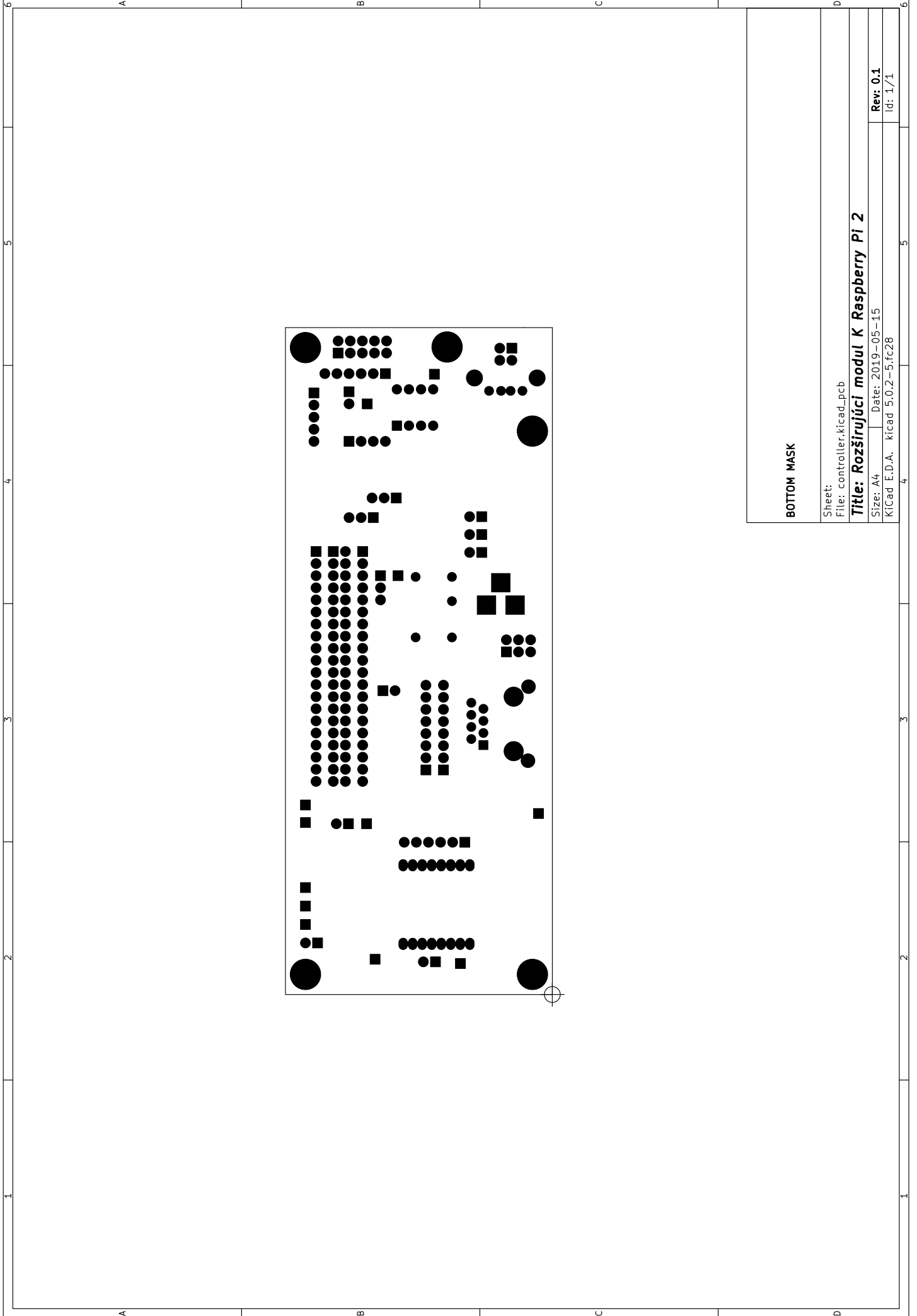


TOP COPPER

Sheet:
File: controller.kicad_pcb

Title: Rozšířující modul K Raspberry Pi 2

Size: A4	Date: 2019-05-15	Rev: 0.1
KiCad: E.D.A.	Kicad: 5.0.2-5.fc28	Id: 1/1



BOTTOM MASK

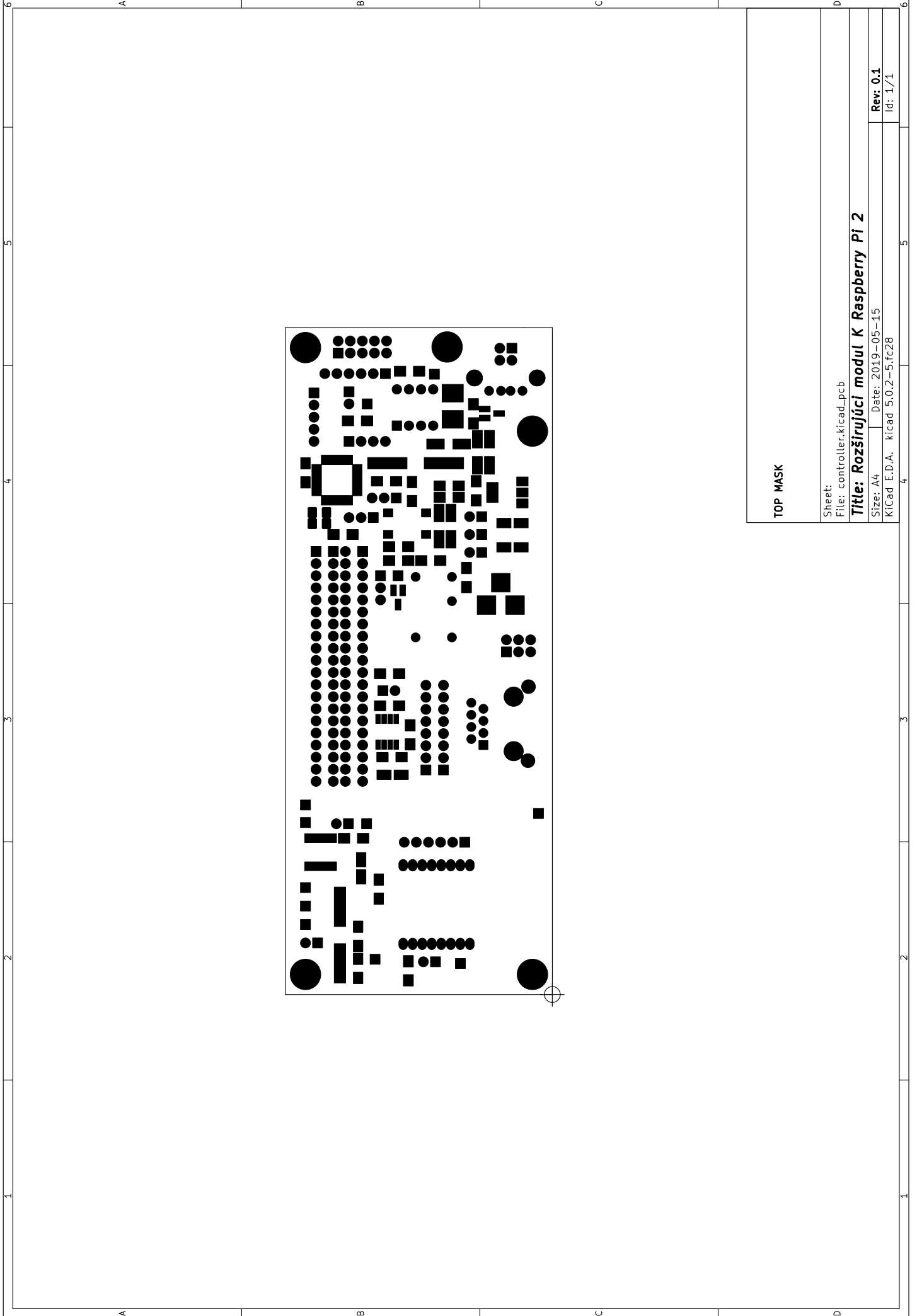
Sheet:
File: controller.kicad_pcb

Title: Rozšířující modul K Raspberry Pi 2

Size: A4 Date: 2019-05-15

Rev: 0.1

Id: 1/1

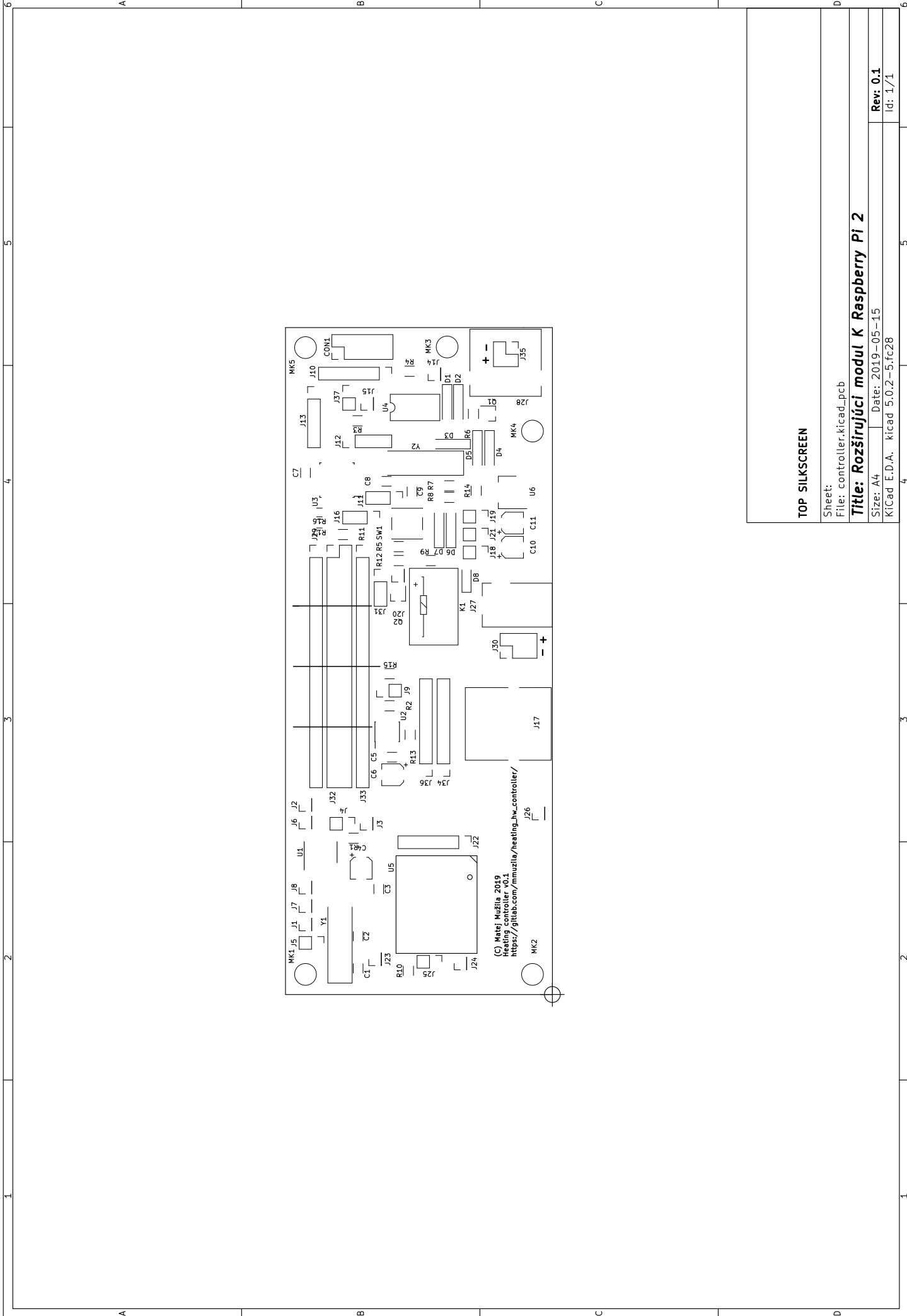


TOP MASK

Sheet:
File: controller.kicad_pcb

Title: Rozšířující modul K Raspberry Pi 2

Size: A4 Date: 2019-05-15 Rev: 0.1
KICad: E.D.A. kicad 5.0.2-5.fc28 Id: 1/1



TOP SILKSCREEN

Sheet:
 File: controller.kicad_pcb

Title: Rozšířující modul K Raspberry Pi 2

Size: A4 Date: 2019-05-15

KiCad E.D.A. kicad 5.0.2-5.fc28

Rev: 0.1

Id: 1/1

B.3 Zoznam komponentov

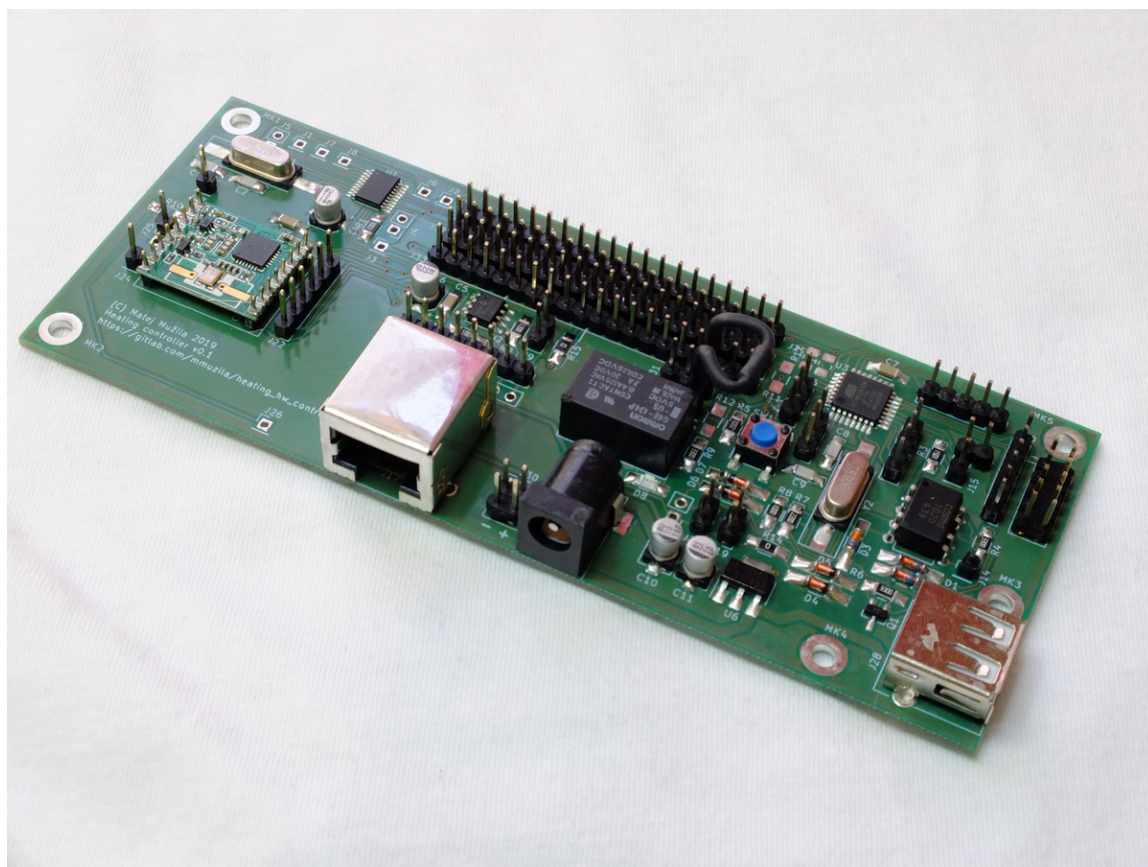
Tabuľka B.1: Zoznam komponentov pre rozširujúci modul
k *Raspberry Pi 2*

Označenie	Hodnota	Identifikátor u predajcu	Púzdro
C1	22p	GME:905-099	1206
C10	10u	TME:RC1C106M04005VR	Elec 4x5.3
C11	22u	TME:RC0J226M04005VR	Elec 4x5.3
C2	22p	GME:905-099	1206
C3	100n	GME:905-081	1206
C4	10u	TME:RC1C106M04005VR	Elec 4x5.3
C5	100n	GME:905-081	1206
C6	22u	TME:RC0J226M04005VR	Elec 4x5.3
C7	10n	GME:905-111	1206
C8	22p	GME:905-099	1206
C9	22p	GME:905-099	1206
CON1	AVR-ISP-10	TME:ZL202-10G	
D1	4V3	GME:919-026	MiniMELF
D2	15V	GME:919-006	MiniMELF
D3	4V7	GME:919-012	MiniMELF
D4	1N4148	GME:917-001	MiniMELF
D5	1N4148	GME:917-001	MiniMELF
D6	1N4148	GME:917-001	MiniMELF
D7	1N4148	GME:917-001	MiniMELF
D8	BL-HKC33H-TRB	GME:960-270	1206
J1	01x01		Pitch 2.54 mm
J10	01x06		Pitch 2.54 mm
J11	01x03		Pitch 2.54 mm
J12	01x04		Pitch 2.54 mm
J13	01x05		Pitch 2.54 mm
J14	01x01		Pitch 2.54 mm
J15	01x01		Pitch 2.54 mm
J16	01x03		Pitch 2.54 mm

J17	MX-85505-5113	TME:MX-85505-5113	
J18	01x02		Pitch 2.54 mm
