



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

DEPARTMENT OF COMPUTER SYSTEMS

**SYSTÉM PRO ZÁZNAM A HLÁŠENÍ CHYBOVÝCH
STAVŮ CHLADICÍCH ZAŘÍZENÍ**

SYSTEM FOR RECORDING AND REPORTING FAULT STATES OF COOLING DEVICES

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. FILIP JAŠEK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. VÁCLAV ŠIMEK

BRNO 2021

Zadání diplomové práce



Student: **Jašek Filip, Bc.**
Program: Informační technologie
Obor: Bezpečnost informačních technologií
Název: **Systém pro záznam a hlášení chybových stavů chladicích zařízení**
System for Recording and Reporting Fault States of Cooling Devices
Kategorie: Vestavěné systémy
Zadání:

1. Prostudujte princip činnosti vybraných případů chladicích zařízení a jejich dílčích komponent. Získané poznatky shrňte formou přehledové studie na dané téma.
2. Specifikujte veličiny vhodné pro zjištění možných chybových stavů chladicích zařízení. Identifikujte místa v chladicím okruhu, kde lze tyto veličiny měřit.
3. Navrhněte architekturu hardwarové části systému pro záznam chybových stavů. Definujte jednotlivé funkční celky a jejich vzájemné závislosti.
4. Na základě získaných poznatků zvolte vhodné komponenty pro realizaci jednotlivých částí systému. Zaměřte se zejména na výběr platformy s mikrokontrolérem, senzory pro měření potřebných veličin a nezbytné komunikační prvky.
5. Proveďte realizaci hardwarové části systému. Následně implementujte obslužný firmware, který bude zajišťovat požadovanou funkcionalitu.
6. Implementujte modulární uživatelské rozhraní pro monitorování a správu jednotlivých chladicích zařízení s využitím webových technologií.
7. Vhodnou formou demonstруйте funkčnost vámi realizovaného řešení. Zhodnoťte dosažené výsledky a navrhněte případná rozšíření.

Literatura:

- Dle pokynů vedoucího.

Při obhajobě semestrální části projektu je požadováno:

- Splnění bodů 1 až 4 zadání.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Šimek Václav, Ing.**

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

Datum zadání: 1. listopadu 2020

Datum odevzdání: 19. května 2021

Datum schválení: 30. října 2020

Abstrakt

Práce se zabývá návrhem a tvorbou systému pro záznam a hlášení chybových stavů chladicích zařízení na principu kompresorového tepelného čerpadla. Důležitý prvek koncepce celého monitorovacího systému představuje vestavěná platforma na bázi mikrokontroléru ESP32 pro vyhodnocování údajů naměřených senzory. Dalším klíčovým prvkem je komunikační modul, který pomocí GPRS odesílá naměřená data formou MQTT zpráv obslužné webové aplikaci. Ta přijímá, vyhodnocuje a zaznamenává naměřená data. Uživatelé jsou měřená data a vyhodnocené chybové stavy prezentovány formou tabulek a interaktivních grafů.

Abstract

The thesis deals with the design and creation of a system for recording and reporting error states of cooling devices on the principle of a compressor heat pump. An important element of the concept of the entire monitoring system is a built-in platform based on the ESP32 microcontroller for evaluating data measured by sensors. Another key element is the communication module, which sends measured data via GPRS in the form of MQTT messages to the service web application. The application receives, evaluates and records measured data. Subsequently are measured data and evaluated error states presented to the user in the form of tables and interactive graphs.

Klíčová slova

Tepelná čerpadla, chladicí zařízení, chladicí oběh, hlášení chyb, měření, monitorování, ESP32, GPRS, MQTT, senzory, webová aplikace, Python, Django.

Keywords

Heat Pump, Cooling Devices, Cooling Circuit, Error Reporting, Measuring, Monitoring, ESP32, GPRS, MQTT, Sensors, Web Application, Python, Django.

Citace

JÁŠEK, Filip. *Systém pro záznam a hlášení chybových stavů chladicích zařízení*. Brno, 2021. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Václav Šimek

Systém pro záznam a hlášení chybových stavů chladicích zařízení

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Václava Šimka. Další informace mi poskytli technici z praxe z firmy Chlazení Brno. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Filip Jašek
17. května 2021

Poděkování

Za odborné vedení a cenné rady při vypracování této diplomové práce chci poděkovat svému vedoucímu, panu Ing. Václavu Šimkovi. Rovněž jsem vděčný za technické informace, které mi poskytla firma Chlazení Brno a lidé kolem ní.

Velké poděkování patří i mé rodině a přátelům za jejich upřímnou podporu při vypracování této závěrečné práce i během celého studia na vysoké škole. V neposlední řadě chci poděkovat Ing. Jurajovi Kubišovi, Ing. Viktoru Kovaříkovi, Ing. Janu Bartoni, Bc. Martine Grzybowskej, Bc. Pavlu Dohnalíkovi, Bc. Miroslavovi Kažimírovi, Bc. Patriku Velemu, Bc. Ondřejovi Dubajovi, Ing. Tomáši Goldmanovi a všem ostatním spolustudujícím za čas s nimi strávený na fakultě i mimo ni.

Obsah

1	Úvod	6
2	Chladicí zařízení a jejich principy	7
2.1	Nízkoenergetická chladicí zařízení	7
2.2	Termoelektrická chladicí zařízení	7
2.3	Tepelná čerpadla	8
2.4	Kompresorový chladicí oběh	10
2.4.1	Kompresor	11
2.4.2	Výměníky tepla	12
2.4.3	Redukční ventil	12
2.4.4	Pomocné komponenty	13
2.4.5	Chladivo	13
3	Chybové stavy v systémech chlazení	15
3.1	Současný stav zjišťování problémů	15
3.2	Problémy, které mohou nastat v chladicím oběhu	16
3.3	Veličiny měřené v chladicím oběhu	17
4	Architektura měřicího zařízení	19
4.1	Mikrokontrolér	19
4.2	Senzory	20
4.2.1	Analogové senzory	21
4.2.2	Digitální senzory	22
4.2.3	Volba konkrétních senzorů	23
4.3	Komunikační rozhraní	24
4.4	Shrnutí návrhu	27
5	Technická realizace měřicího zařízení	28
5.1	Hardware	28
5.2	Firmware	30
5.2.1	Komunikační rozhraní	31
5.2.2	Práce s nevolatilní pamětí	31
5.2.3	Měření hodnot a A/D převod	32
5.2.4	Význam jednotlivých MQTT témat	32

6	Webová aplikace pro obsluhu systému	35
6.1	Návrh aplikace	35
6.2	Implementace aplikace	37
6.2.1	Správa zařízení a záznamu hodnot	37
6.2.2	Správa uživatelů	40
6.2.3	Restrikce přístupu a MQTT klient	41
7	Demonstrace funkčnosti a návrhy rozšíření	43
7.1	Testování za provozu chladicího zařízení	44
7.1.1	Korektní chod tepelného čerpadla	44
7.1.2	Simulace špatné funkce kondenzátoru	45
7.1.3	Simulace špatné funkce vstřikovacího ventilu či výparníku	46
7.2	Ověření zápisu hodnot do paměti	46
7.3	Návrhy na rozšíření	47
8	Závěr	50
	Literatura	51
A	Kusovník a schéma zapojení měřicího zařízení	54
B	Snímky některých obrazovek aplikace	56
C	Obsah příloženého CD	61

Seznam obrázků

2.1	Konstrukce jednořadého Peltierova článku [4].	8
2.2	Schéma absorpčního chladicího oběhu [24].	9
2.3	Schéma kompresorového chladicího oběhu [24].	10
2.4	T – s diagram pracovního cyklu kompresorového tepelného čerpadla [13]. . .	11
2.5	Schéma kompresorového chladicího oběhu s vnitřní výměnou tepla.	13
3.1	Servisní manometry pro měření tlaku v oběhu.	15
3.2	Místa měření veličin ve skutečném chladicím oběhu.	18
4.1	Příklad kvantizační chyby. V_{FS} je plný rozsah snímaného napětí, Q je velikost LSB (nejméně významného bitu) [25].	21
4.2	Diagram propojení zařízení v SPI rozhraní.	22
5.1	Schéma napájecího obvodu.	29
5.2	Deska plošných spojů.	29
5.3	Výsledná podoba měřicího zařízení.	30
6.1	Diagram Django MTV architektury.	36
6.2	Blokové schéma aplikace.	38
6.3	Sekvenční diagram vykreslení stránky detailu zařízení.	39
6.4	Karta jednoho chladicího zařízení (výřez snímku obrazovky aplikace). . . .	40
6.5	Graf historie měřených hodnot (výřez snímku obrazovky aplikace).	41
7.1	Proces nasazení měřicího zařízení na tepelné čerpadlo.	43
7.2	Záznam měření při běžném chodu tepelného čerpadla (upravený výřez snímku obrazovky aplikace).	44
7.3	Záznam měření při simulaci špatné funkce kondenzátoru (výřez snímku obrazovky aplikace).	45
7.4	Záznam měření při simulaci podsání kompresoru (výřez snímku obrazovky aplikace).	46
7.5	Modul pro komunikaci po sběrnici RS-485.	48
B.1	Přihlašovací obrazovka	56
B.2	Základní obrazovka – přehled všech sledovaných zařízení.	57
B.3	Detail vybraného zařízení s možností prohlížení historie měření.	58
B.4	Úprava údajů vybraného zařízení	59
B.5	Snímky aplikace zobrazené na mobilním zařízení	60

Seznam použitých zkratek

- ADC** Analog-to-Digital Converter. 21
- AJAX** Asynchronous JavaScript and XML. 38, 39
- API** Application Programming Interface. 24, 32, 36
- APN** Access Point Name. 31
- BPSK** Binary-Phased Shift Keying. 25
- CAN** Controller Area Network. 23
- CDN** Content Delivery Network. 37, 39
- CSS** Cascading Style Sheets. 37
- DOM** Document Object Model. 37, 38
- DTL** Django Template Language. 36, 39
- EEV** Elektronický expanzní ventil. 12
- GPIO** General Purpose Input/Output. 20, 28, 49
- GPRS** General Packet Radio Service. 26–31, 46, 47, 49, 50
- GSM** Global System for Mobile Communication. 26
- GUI** Graphical User Interface. 36
- GWP** Global Warming Potential. 14
- HTML** HyperText Markup Language. 36–39
- HTTP** HyperText Transfer Protocol. 24
- I²C** Inter-Integrated Circuit. 23, 27, 28, 48
- IoT** Internet of Things. 24–26
- IP** Internet Protocol. 26
- JSON** JavaScript Object Notation. 24

LPWAN Low Power Wide Area Network. 25

MQTT Message Queuing Telemetry Transport. 24, 27, 30–32, 35, 37, 39–41, 43, 44, 46–50