J19	01x02		Pitch 2.54 mm
J2	01x01		Pitch 2.54 mm
J20	01x01		Pitch 2.54 mm
J21	01x02		Pitch 2.54 mm
J22	01x06		Pitch 2.54 mm
J23	01x01		Pitch 2.54 mm
J24	01x01		Pitch 2.54 mm
J25	01x02		Pitch 2.54 mm
J26	01x01		Pitch 2.54 mm
J27	NEB21R	TME:NEB21R	
J28	USB A	TME:KEYS921	
J29	01x20		Pitch 2.54 mm
J3	01x01		Pitch 2.54 mm
J30	02x03		Pitch 2.54 mm
J31	01x03		Pitch 2.54 mm
J32	02x20		Pitch 2.54 mm
J33	01x20		Pitch 2.54 mm
J34	01x08		Pitch 2.54 mm
J35	02x02		Pitch 2.54 mm
J36	01x08		Pitch 2.54 mm
J37	01x02		Pitch 2.54 mm
J4	01x02		Pitch 2.54 mm
J5	01x02		Pitch 2.54 mm
J6	01x01		Pitch 2.54 mm
J7	01x01		Pitch 2.54 mm
J8	01x01		Pitch 2.54 mm
J9	01x02		Pitch 2.54 mm
K1	G6E-134P-US-5D	TME:G6E-134P-US-5D	
MK1	M3		
MK2	M3		
MK3	M3		

MK4	M3		
MK5	M3		
Q1	BC856	GME:912-030	SOT-23
Q2	2N7002	TME:2N7002-LGE	SOT-23
R1	10k	GME:900-508	1206
R10	0	GME:900-001	1206
R11	inf		1206
R12	inf		1206
R13	1k	GME:900-033	1206
R14	0	GME:900-001	1206
R15	inf		1206
R16	1k	GME:900-033	1206
R17	1k	GME:900-033	1206
R2	120	GME:900-232	1206
R3	1k5	GME:900-214	1206
R4	330	GME:900-310	1206
R5	10k	GME:900-508	1206
R6	330	GME:900-310	1206
R7	220	GME:900-295	1206
R8	100	GME:900-192	1206
R9	270	GME:900-067	1206
SW1	G6E-134P-US-5D	TME:PTS645SM43SMTR92LF	
U1	MCP2515-I/ST	GME:959-360	TSSOP-20
U2	MCP2551-I/SN	TME:MCP2551-I/SN	SOIC-8
U3	ATMEGA328-AU	TME:ATMEGA328-AU	TQFP-32
U4	PC827	GME:523-055	DIP-8
U5	RFM69HW	TME:RFM69HW-433S2	
U6	AZ1117-3.3	TME:AZ1117CH-3.3TRG1	SOT-223-3
Y1	8MHz	GME:970-023	HC49-SD
Y2	8MHz	GME:970-023	HC49-SD

B.4 Obrázky



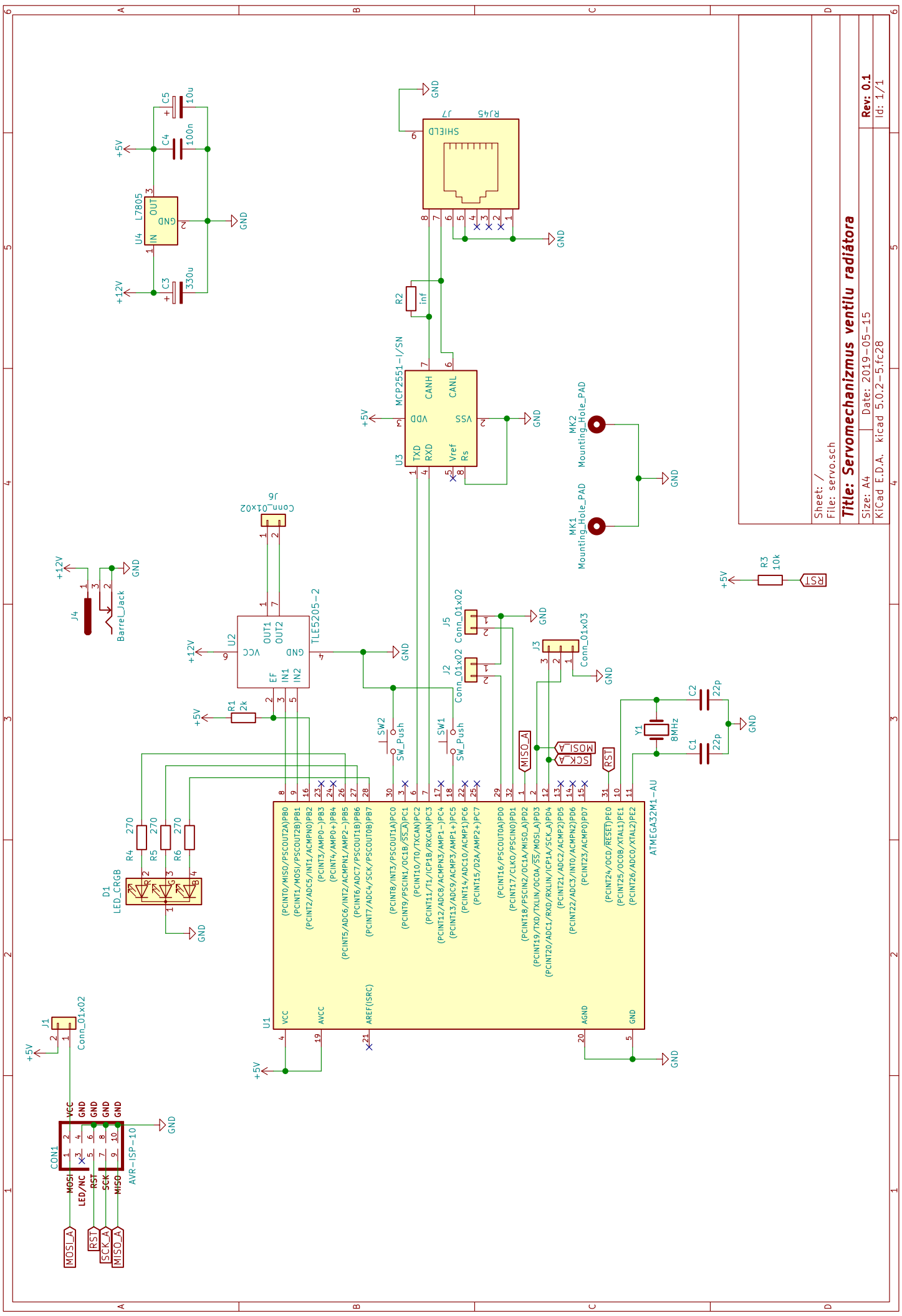
Obr. B.1: Fotografia rozširujúceho modulu k *Raspberry Pi 2*

Príloha C

Kompletá konštrukcia servomechanizmu ovládania ventilu radiátora

C.1 Elektrická konštrukcia

C.1.1 Schéma

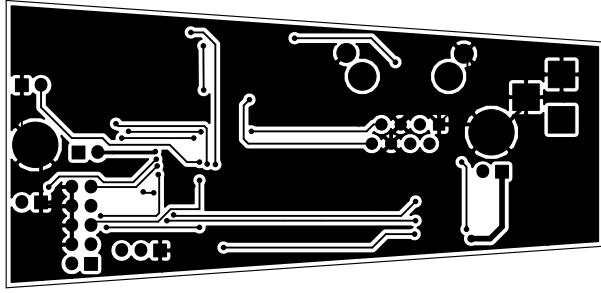
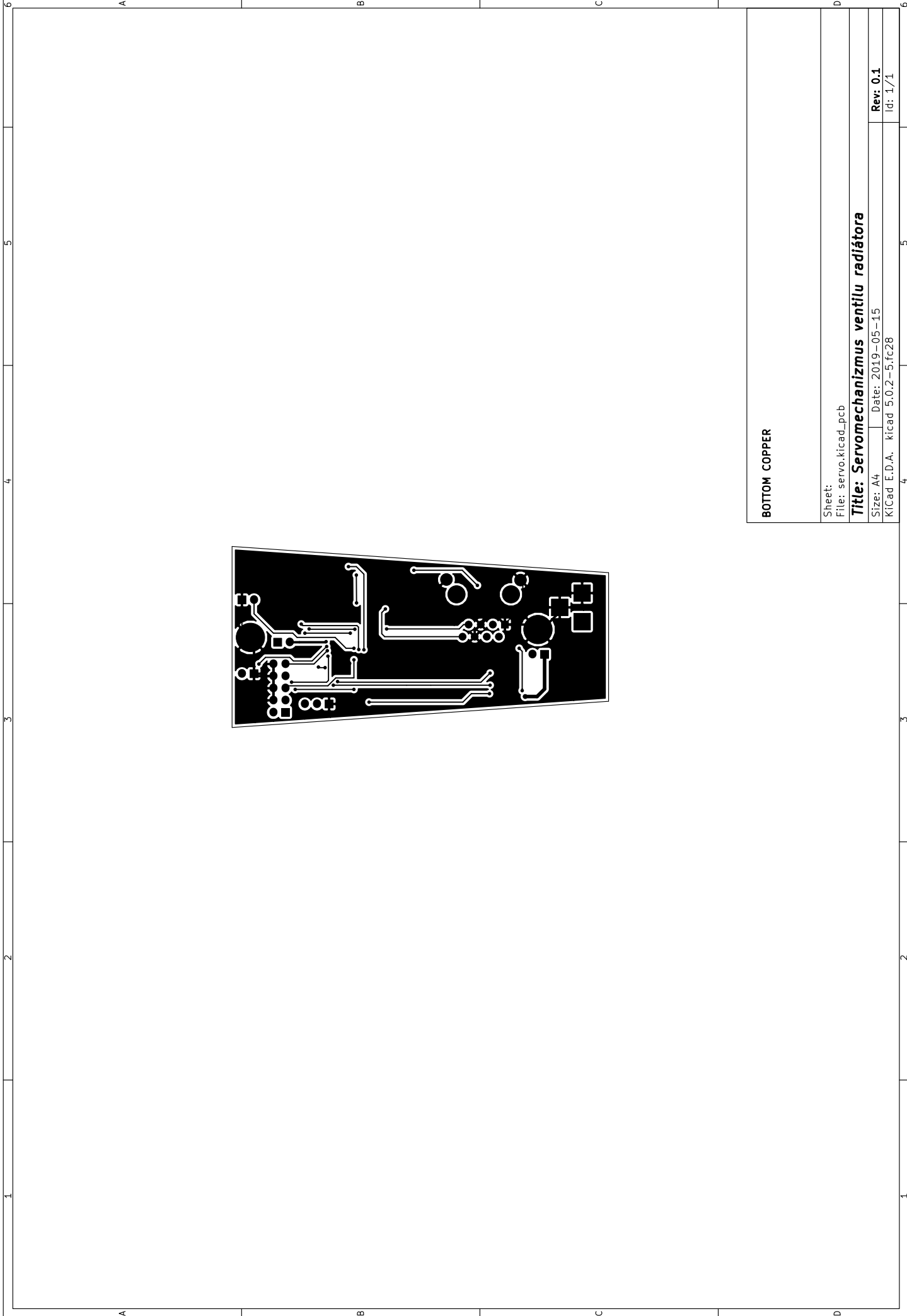