MTV Model Template View. 3, 36

MVC Model View Controller. 35, 36

NB-IoT Narrowband Internet of Things. 26

ODP Ozone Depletion Potential. 14

PCM Phase Changing Materials. 7

PLC Programmable Logic Controller. 19

QoS Quality of Service. 24

REST Representational State Transfer. 24

RISC Reduced Instruction Set Computer. 20

RTC Real-Time Clock. 31

SIM Subscriber Identity Module. 31

SMTP Simple Mail Transfer Protocol. 42

SPI Serial Peripheral Interface. 3, 20, 22, 23, 27, 28, 48

SQL Structured Query Language. 36

TCP Transmission Control Protocol. 24

TEV Termostatický expanzní ventil. 12

UART Universal Asynchronous Receiver Transmitter. 23, 48

URL Uniform Resource Locator. 36, 38

USB Universal Serial Bus. 23

VPS Virtual Private Server. 43, 47

WPAN Wireless Personal Area Network. 25

WSGI Web Server Gateway Interface. 37

XML Extensible Markup Language. 24

Kapitola 1

Úvod

Chlazení je řazeno mezi jedno z dvaceti nejdůležitějších technických odvětví dvacátého století¹. Každoročně je téměř 20% celosvětově spotřebované energie vynaloženo na chlazení nebo klimatizaci². Kromě domácích klimatizací nachází chlazení své využití v širokém spektru průmyslových odvětví – chlazení potravin při výrobě či skladování, chlazení strojů, chladicí systémy ve farmaceutických aplikacích a podobně. Stejně jako v ostatních technických odvětvích i v tomto se čím dál častěji objevují požadavky na automatizaci a automatické monitorování. Právě automatickým monitorováním funkce chladicích zařízení se zabývá tato diplomová práce.

První část práce je věnována přehledu chladicích zařízení a principům, na kterých fungují. Zvláštní pozornost je věnována kompresorovým tepelným čerpadlům, neboť se tato práce bude zabývat odhalováním chybových stavů právě těchto typů chladicích zařízení. Následuje výčet možných chybových stavů v kompresorovém chladicím oběhu a veličin relevantních pro jejich odhalení.

Další část práce se zabývá přehledem vhodných technologií a návrhem hardwarové části systému pro záznam a hlášení chybových stavů chladicích zařízení. Důraz je kladen na volbu typu senzorů pro měření tlaku a teploty a také na výběr komunikační platformy pro přenos získaných dat do modulární webové aplikace.

Následující dvě kapitoly jsou věnovány implementaci systému. V první z nich je popsáno řešení měřicího zařízení, tedy zpracování hardwarové části systému, a jeho obslužný firmware. Měřicí zařízení využívá digitální i analogové senzory pro měření teplot a tlaků v chladicím oběhu. Snímaná data za použití bezdrátových technologií odesílá další části systému – obslužné aplikaci. Ta je realizována za pomoci webových technologií a její hlavní účel je ukládat naměřené hodnoty, vyhodnocovat případné chybové stavy chladicího zařízení a vše vhodnou formou prezentovat uživateli.

Závěr práce demonstruje správnou funkci systému na několika měřeních při provozu chladicího zařízení na principu kompresorového tepelného čerpadla. Při měření byly simulovány některé chybové stavy pro otestování schopnosti vyvíjeného systému tyto stavy odhalit a upozornit na ně. Následně je uvedeno několik návrhů na rozšíření systému pro záznam a hlášení chybových stavů chladicích zařízení.

¹Podle: <https://www.nap.edu/catalog/10726/a-century-of-innovation-twenty-engineering-achievements-that-transformed-our>

²Podle: <https://www.sciencedirect.com/science/article/pii/S0140700715001759?via%3Dihub>

Kapitola 2

Chladicí zařízení a jejich principy

Existuje několik základních principů chladicích zařízení. Od primitivních zařízení s přirozeným prouděním tepla přes termoelektrická zařízení až po různá provedení tepelných čerpadel.

Informace pro zpracování následujících dvou podkapitol byly čerpány ze zdrojů: [29, 28, 4].

2.1 Nízkoenergetická chladicí zařízení

Nejjednodušším principem chladicího zařízení je chlazení s přirozeným prouděním tepla (tzv. *Free cooling*). Pro chlazení se využívá zdrojů s již nízkou teplotou – např. studený venkovní vzduch nebo země. Tento princip je použitelný zejména pro chlazení vysokých teplotních hladin (motory, serverovny a podobně).

Mezi nízkoenergetické principy se řadí i tzv. adiabatické odpařovací chlazení, které využívá odběr tepla potřebný k vypařování chladiva (nejčastěji vody) v otevřeném okruhu. U přímého adiabatického chlazení je chladivo rozprášeno do vzduchu, odpařuje se a teplota vzduchu klesá. U nepřímého je chlazenou látkou voda, která je dále využívána pro chlazení pomocí výměníků.

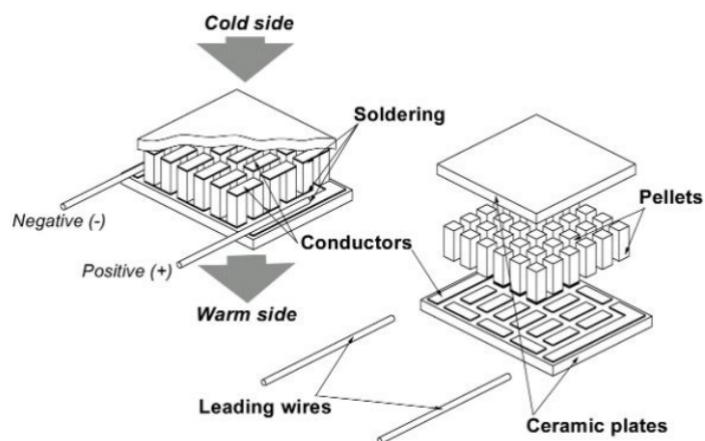
Některé systémy tohoto typu uchovávají získaný chlad ze vzduchu v noci pomocí speciálních materiálů – **Phase Changing Materials (PCM)**. Ochlazený **PCM** materiál je následně využíván pro chlazení během dne.

Přirozeného proudění tepla se někdy využívá i v dalších typech chladicích zařízení – například v zimě, kdy je teplota venkovního vzduchu natolik nízká, že stačí pro dosažení požadovaného chladicího výkonu.

Hlavní nevýhodou je skutečnost, že taková zařízení jsou limitována okolním prostředím. Výhodou je úspora energie a také nulová produkce skleníkových plynů.

2.2 Termoelektrická chladicí zařízení

Dalším typem jsou termoelektrická zařízení, která využívají tzv. Peltierův článek – matice polovodičových termočlánků mezi dvěma keramickými deskami (konstrukci Peltierova článku ukazuje obrázek 2.1). Při průchodu elektrického proudu se jedna strana Peltierova článku ohřívá a druhá ochlazuje (v závislosti na směru průchodu proudu). Peltierovy články je možné skládat do kaskádových termobaterií pro dosažení větších teplotních rozdílů.



Obrázek 2.1: Konstrukce jednořadého Peltierova článku [4].

Chladicí zařízení založená na principu termoelektrických jevů jsou díky jejich malým rozměrům a schopnosti velmi přesně regulovat teplotu využívána nejčastěji pro teplotní stabilizaci elektronických součástek (mikroprocesory, laserové diody, krystalové oscilátory a podobně). Dalším využitím je chlazení v přenosných lednicích či při skladování potravin v letadlech nebo automobilech. Nevýhodou je nízká účinnost. Pro větší chladicí výkony tedy termoelektrické chlazení není vhodné.

2.3 Tepelná čerpadla

Tato podkapitola vychází z informací obsažených ve zdrojích: [21, 7, 22, 13, 24].

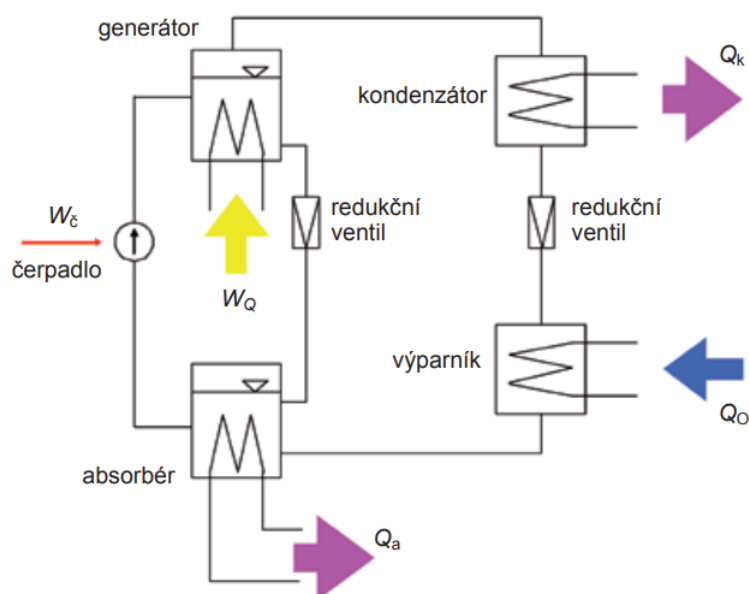
Většina chladicích zařízení od lednicek přes domácí klimatizace až po velké průmyslové chladírny je konstruována na principu tepelného čerpadla využívajícího změny skupenství chladiva. Tepelné čerpadlo je stroj převádějící teplo z nižší teplotní hladiny na vyšší, což zdánlivě porušuje druhý termodynamický zákon, který tvrdí, že chladnější těleso samovolně nepředává teplo tělesu teplejšímu. Předávání tepla však v tomto případě není samovolné, ale při procesu je dodávána vnější energie (elektrická energie pro pohon kompresoru nebo tepelná energie pro sorpční čerpadla). Pak je možné i teplo z relativně chladného tělesa převést – přečerpat – na vyšší teplotní hladinu. Efektivitu tohoto přenosu udává tzv. topný (resp. chladicí) faktor ε_T (v anglické literatuře se používá parametr COP – Coefficient of Performance). Je to poměr užitečně získaného (resp. odebraného) tepla Q_T k množství vynaložené energie W k jeho získání (resp. odebrání), tedy:

$$\varepsilon_T = \frac{|Q_T|}{W} \quad (2.1)$$

Tepelná čerpadla se dále dělí podle typu oběhu. Nejčastější typy oběhu jsou kompresorové, sorpční, proudové a plynné oběhy. Pro konkrétní typy je upraven i vzorec pro výpočet topného faktoru.

Sorpční tepelná čerpadla

Hlavní odlišností sorpčních oběhů je proces, v němž se chladivo za nízkého tlaku pohlcuje absorbentem. Takto vzniklý roztok je následně čerpadlem dopraven do varníku, kde se za současného přívodu tepla chladivo opět vypuzuje. Výsledkem je pára chladiva o vyšším tlaku, než má pohlčované chladivo. Tato pára následně putuje do kondenzátoru, kde je jí odebráno teplo. Zkondenzované chladivo je v redukčním ventilu expandováno do výparníku, kde je mu chlazenou látkou předáváno teplo. Vypařené chladivo je opět absorbováno, a tím je sorpční chladicí oběh uzavřen. Schéma takového okruhu je uvedeno na obrázku 2.2. Energie dodávaná sorpčním čerpadlům je tepelná pro ohřev ve varníku a elektrická pro pohon čerpadla. Tato je však ve většině případů zanedbatelná.



Obrázek 2.2: Schéma absorpčního chladicího oběhu [24].