Sheet: /
File: servo.sch

Title: Servomechanizmus ventili radiátora

Size: A4
Date: 2019-05-15
KICad: E.D.A. kicad 5.0.2-5.fc28
Rev: 0.1
Id: 1/1

C.1.2 Plošný spoj



BOTTOM COPPER

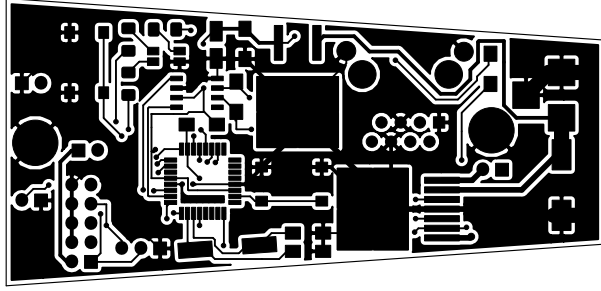
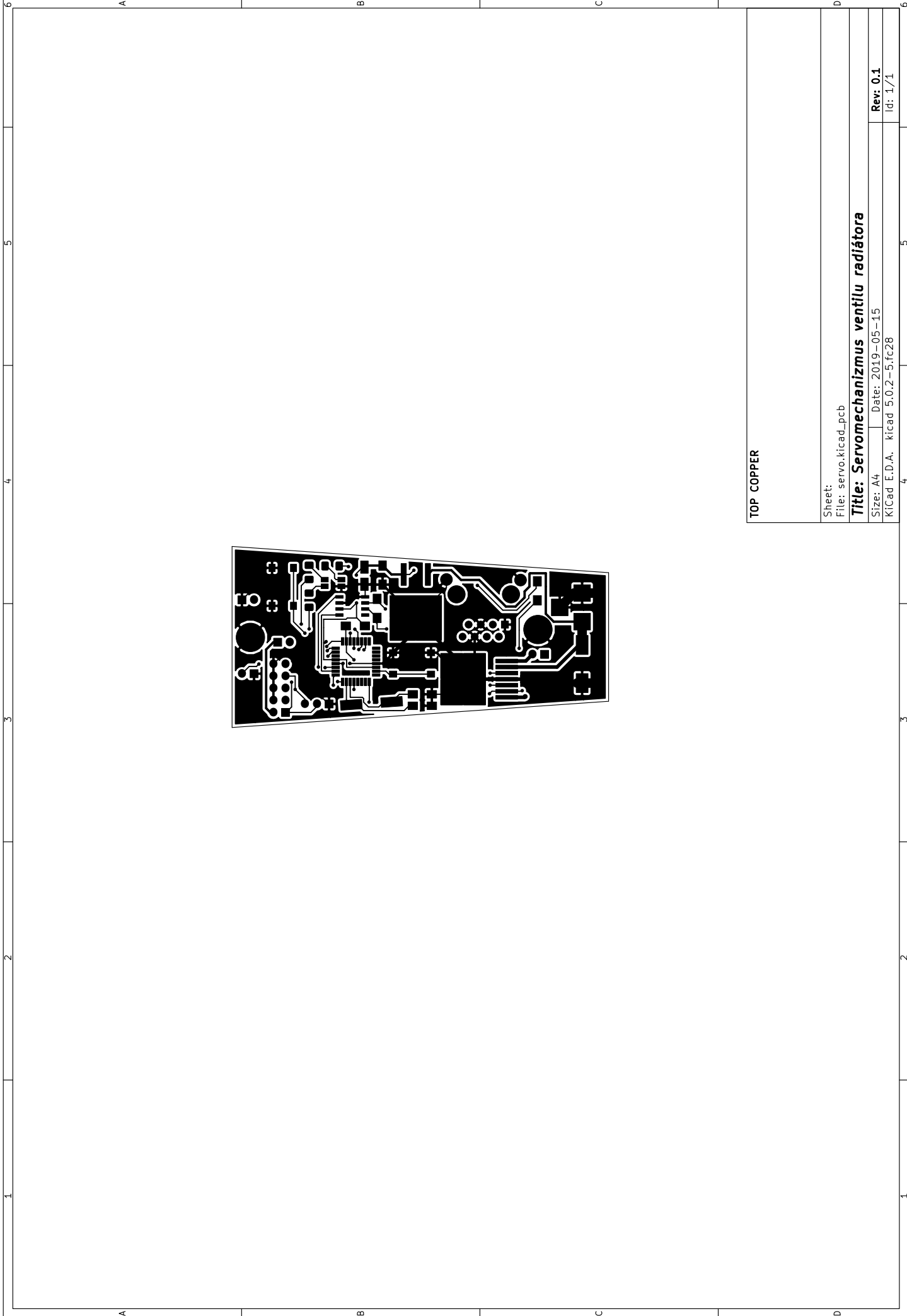
Sheet:
File: servo.kicad_pcb

Title: Servomechanizmus ventilu radiátora

Size: A4 Date: 2019-05-15

Rev: 0.1

Id: 1/1



TOP COPPER

Sheet:
File: servo.kicad_pcb

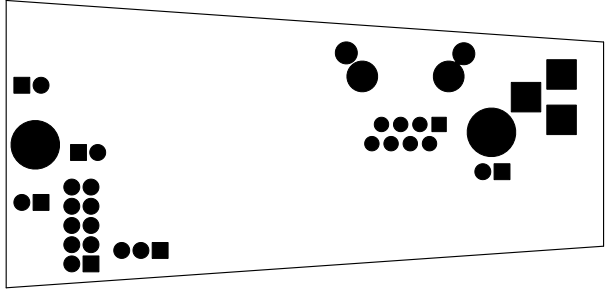
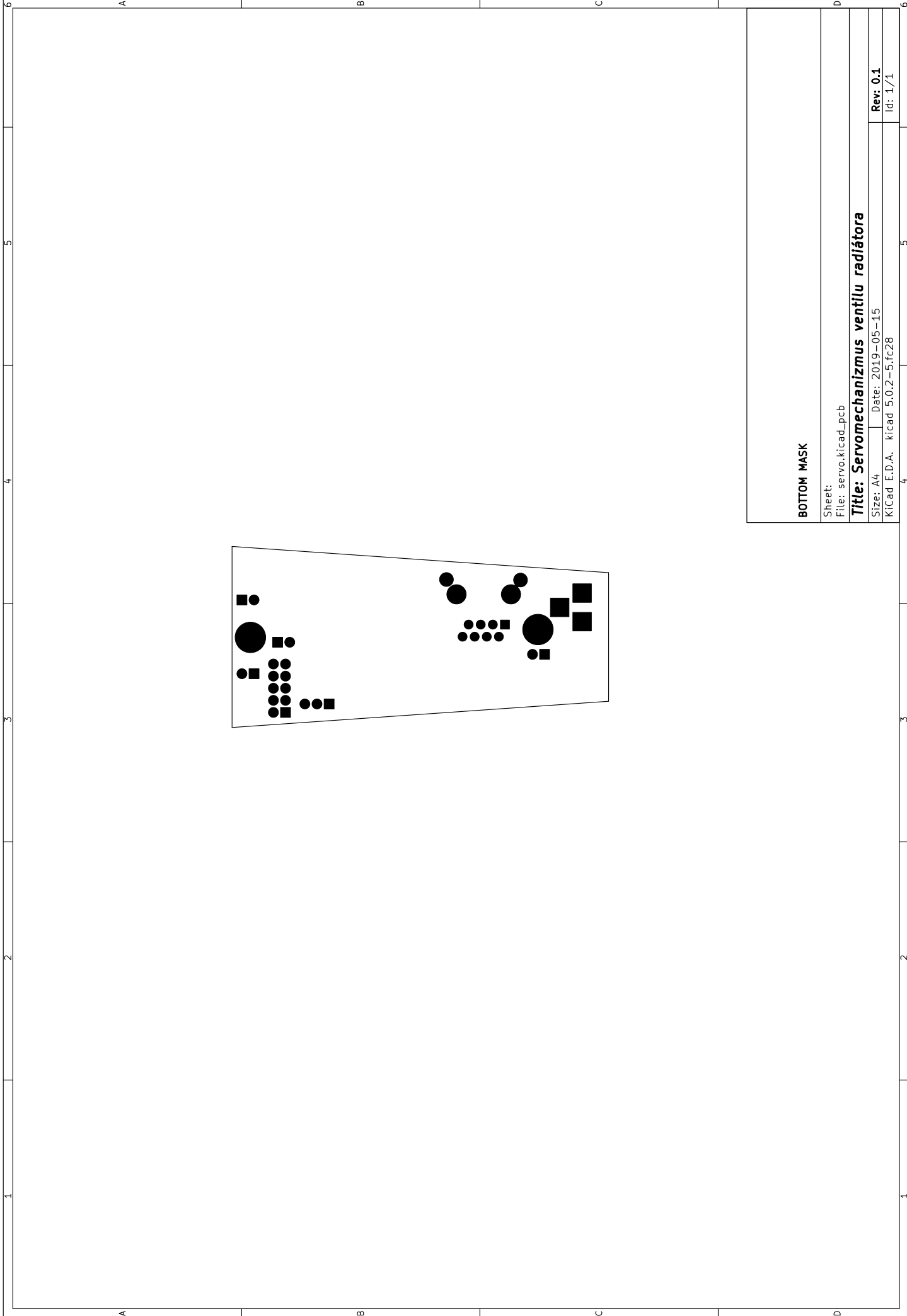
Title: Servomechanizmus ventilu radiátora

Size: A4 Date: 2019-05-15

KiCad: E.D.A. Kicad 5.0.2-5.fc28

Rev: 0.1

Id: 1/1



BOTTOM MASK

Sheet:
File: servo.kicad_pcb

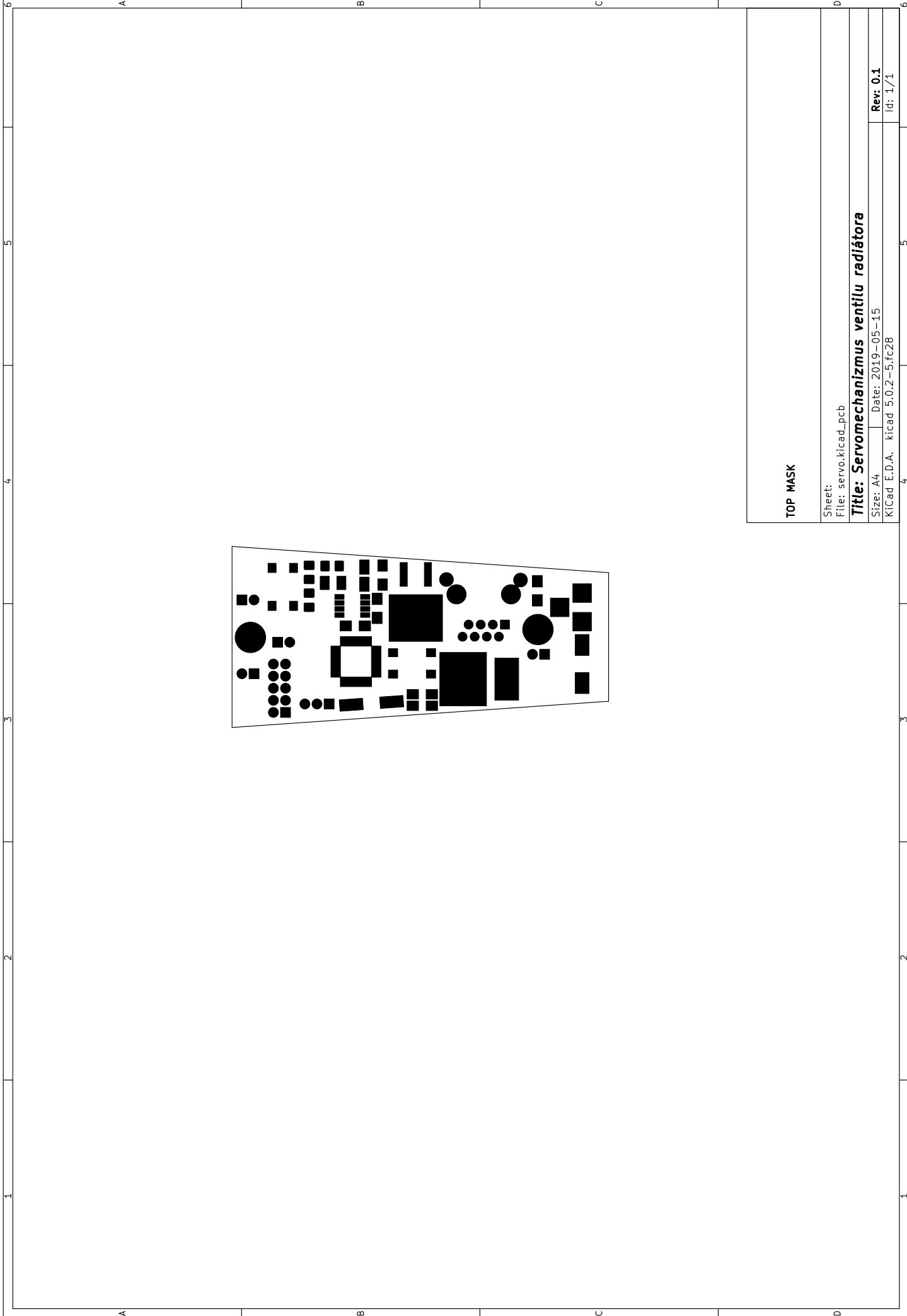
Title: Servomechanizmus ventilu radiátora