Existují tři typy sorpčních oběhů: absorpční, adsorpční a resorpční. Při absorpčním oběhu je chladivo i absorbent ve stejném skupenství. Nejčastějšími dvojicemi chladivo – absorbent jsou čpavek (NH_3) – voda a voda – bromid lithný ($LiBr$). Adsorpční oběh pracuje s tuhými látkami, které za nízkého tlaku adsorbují na svůj povrch chladivo. Chladivem je velmi často voda, jejíž molekuly dokážou snadno vytvářet vazby s určitou hydrofobickou solí, která slouží jako adsorbent. Resorpční oběh nahrazuje kondenzaci a vypařování v oběhu opětovnou absorpcí a resorpcí.

Tepelná čerpadla s proudovým oběhem

V tomto typu oběhu je k odsání chladiva z výparníku použitý ejektor (paroproudý kompresor). Pracovní látkou je přehřátá pára získaná v parogenerátoru, která je v trysce ejektoru urychlena a strhává plynné chladivo z výparníku do difuzoru, kde je kinetická energie přeměněna na tlakovou. Zbytek chladicího oběhu je obdobný jako u sorpčních nebo kompresorových čerpadel. Nejčastější je tzv. paroproudé chladicí zařízení, ve kterém je jako hnací látka použita vodní pára a zároveň je pracovní látkou (chladivem) voda.

Tepelná čerpadla se Stirlingovým (plynným) oběhem

Princip tohoto typu tepelného čerpadla není založen na fázové změně pracovní látky, ale na cyklické kompresi a expanzi pracovního plynu (tím je v tomto typu oběhu CO_2 nebo He) uzavřeného v prostoru dvou vzájemně propojených válců. Pracovní prostor válců je omezen pohybujícími se písty hnanými společnou klikovou hřídelí. Při otáčení hřídele dochází ke kompresi pracovního plynu a tím k nárůstu teploty. Pracovní plyn je následně v „kompresním válci“ ochlazen a poté přesunut do „expanzního válce“ kde při expanzi klesá teplota plynu a je tedy možno odebírat teplo z okolí.

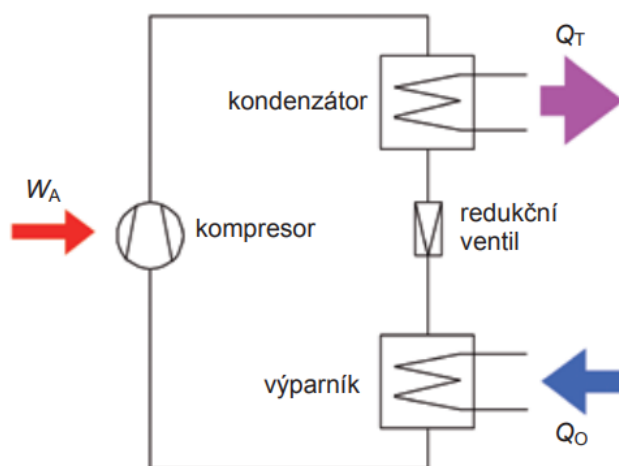
Kompresorová tepelná čerpadla

Zdaleka nejčastějším typem tepelných čerpadel jsou ta s kompresorovým oběhem¹. Chladicí oběh tohoto typu tvoří čtyři nezbytné komponenty: kompresor, kondenzátor, redukční ventil a výparník. Právě kompresorové tepelné čerpadlo je objektem této práce, a proto je v další části uvažován tento typ tepelného čerpadla (chladicího oběhu), není-li explicitně stanoveno jinak.

2.4 Kompresorový chladicí oběh

Při zpracování této podkapitoly byly čerpány informace z [11, 24, 8, 13]

Tento typ chladicího oběhu je tvořen čtyřmi nezbytnými komponenty, často však v praxi bývá doplněn o další pomocné aparáty (pojišťovací ventil, filtr, sběrač chladiva, . . .). Schéma oběhu kompresorového tepelného čerpadla je na obrázku 2.3.

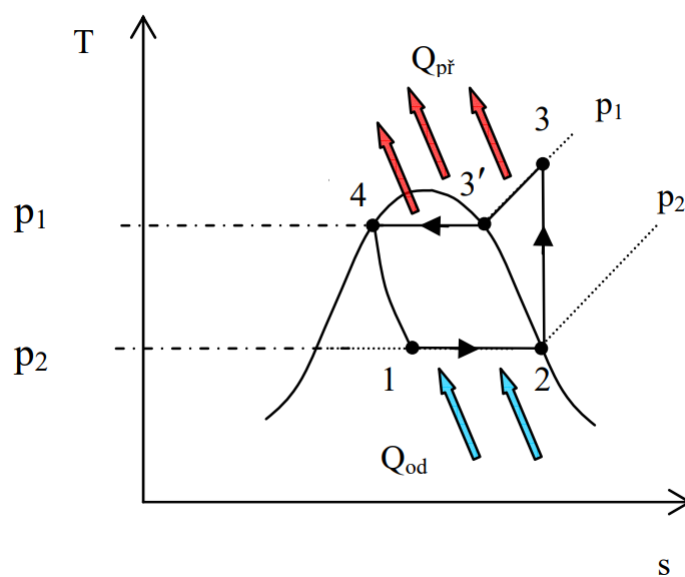


Obrázek 2.3: Schéma kompresorového chladicího oběhu [24].

Pracovní cyklus tepelného čerpadla lze zakreslit tzv. $T - s$ diagramem (vycházejícím z ideálního obráceného Carnotova oběhu) s teplotou na ose Y a entalpií (tepelnou energií uloženou v pracovní látce) na ose X – obrázek 2.4. Páry chladiva z výparníku jsou nasávány

¹Podle: <https://www.sciencedirect.com/science/article/pii/S0140700715001759?via%3Dihub>

kompresorem a stlačovány za zvyšování jejich teploty (*adiabatická komprese* mezi body 2 a 3 v diagramu). Stlačené páry s vyšší teplotou, než je teplota okolí, pokračují do kondenzátoru, kde okolí předávají teplo Q_{pr} (mezi body 3 a 3') a dochází ke kondenzaci (mezi body 3' a 4), takže je chladivo na výstupu z kondenzátoru opět v kapalném skupenství (tato fáze je označována jako *izotermická komprese*). Následně je za použití redukčního (škrtičího) ventilu kapalně chladivo expandováno a jeho teplota klesne (*adiabatická expanze* mezi body 4 a 1). Malá část chladiva se během expanze vypaří, ale většina zůstane v kapalném skupenství. Výsledná směs páry a kapaliny je vedena do výparníku, kde se zbylá kapalina vypaří vlivem přívodu tepla Q_{od} , které je odebráno chlazené látce (*izotermická expanze* mezi body 1 a 2).