Size: A4 Date: 2019-05-15

KiCad: E.D.A. kicad 5.0.2-5.fc28

Rev: 0.1

Id: 1/1



TOP MASK

Sheet:
File: servo.kicad_pcb

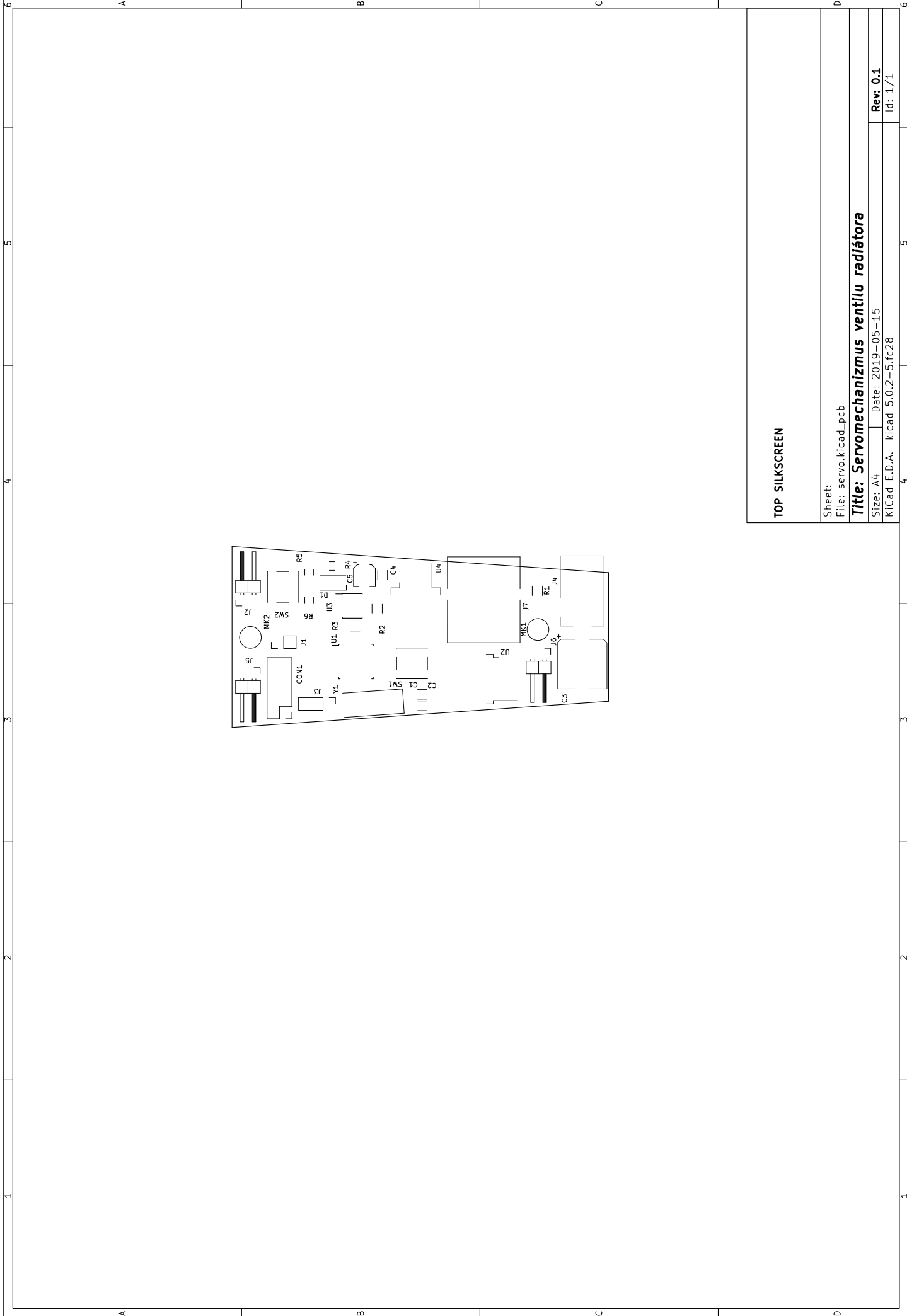
Title: Servomechanizmus ventilu radiátora

Size: A4 Date: 2019-05-15

KiCad E.D.A. kicad 5.0.2-5.fc28

Rev: 0.1

Id: 1/1



TOP SILKSCREEN

Sheet:
File: servo.kicad_pcb

Title: Servomechanizmus ventilu radiátora

Size: A4 Date: 2019-05-15

KiCad E.D.A. kicad 5.0.2-5.fc28

Rev: 0.1

Id: 1/1

C.1.3 Zoznam komponentov

Tabuľka C.1: Zoznam elektrických komponentov pre ser-
vomechanizmus ventilu

Označenie	Hodnota	Identifikátor u predajcu	Púzdro
C1	22p	GME:905-099	1206
C2	22p	GME:905-099	1206
C3	330u	TME:CA1E337M10010VR	Elec 10x10
C4	100n	GME:905-081	1206
C5	10u	TME:RC1C106M04005VR	Elec 4x5.3
CON1	AVR-ISP-10	TME:ZL202-10G	Pitch 2.54 mm
D1	HSMF-C118	TME:HSMF-C118	1210
J1	01x02		Pitch 2.54 mm
J2	01x02		Pitch 2.54 mm
J3	01x03		Pitch2.54 mm
J4	NEB21R	TME:NEB21R	
J5	01x02		Pitch 2.54 mm
J6	01x02		90°, Pitch 2.54 mm
J7	MX-85505-5113	TME:MX-85505-5113	
MK1	M3		
MK2	M3		
R1	2k	GME:900-070	1206
R2	inf		1206
R3	10k	GME:900-508	1206
R4	270	GME:900-070	1206
R5	270	GME:900-070	1206
R6	270	GME:900-070	1206
SW1	PTS645SM	TME:PTS645SM43SMTR92LF	
SW2	PTS645SM	TME:PTS645SM43SMTR92LF	
U1	ATMEGA32M1-AU	TME:ATMEGA32M1-AU	TQFP-32
U2	TLE5205-2	TME:TLE5205-2G	SMD:TO-263-7
U3	MCP2551-I/SN	TME:MCP2551-I/SN	SOIC-8
U4	L7805	TME:L7805ABD2T	TO-263-2
Y1	8MHz	GME:970-023	HC49-SD

	MSW-04A	GME:631-572	
	MSW-04A	GME:631-572	
	KONPC-SPK-2	GME:840-002	
	KONPC-SPK-2	GME:840-002	
	KONPC-SPK-2	GME:840-002	
	KONPC-SPK-PI	GME:840-004	
	KONPC-SPK-PI	GME:840-004	
	KONPC-SPK-PI	GME:840-004	
	KONPC-SPK-PI	GME:840-004	
	KONPC-SPK-PI	GME:840-004	
	KONPC-SPK-PI	GME:840-004	

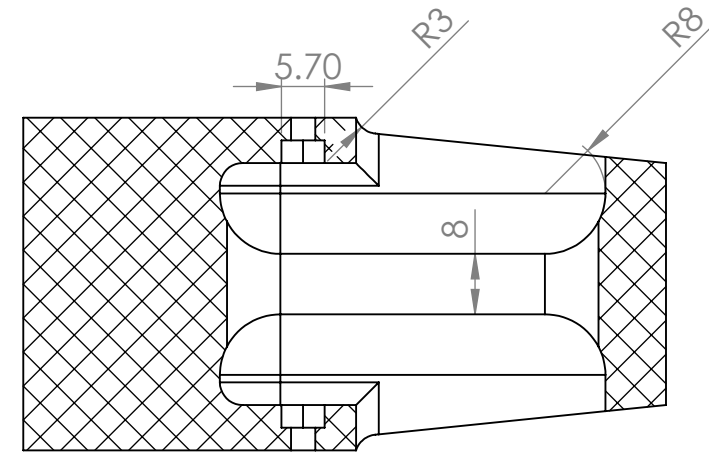
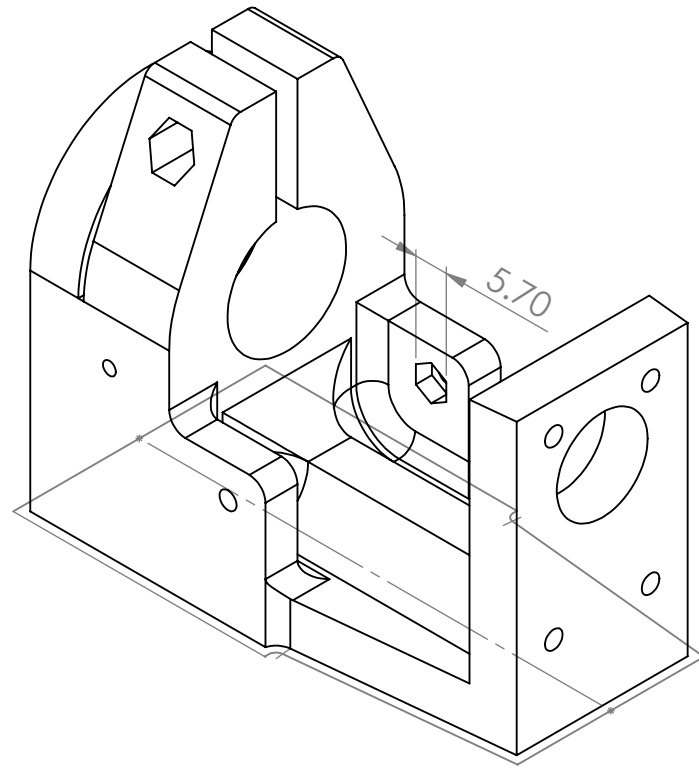
C.2 Mechanická konštrukcia

C.2.1 Zoznam spojovacieho materiálu

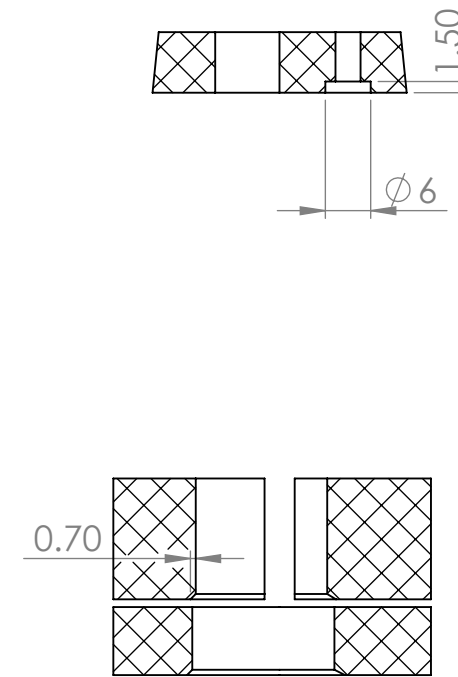
Tabuľka C.2: Zoznam spojovacieho materiálu pre servomechanizmus ventilu

Označenie	Množstvo
Skrutka M3×16 ISO 7380	2
Skrutka M3×18 ISO 7380	2
Skrutka M3×16 DIN 933	3
Matica krídlová M3 DIN 315	1
Podložka 3.2 DIN 125A	1
Podložka 4.3 DIN 125A	1
Skrutka M4×20 DIN 912	1
Matica M3 DIN 934	2
Matica M4 DIN 934	1
Skrutka M3×10 DIN 912	4
Skrutka M3×30 ČSN 02 1022	2
Skrutka M4×14 DIN 7985, polykarbonát	1