Obrázek 2.4: T – s diagram pracovního cyklu kompresorového tepelného čerpadla [13].

2.4.1 Kompresor

Existuje několik typů kompresorů vhodných pro použití v tepelných čerpadlech. Takové kompresory musí splňovat zejména vysoké nároky na těsnost – únik chladiva má za následek provozní problémy, chladivo může nepříznivě působit na obsluhu a životní prostředí, u hořlavých chladiv může dojít ke zvýšení rizika požáru nebo výbuchu. Kompresory proto bývají hermeticky uzavřené v nerozebíratelném plášti s vyvedenými přípojovacími hrdly. Jako chladivové kompresory se používají hlavně pístové (píst periodicky stlačuje kompresní prostor válce), lopatkové (turbokompresory na principu lopatkové turbíny) a spirálové (tzv. scroll kompresory, u kterých je plyn stlačován pomocí dvou spirál valíčků se po sobě za stále se zmenšujícího prostoru mezi nimi), které mají tišší chod, delší životnost a téměř o 20% vyšší účinnost než klasické pístové kompresory.²

²Podle diplomové práce Lucie Koubkové *Vytápění rodinného domu s využitím tepelného čerpadla* [13]

2.4.2 Výměníky tepla

Výměník tepla je zařízení, ve kterém dochází k přenosu tepla mezi dvěma nebo více látkami. V chladicích zařízeních jsou nejčastěji využívány povrchové výměníky, ve kterých jsou látky od sebe odděleny pevnou stěnou, přes kterou kontinuálně přechází teplo. Tyto výměníky lze dále dělit podle konstrukce na trubkové, deskové a jiné (např. kompaktní nebo s tepelnými trubkami). A podle směru proudění látek na souproudé a protiproudé.

Trubkové – například trubka v trubce nebo plášťové výměníky. Pro zvýšení účinnosti přenosu tepla jsou často používány modifikace s žebrováním, které jsou potom označovány jako lamelové výměníky. Hlavními výhodami trubkových výměníků je snadné čištění a široký rozsah provedení, výkonů a hraničních tlaků.

Deskové – dosahují stejného výkonu jako trubkové výměníky při mnohem menších rozměrech, neboť jejich účinnost je při dostatečném proudění vyšší než 90%, což je jejich hlavní výhodou. Další výhodou je odolnost proti zanášení díky vysoké turbulenci proudění. Při nízkých rychlostech proudění ($0,1 \text{ m} \cdot \text{s}^{-1}$) dochází k rapidnímu poklesu účinnosti.

V chladicím oběhu jsou nejméně dva výměníky tepla, a to kondenzátor a výparník. Kondenzátory jsou v typických aplikacích chladicích zařízení lamelové, doplněné o ventilátory pro zvýšení účinnosti. Teplo odváděné v kondenzátoru je možné využít například pro ohřev užitkové vody, a tedy doplnit chladicí oběh o další výměník (například deskový). V zastavěných oblastech je pro eliminaci hluku možné použít pomaloběžný ventilátor.

Typ výměníku, sloužícího jako výparník, závisí na typu chladiva i chlazené látky. V typických průmyslových aplikacích je použitý deskový výměník pro přenos tepla z plynu nebo kapaliny (voda smíchaná s glykolem pro dosažení nižšího bodu tuhnutí). Takto chlazená kapalina je dále využívána pro chlazení vzduchu či jiných látek. Některé aplikace využívají přímého chlazení vzduchu. V tomto případě je výparník konstruován podobně jako kondenzátor – tedy lamelový výměník s ventilátorem.

2.4.3 Redukční ventil

Redukční ventil slouží jako přímý proporcionální regulátor, který řídí přívod kapalného chladiva (nástřík) do výparníku na základě stupně přehřátí par odsávaných z výparníku, a tím přímo určuje chladicí výkon. Existují dvě hlavní konstrukce redukčních ventilů, a to termostatický a elektronický. Vstupní veličinou je v obou případech teplota (případně tlak) a regulovanou veličinou je množství chladiva vstříkovaného do výparníku.

Termostatický expanzní ventil (TEV) pracuje na principu tepelné roztažnosti látek. Jeho součástí je tykavka připojená kapilárou a umístěná na výstup výparníku. Uvnitř tykavky je plyn s podobnými vlastnostmi, jako má chladivo. Při zvýšení teploty par odsávaných z výparníku dojde k roztažení plynu v tykavce a zvýšení tlaku způsobí otevření ventilu. Jakmile teplota nasávaných par poklesne, poklesne i tlak plynu v tykavce a dojde k přivření ventilu působením vratné pružiny. Pro každé chladivo je potřeba použít jemu odpovídající tykavku, jinak by redukční ventil nefungoval správně.

Elektronický expanzní ventil (EEV) pomocí senzoru měří teplotu (případně tlak) na výstupu z výparníku a na základě hodnot z tohoto senzoru krokovým motorem reguluje otevření ventilu.

2.4.4 Pomocné komponenty

Pro spolehlivou a bezpečnou funkci chladicího oběhu v širokém rozsahu pracovních podmínek je využíváno několik pomocných komponentů.