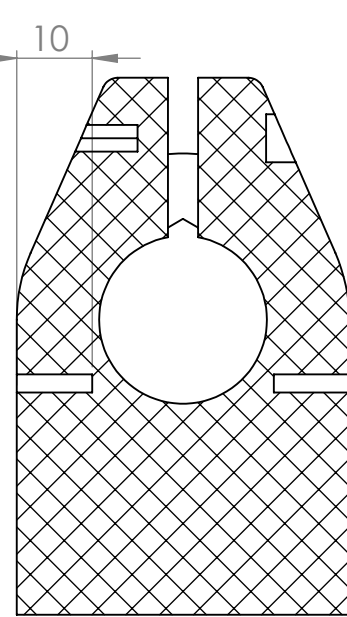
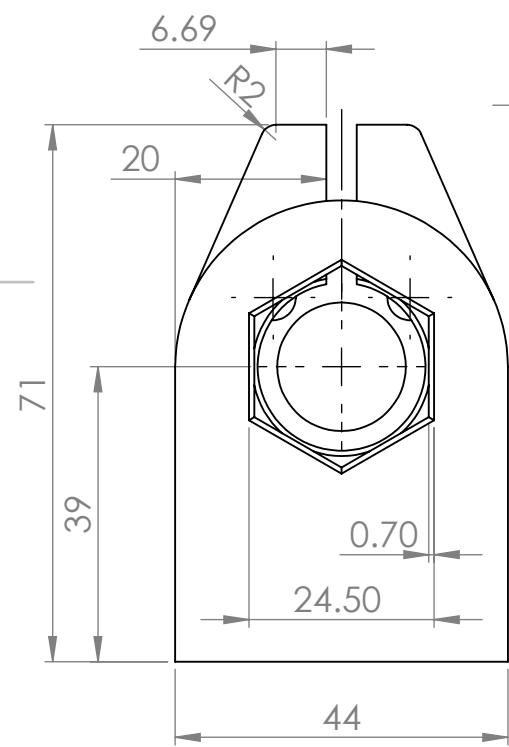
C.2.2 Technické výkresy



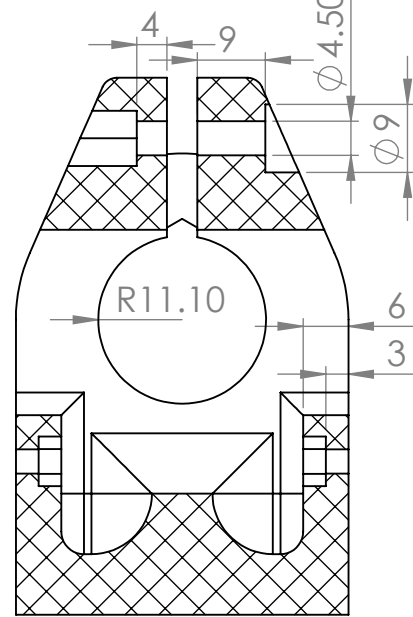
SECTION P-P



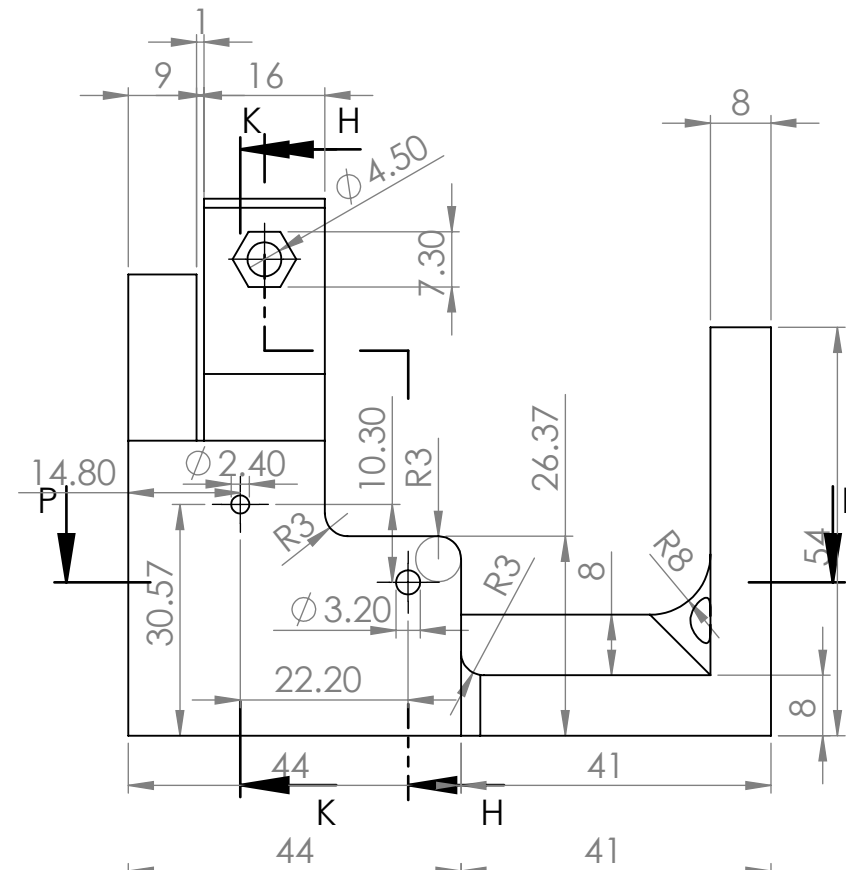
SECTION N-N



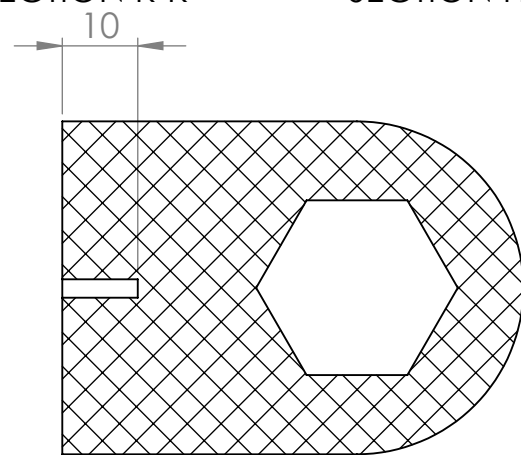
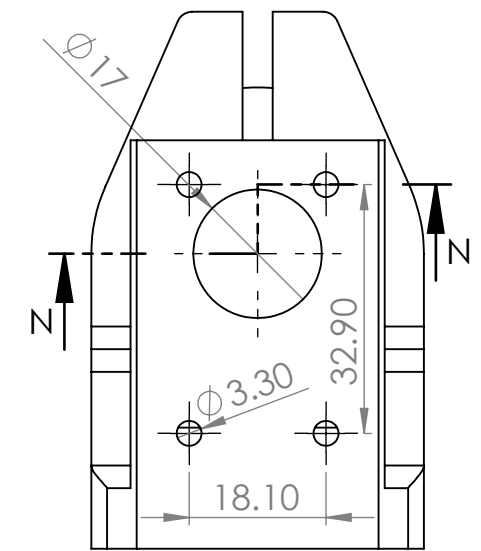
SECTION K-K



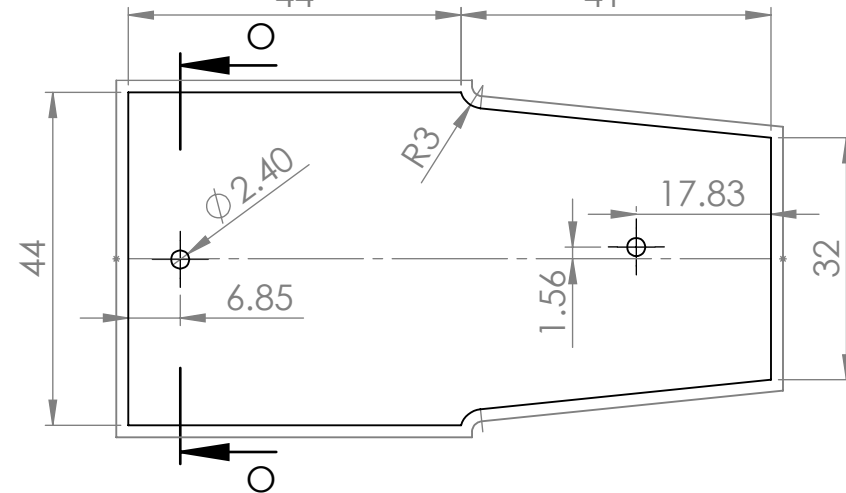
SECTION H-H



SECTION O-O



SECTION O-O



DO NOT SCALE DRAWING		REVISION
TITLE: Držiak		
MATERIAL: PETG		A3
SCALE:1:1	SHEET 1 OF 5	

8 7 6 5 4 3 2 1

F

F

E

E

D

D

C

C

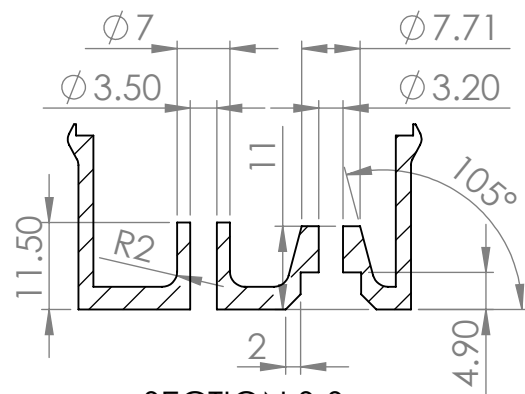
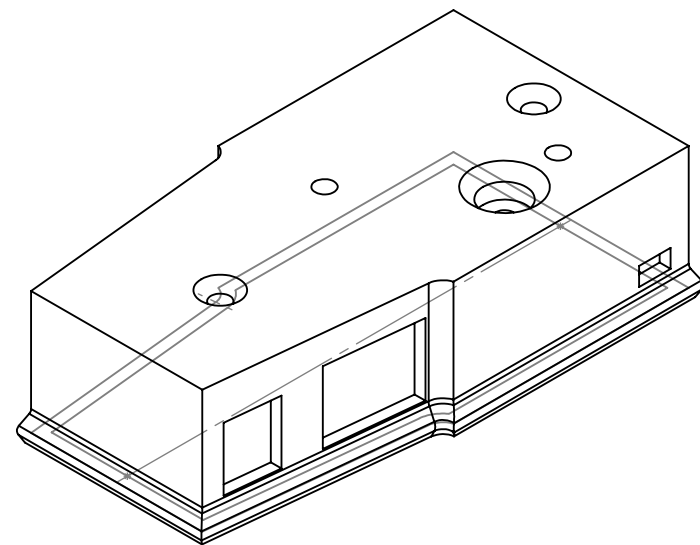
B

B

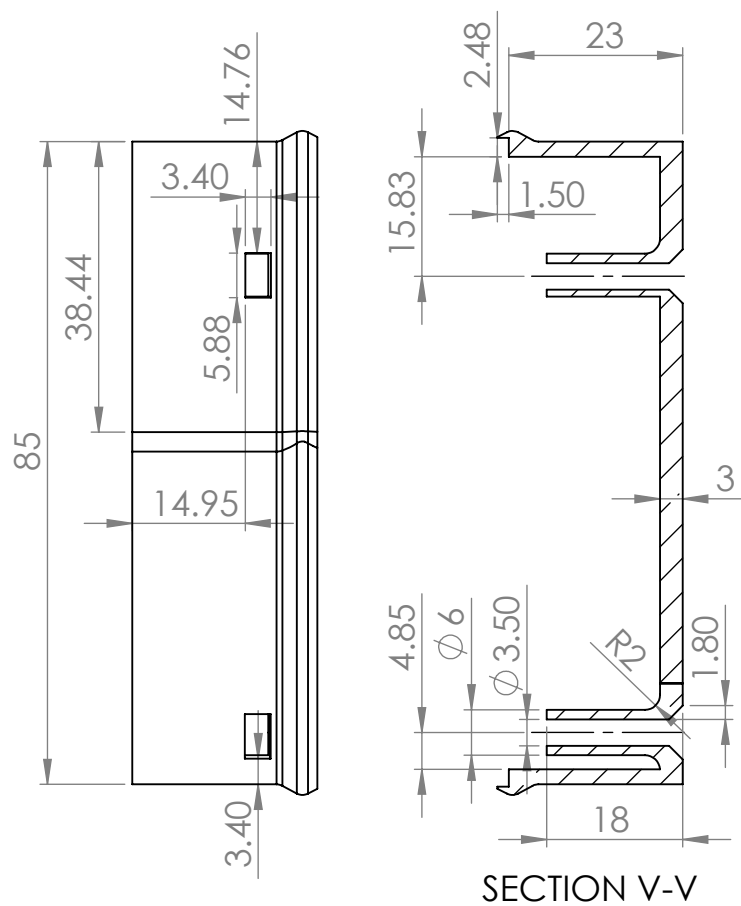
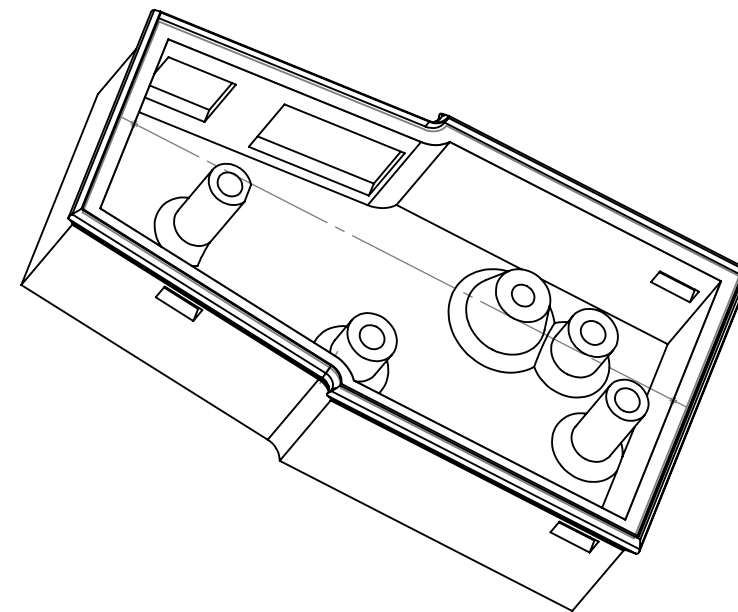
A

A

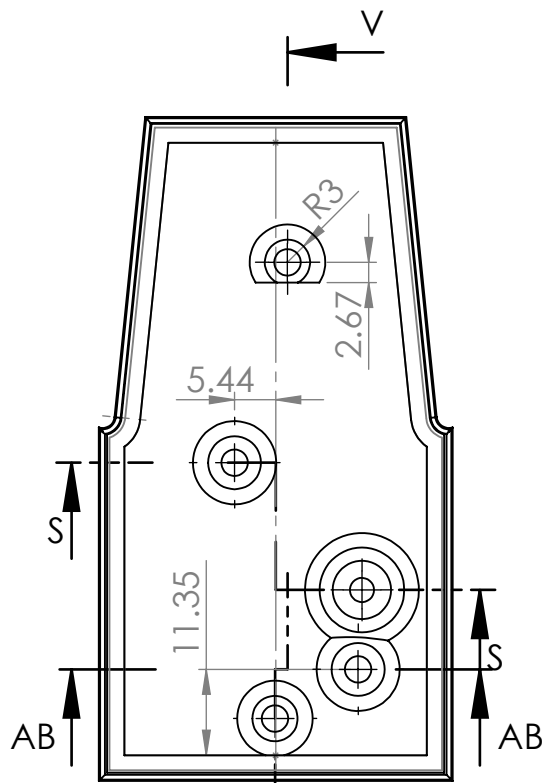
8 7 6 5 4 3 2 1



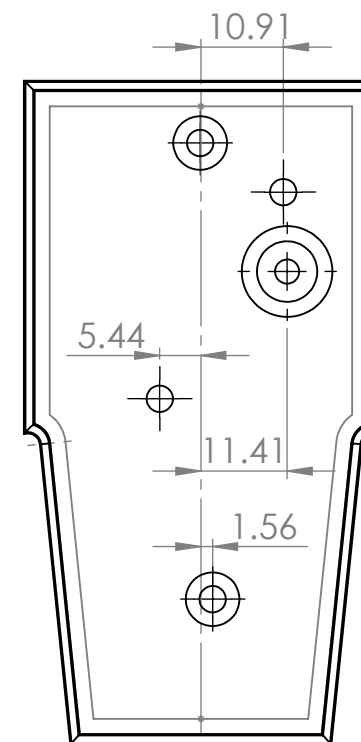
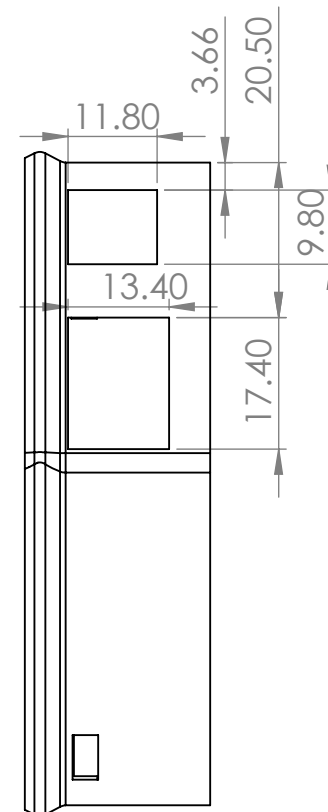
SECTION S-S



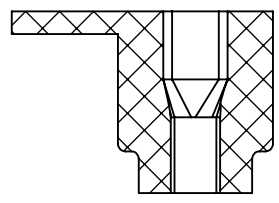
SECTION V-V



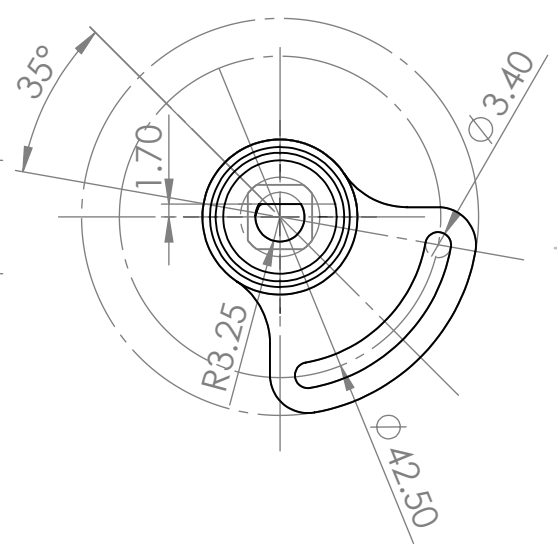
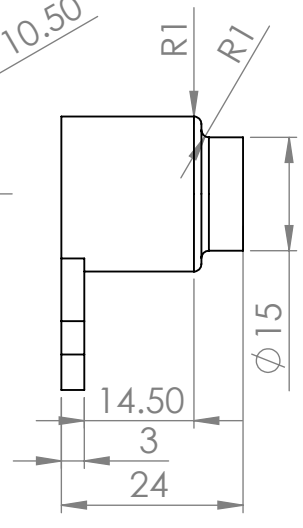
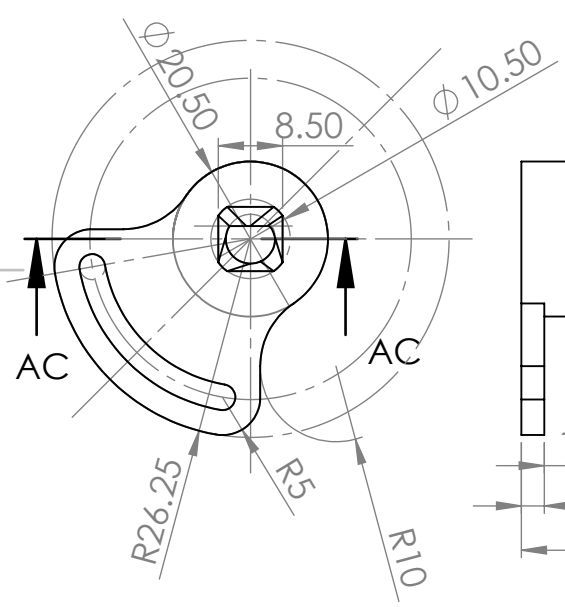
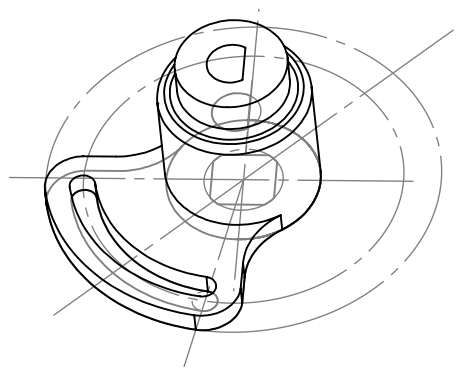
SECTION AB-AB



DO NOT SCALE DRAWING		REVISION
TITLE: Kryt		
MATERIAL: PETG		A3
SCALE: 1:1	SHEET 2 OF 5	



SECTION AC-AC

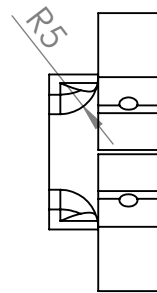
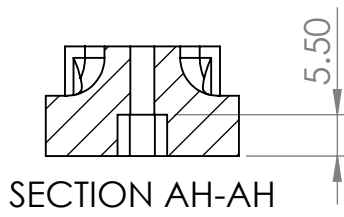
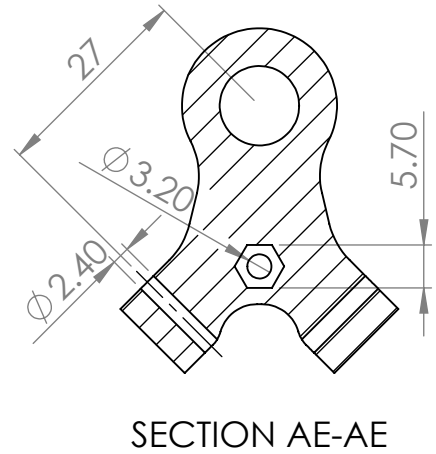
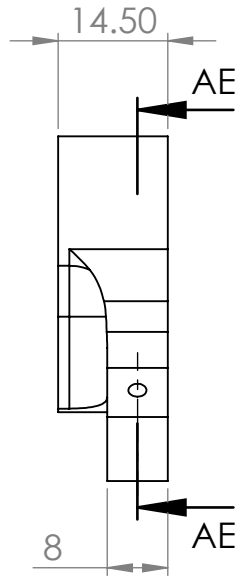
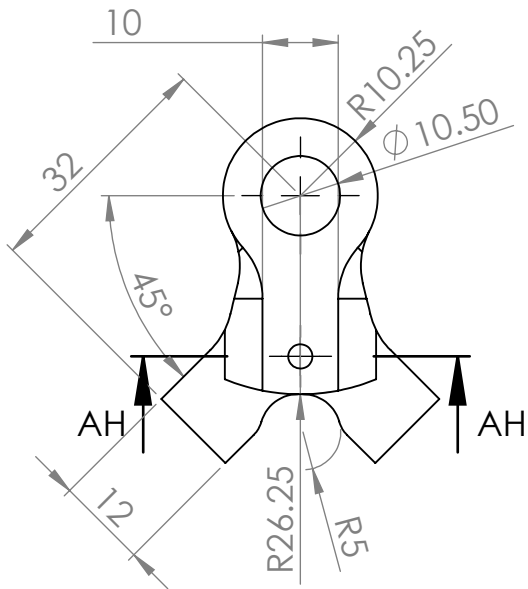
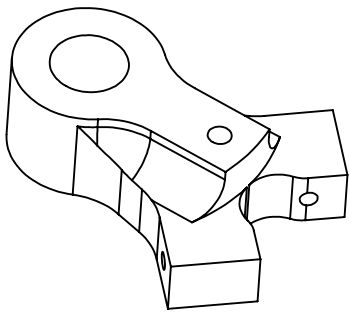


DO NOT SCALE DRAWING		REVISION	
TITLE:			
<h1>Spojka</h1>			
MATERIAL:			A4
PETG			

WEIGHT:

SCALE:1:1

SHEET 3 OF 5

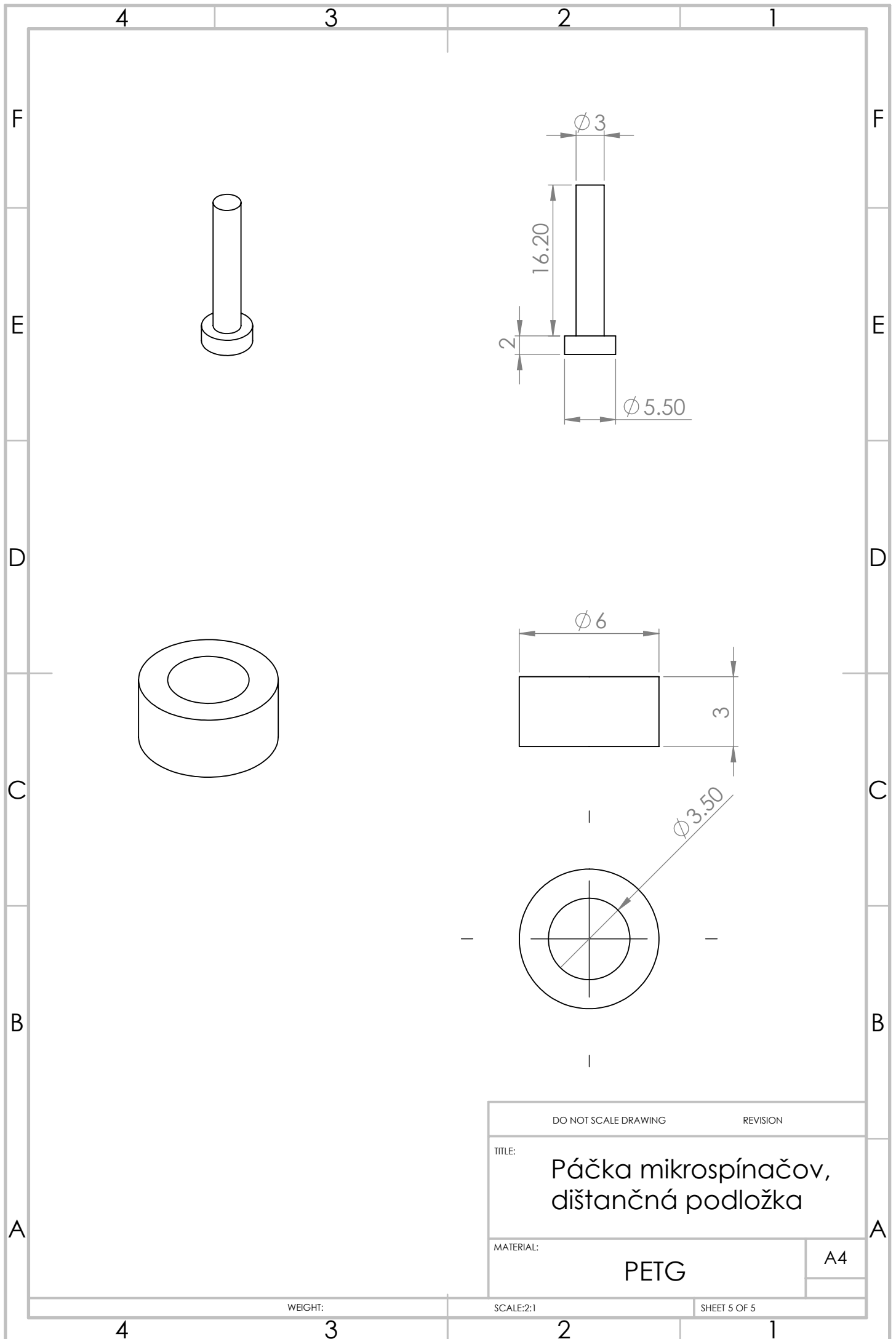


DO NOT SCALE DRAWING		REVISION
TITLE: Páčka koncových vypínačov		
MATERIAL: PETG		A4