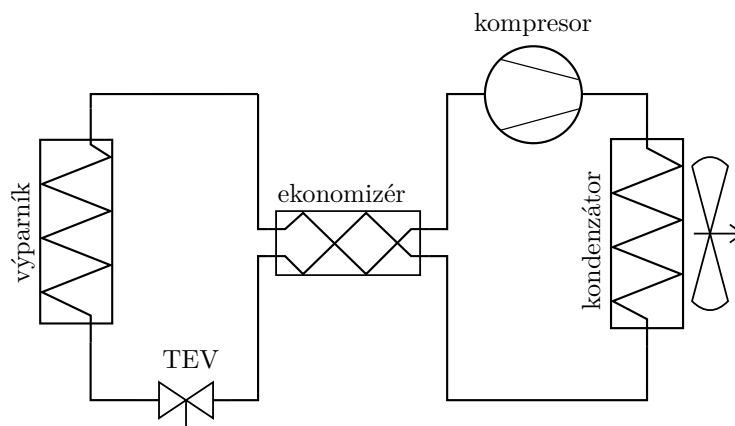
Jedním takovým prvkem je sběrač chladiva – tlaková nádoba sloužící jako zásoba chladiva v chladicím oběhu pro vyrovnání rozdílů v náplni potrubí a jednotlivých prvků při změně pracovních podmínek. Sběrač chladiva je vhodné opatřit pojišťovacím ventilem pro předejití možné destrukci zařízení při překročení nastaveného hraničního tlaku.

Dalším pomocným prvkem jsou filtry pro zachycení nečistot, které vznikají při nedostatečném čištění před a po montáži. Filtry jsou zařazovány před nejvíce náchylné části oběhu, což je redukční ventil a kompresor. V aplikacích s uhlovodíkovými chladivými je filtr kombinován s dehydrátorem pro odstranění nežádoucí vlhkosti.

Do kapalinové části (před redukční ventil) bývá řazeno průhledítko pro vizuální kontrolu proudění chladiva. Průhledítko může být také opatřeno indikátorem vlhkosti.

Při vypnutí kompresoru by se mohlo vlivem netěsnosti expanzního ventilu do výparníku postupně dostat kapalně chladivo v takovém množství, že by se po startu kompresoru nestačilo odpařit, a tak by kompresor nasál nestlačitelné kapalně chladivo, což by vedlo k jeho poškození. Z tohoto důvodu je před expanzní ventil umístován ještě elektromagnetický ventil (EMV), který v okamžiku vypnutí kompresoru uzavře potrubí s kapalným chladivem.

V některých aplikacích je možné použít tzv. ekonomizér, což je přídavný výměník tepla zapojený dle obrázku 2.5 mezi nízkotlakou a vysokotlakou částí oběhu. Ekonomizér přebytečným teplem neodvedeným z kondenzátoru předeřívá páry nasávané kompresorem, u kterých je výhodné zvyšovat jejich teplotu. Tím je zvýšena účinnost chladicího zařízení. Oběh s ekonomizérem bývá označován jako oběh s vnitřní výměnou tepla.



Obrázek 2.5: Schéma kompresorového chladicího oběhu s vnitřní výměnou tepla.

2.4.5 Chladivo

Při zpracování přehledu chladiv bylo vycházeno zejména ze zdroje [17] (kromě zdrojů zmíněných v nadřazené podkapitole).

Druh chladiva, tedy chemické látky cirkulující v hermeticky uzavřeném chladicím oběhu, ovlivňuje chování tohoto oběhu a volbu jeho komponentů i použitelných materiálů. Chladiva jsou označována kódem, který začíná velkým R (z anglického *refrigerant*) následovaným číslem a volitelně i písmenem na konci. Kromě termodynamických vlastností chladiv se udá-

vají ještě dva významné parametry: **Ozone Depletion Potential (ODP)** udávající relativní poškození ozonové vrstvy daným chladivem a **Global Warming Potential (GWP)**, což je relevantní u tzv. skleníkových plynů a je měřítkem toho, kolikrát více tepla zachytí skleníkový plyn v atmosféře ve vztahu k oxidu uhličitému, jehož **GWP** koeficient je standardizován na hodnotu 1.

Chladiva prošla (a stále procházejí) velkým vývojem pro nalezení chladiva vhodného nejen svými termodynamickými vlastnostmi, ale také z hlediska toxicity a dopadů na životní prostředí. V minulosti byly používány jak látky velmi toxické (např. oxid siřičitý), tak látky, které jsou dodnes excelentními chladivy (např. amoniak, propan nebo oxid uhličitý). Později se pro chlazení začala hojně využívat skupina chladiv CFC – označovány jako tvrdé freony (např. R12, R13, R502), které jsou dnes již zakázané kvůli vysokým hodnotám **ODP**, a skupina HCFC – měkké freony (např. R22), které sice nepoškozují ozonovou vrstvu, ale mají velmi vysoký koeficient **GWP** a kvůli tomu dnes již také nejsou v nových zařízeních používány. Náhradou za ně je skupina HFC – fluorované uhlovodíky (R407, R449, R134a), které mají nulové **ODP**, ale stále celkem vysoké **GWP** (do nových zařízení se dnes nesmějí používat chladiva s **GWP** větším než 2500), a proto je snaha o jejich nahrazení jinými chladivy.

Bohužel zatím neznáme látku, která by vyhověla všem požadavkům na ideální chladivo (s dobrými termodynamickými vlastnostmi, netoxické, nehořlavé, s nulovým **ODP** a **GWP**, ...). Postupně se vracejí původní chladiva jako amoniak, propan nebo oxid uhličitý pro jejich nízké **ODP** a **GWP**. Zároveň je prováděn výzkum směsí chladiv s cílem nalézt co nejlepší kompromis mezi požadovanými vlastnostmi, ekologickými a ekonomickými aspekty.