WEIGHT:

SCALE:1:1

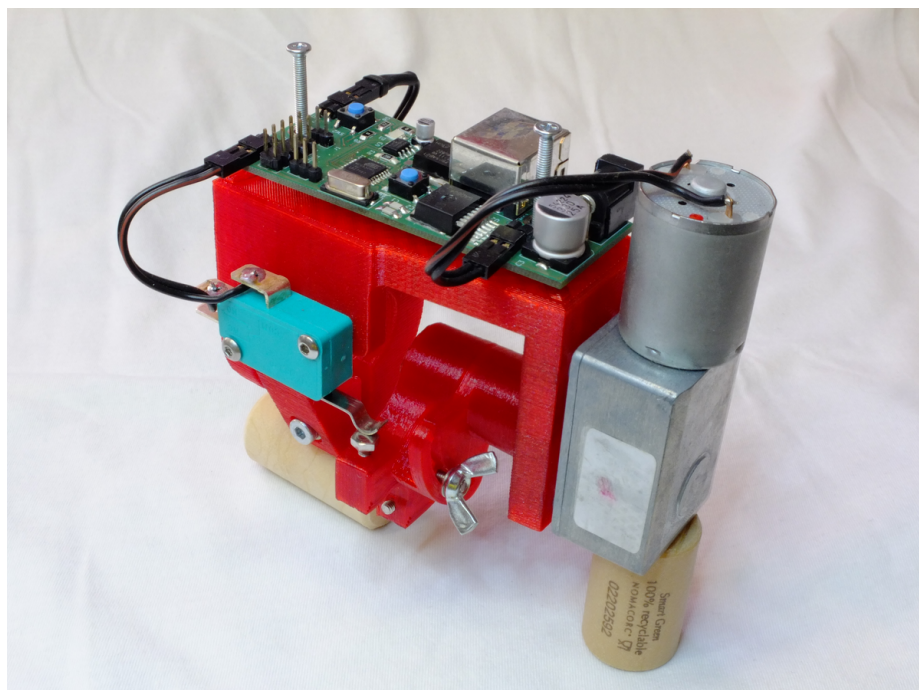
SHEET 4 OF 5



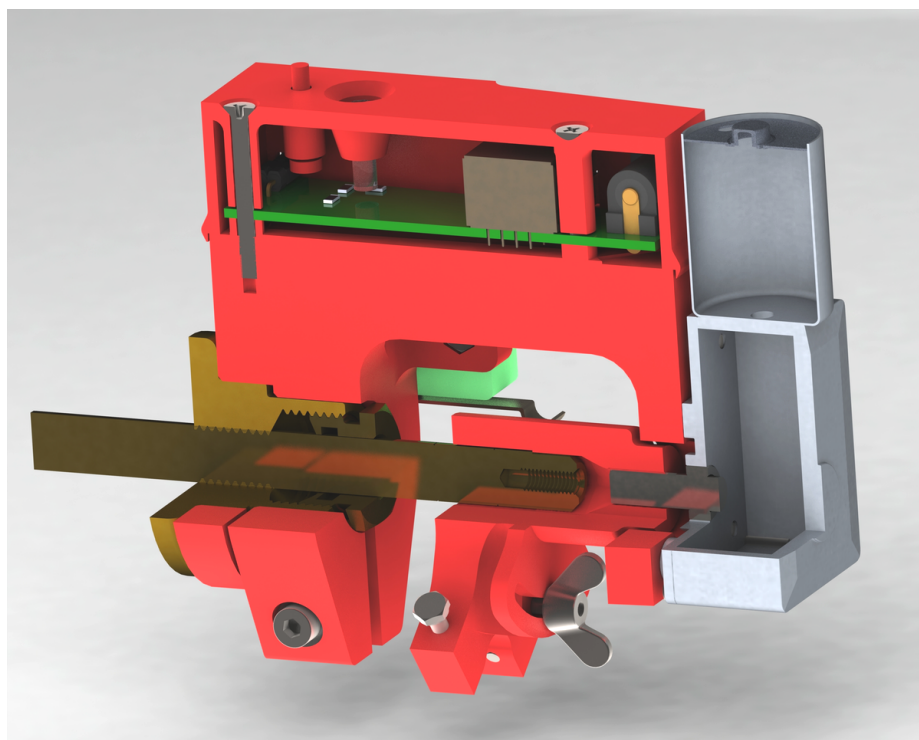
C.3 Obrázky



Obr. C.1: Fotografia plošného spoja servomechanizmu ventilu



Obr. C.2: Fotografia servomechanizmu ventilu bez nasadeného krytu elektroniky

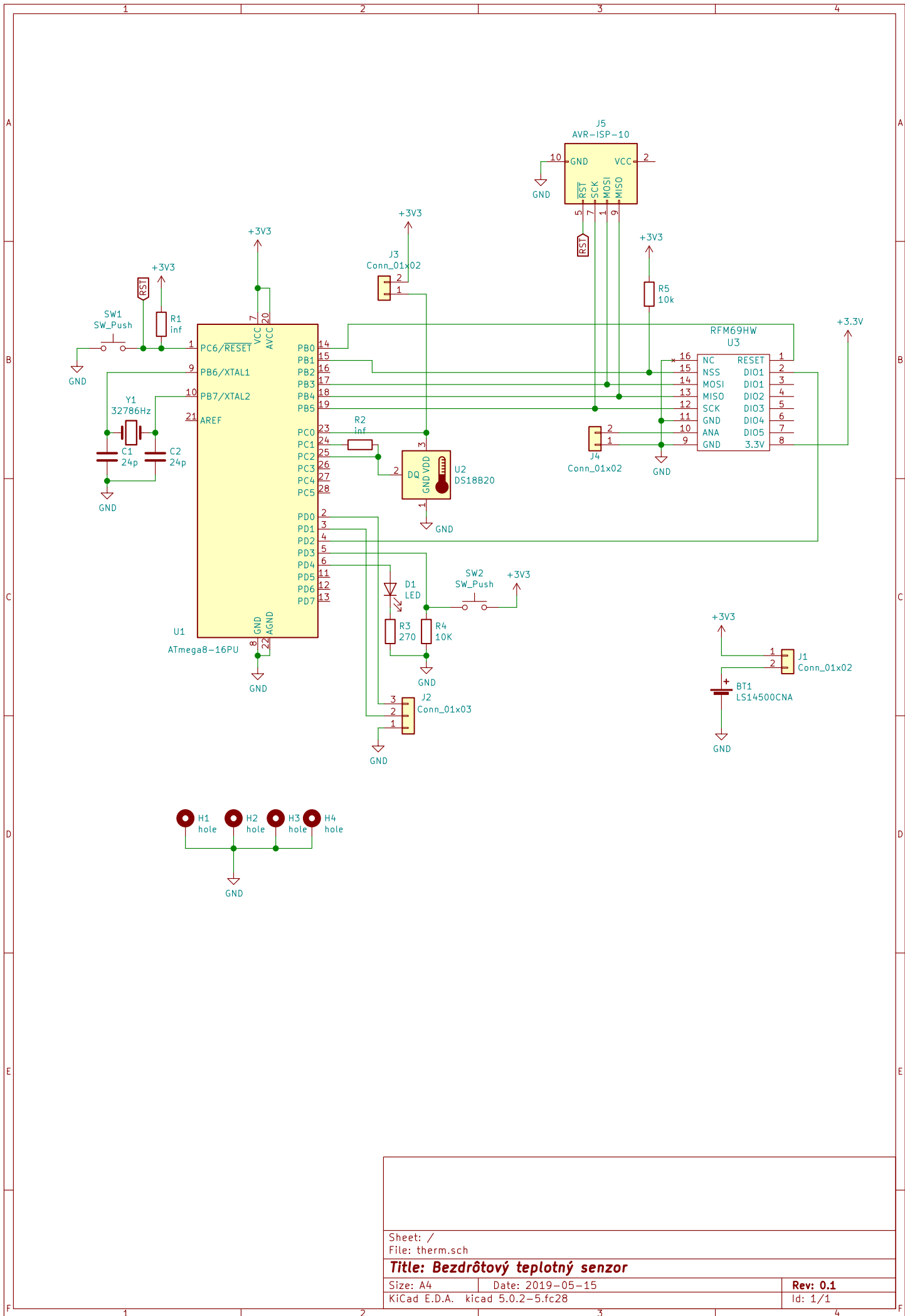


Obr. C.3: Rez zjednodušeným modelom servomechanizmu ovládania ventilu a ventilu samotného

Príloha D

Kompletá konštrukcia bezdrôtového teplotného senzora

D.1 Schéma



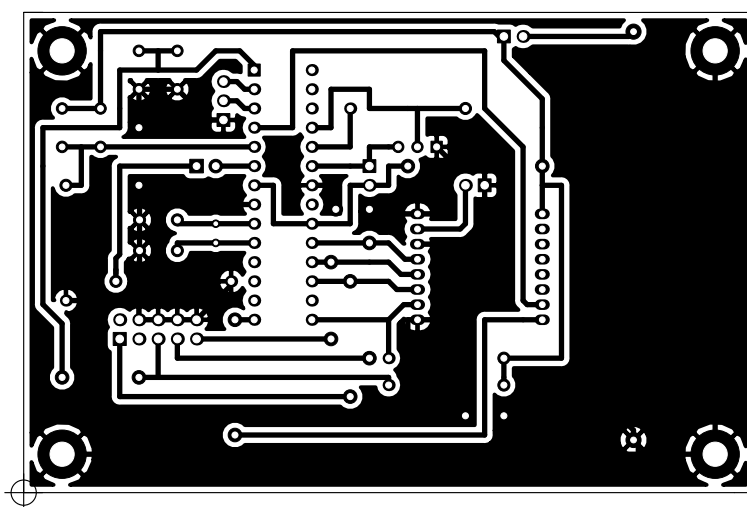
Sheet: /
File: therm.sch

Title: Bezdrôtový teplotný senzor

Size: A4 Date: 2019-05-15
KiCad E.D.A. kicad 5.0.2-5.fc28

Rev: 0.1
Id: 1/1

D.2 Plošný spoj



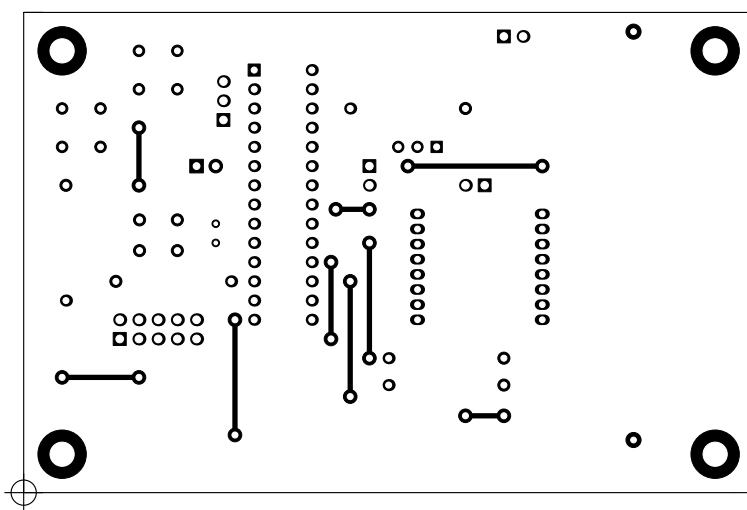
BOTTOM COPPER

Sheet:
File: therm.kicad_pcb

Title: Bezdřetový teplotný senzor

Size: A4 Date: 2019-05-15
KiCad E.D.A. kicad 5.0.2-5.fc28

Rev: 0.1
Id: 1/1



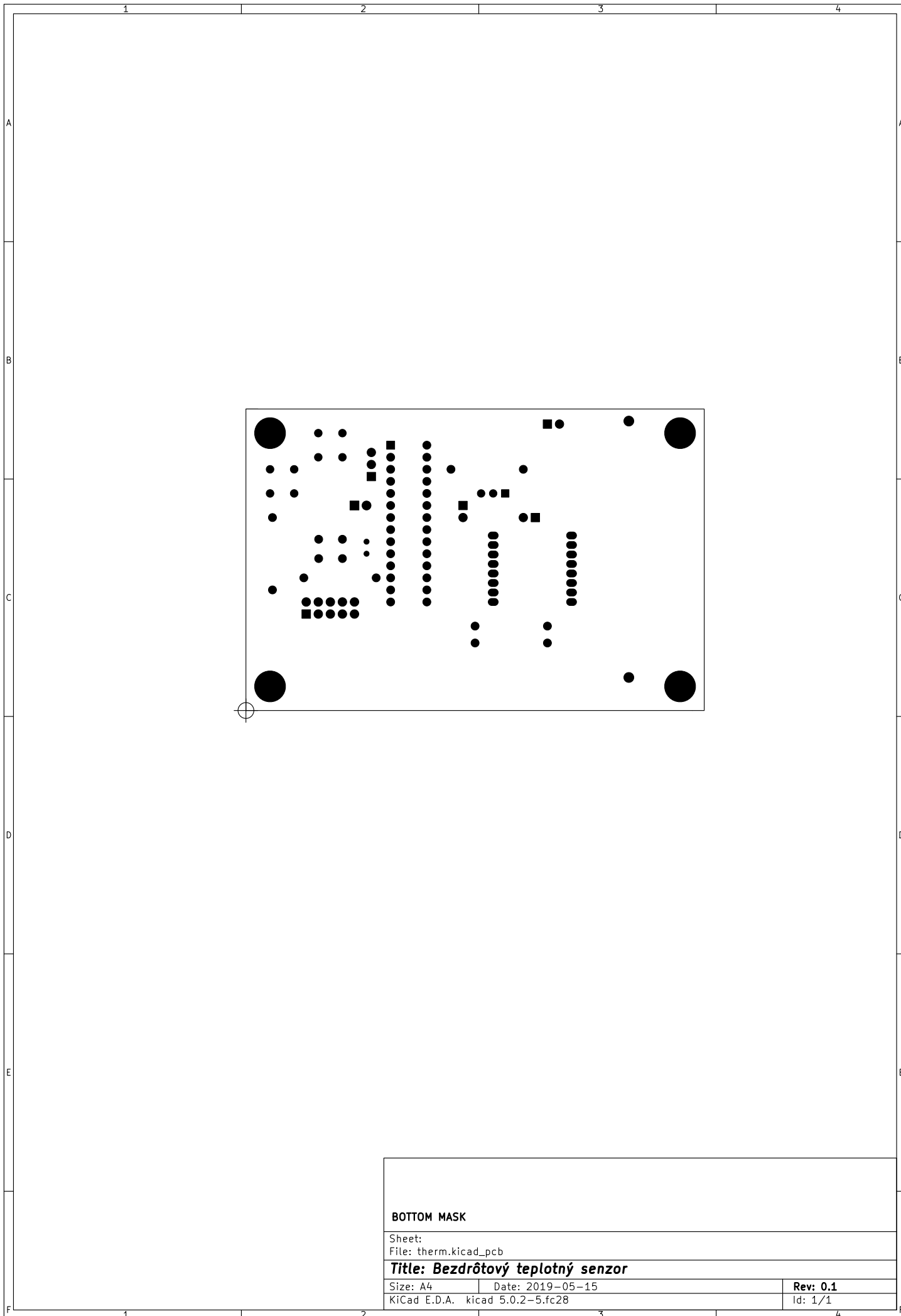
TOP COPPER

Sheet:
File: therm.kicad_pcb

Title: Bezdrôtový teplotný senzor

Size: A4 Date: 2019-05-15
KiCad E.D.A. kicad 5.0.2-5.fc28

Rev: 0.1
Id: 1/1



BOTTOM MASK

Sheet:

File: therm.kicad_pcb

Title: *Bezdrôtový teplotný senzor*

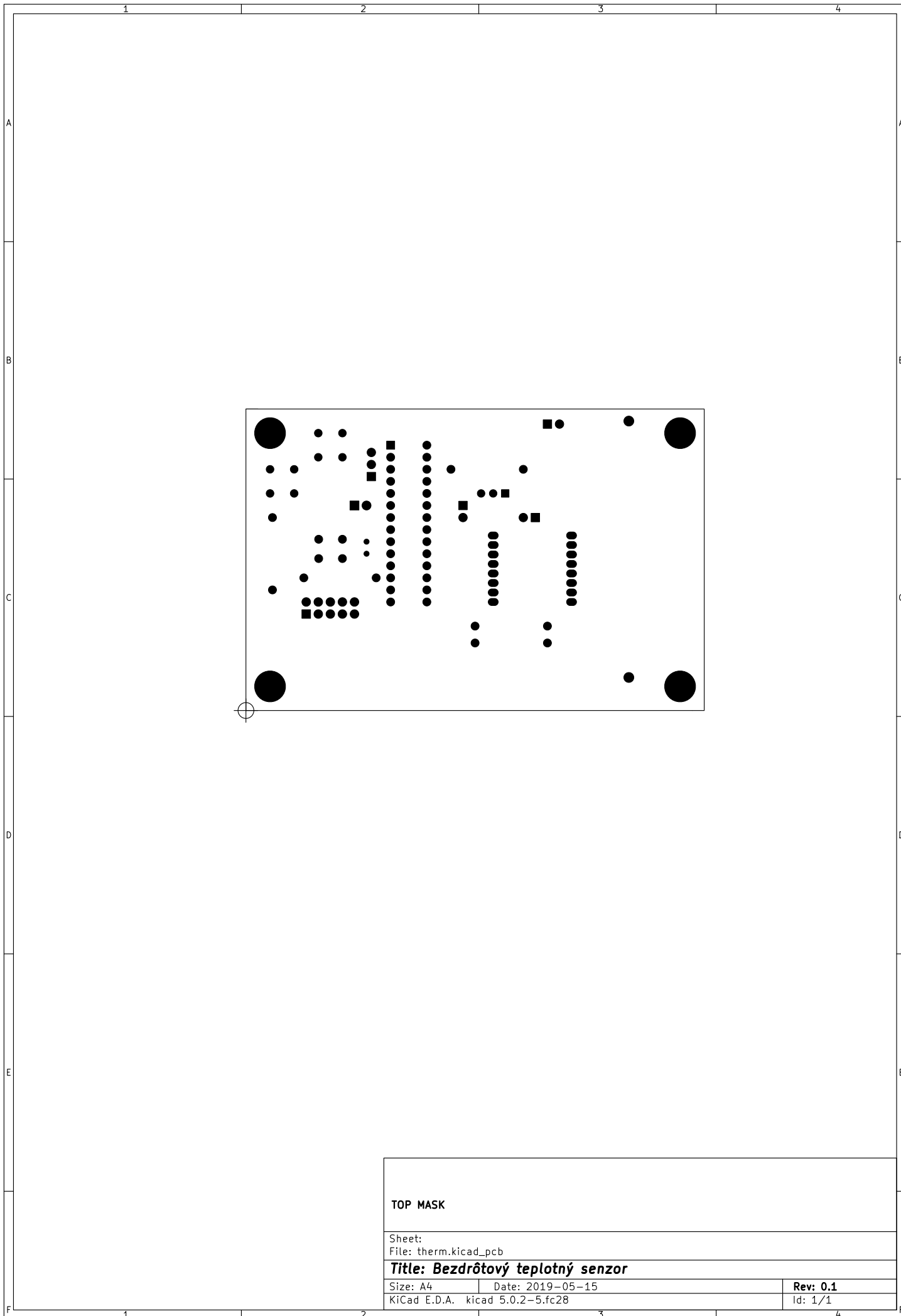
Size: A4

Date: 2019-05-15

Rev: 0.1

KiCad E.D.A. kicad 5.0.2-5.fc28

Id: 1/1



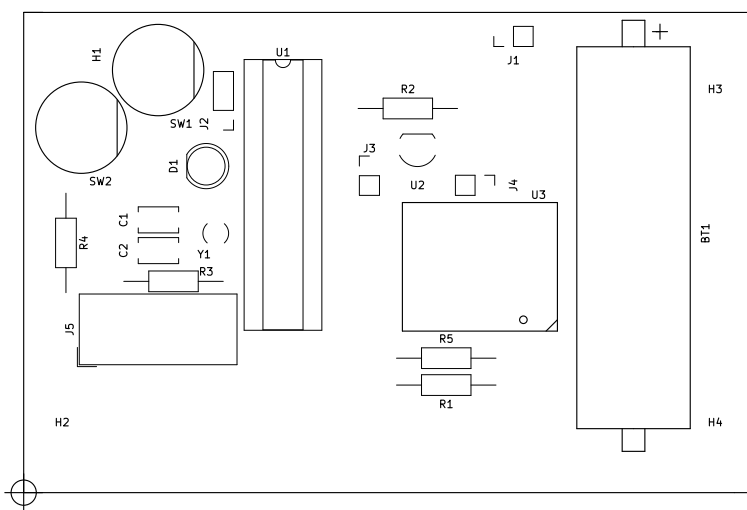
TOP MASK

Sheet:
File: therm.kicad_pcb

Title: Bezdrôtový teplotný senzor

Size: A4 Date: 2019-05-15
KiCad E.D.A. kicad 5.0.2-5.fc28

Rev: 0.1
Id: 1/1



TOP SILKSCREEN

Sheet:
File: therm.kicad_pcb

Title: Bezdrôtový teplotný senzor

Size: A4 Date: 2019-05-15
KiCad E.D.A. kicad 5.0.2-5.fc28

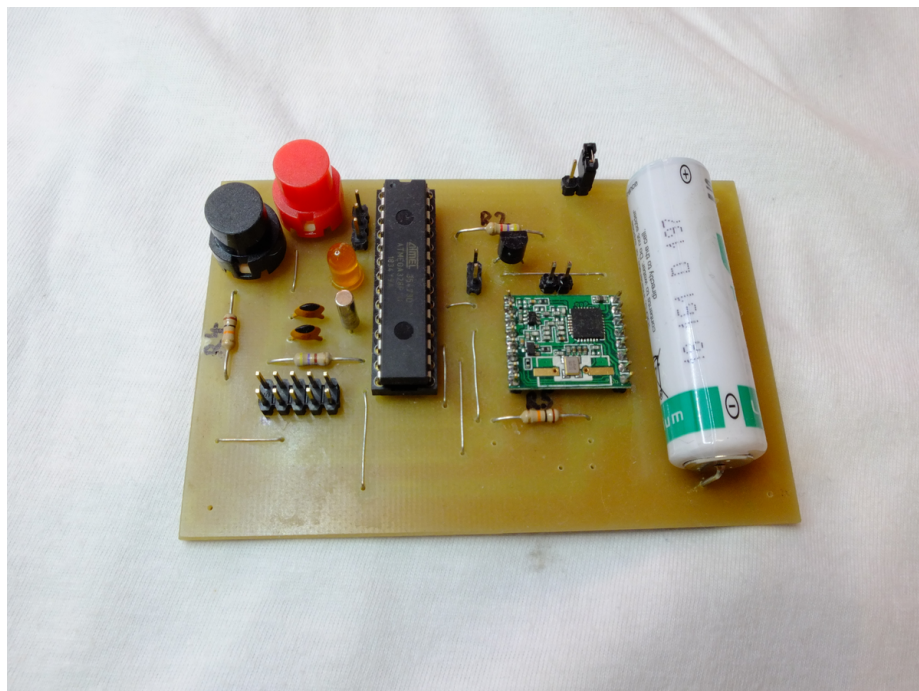
Rev: 0.1
Id: 1/1

D.3 Zoznam komponentov

Tabulka D.1: Zoznam komponentov pre bezdrôtový teplotný senzor

Označenie	Hodnota	Púzdro
BT1	LS14500CNA	
C1	24p	Disc 5.1 mm × 3.2 mm
C2	24p	Disc 5.1 mm × 3.2 mm
D1	LED	5.0 mm
H1	M3	
H2	M3	
H3	M3	
H4	M3	
J1	01x02	Pitch 2.54 mm
J2	01x03	Pitch 2.54 mm
J3	01x02	Pitch 2.54 mm
J4	01x02	Pitch 2.54 mm
J5	AVR-ISP-10	
R1	inf	Axial DIN0207
R2	inf	Axial DIN0207
R3	270	Axial DIN0207
R4	10K	Axial DIN0207
R5	10k	Axial DIN0207
SW1	P-DT6SW	
SW2	P-DT6SW	
U1	ATmega8-16PU	DIP-28
U2	DS18B20	TO-92
U3	RFM69HW	
Y1	32786Hz	AT310

D.4 Obrázky



Obr. D.1: Fotografia plošného spoja bezdrôtového teplomera

Príloha E

Obsah priloženého dátového nosiča

- `hardware/rpi_module` – Elektrická konštrukcia rozširujúceho modulu pre *Raspberry Pi 2*
- `hardware/servomechanism` – Elektrická konštrukcia servomechanizmu ventilu
- `hardware/wireless_thermometer` – Elektrická konštrukcia bezdrôtového teplomera
- `firmware/rpi_module` – Firmvér pre rozširujúci modul pre *Raspberry Pi 2*
- `firmware/servomechanism` – Firmvér pre servomechanizmus ventilu
- `firmware/wireless_thermometer` – Firmvér pre bezdrôtový teplotný senzor
- `software/central_server` – Priečink obsahujúci softvér pre centrálny server
- `software/rpi` – Priečink obsahujúci softvér pre riadiaci počítač *Raspberry Pi 2*
- `software/central_server/ddnsUpdater` – Program *ddnsUpdater* pre centrálny server
- `software/rpi/backend` – Web server poskytujúci REST API pre vykurovací systém
- `software/rpi/frontend` – Webové užívateľské rozhranie pre vykurovací systém
- `software/rpi/controller_can` – Kontrolér pre zbernicu CAN
- `software/rpi/controller_common` – Podporná knižnica zdieľaná naprieč kontrolérmi
- `software/rpi/controller_i2c` – Kontrolér pre zbernicu I²C™
- `software/rpi/controller_main` – Hlavný kontrolér
- `software/rpi/radio` – Kontrolér bezdrôtových zariadení
- `software/rpi/mi2c` – Knižnica pre abstrakciu práce so zbernicou I²C™ pre účely komunikácie s kotlom
- `software/rpi/otherm` – Knižnica pre komunikáciu s kotlom za použitia protokolu OpenTherm™
- `software/rpi/RFM69` – Knižnica pre komunikáciu s bezdrôtovým modulom *RFM69HW*
- `mechanics/servomechanism` – 3D modely servomechanizmu ventilu

- `thesis_source` – Zdrojový text diplomovej práce
- `thesis` – Text diplomovej práce vo formáte PDF
- `README`
- `COPYING`