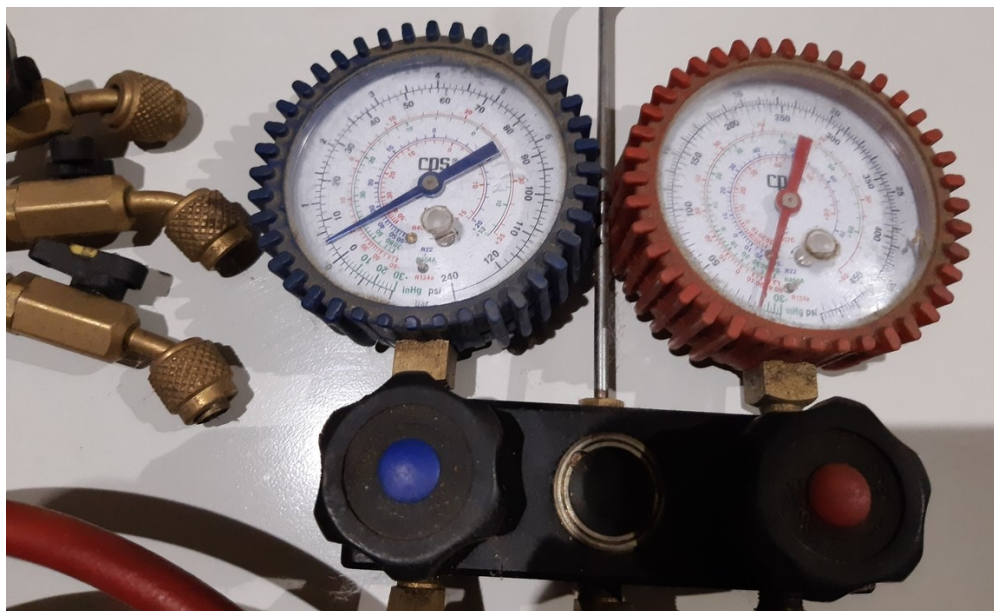
Kapitola 3

Chybové stavy v systémech chlazení

Nejdůležitějšími fyzikálními veličinami jsou v chladicím oběhu teplota a tlak. Sledováním těchto veličin lze zjišťovat některé problémy, které mohou v chladicím oběhu nastat a které ovlivňují funkci chladicího zařízení.

3.1 Současný stav zjišťování problémů

Při zjišťování problémů u chladicích zařízení ve firmě *Chlazení Brno* jsou využívány analogová měřidla tlaku (obrázek 3.1), která se připojují na výstupy připravené v chladicím okruhu při výrobě zařízení. Tlak se měří na vstupu a na výstupu kompresoru – technik má tedy informace o tlaku v nízkotlaké i vysokotlaké části chladicího oběhu. Pro zjišťování problémů mnohdy nestačí znát pouze hodnoty těchto veličin, ale je důležité sledovat jejich vývoj v čase a jejich vzájemné rozdíly při chodu chladicího zařízení.



Obrázek 3.1: Servisní manometry pro měření tlaku v oběhu.

Tlak a teplota na sobě vzájemně závisí – pro jednotlivá chladiva se dá přibližně určit hodnota tlaku měřením teploty a naopak. Proto mají analogová měřidla hlavní stupnici tlakovou a několik orientačních teplotních stupnic pro často používaná chladiva.

Některé komponenty v chladicím okruhu jsou vybaveny ochrannými – většinou binárními – prvky (např. čidlo pro kontrolu hladiny oleje v kompresoru, nebo presostaty hlídající mezní hodnoty tlaků).

3.2 Problémy, které mohou nastat v chladicím oběhu

Funkce chladicího zařízení závisí na správném fungování jeho prvků a jejich vzájemné souhře. U jednotlivých prvků mohou nastat následující problémy.

Kritické problémy

Hlavní součástí chladicího oběhu je kompresor. Jeho porucha znamená konec funkce chladicího zařízení a je tedy jednoduché ji zjistit. Jasným ukazatelem poruchy kompresoru je malý rozdíl tlaku na jeho vstupu a výstupu.

Únik chladiva je další poruchou, která zamezuje funkci chladicího zařízení. Navíc s sebou většinou nese další problémy v závislosti na typu chladiva, jako je např. zvýšené nebezpečí vzniku požáru nebo nepříznivé ovlivňování životního prostředí. Únik chladiva lze identifikovat poklesem tlaku jak na vstupu, tak na výstupu kompresoru.

Může také nastat chyba expanzního ventilu spočívající například v jeho zaseknutí. Taková porucha se neobjevuje příliš často (i díky předřazenému filtru). Může se však stát, že již při montáži nebo při špatném zacházení je poškozena kapilára termostatického ventilu, a ventil tak nepracuje správně. Špatná funkce expanzního ventilu může být odhalena sledováním rozdílů tlaků v nízkotlaké a vysokotlaké části chladicího oběhu (tedy před a za kompresorem).

Špatná funkce kondenzátoru

Nedostatečný odvod tepla v kondenzátoru má vliv na celkovou účinnost chladicího zařízení. Je žádoucí, aby na výstupu z kondenzátoru byla teplota kapalného chladiva co nejnižší.

Ukazatelem špatné efektivity předávání tepla v kondenzátoru je zvyšující se tlak na výstupu kompresoru (tedy na vstupu kondenzátoru). U kondenzátorů lamelových se může prostor mezi lamelami zanést prachem a nečistotami, kondenzace potom probíhá pomaleji, což má za následek zvyšování tlaku, a tedy i teploty na vstupu a výstupu kondenzátoru. Kondenzátory jsou často pro dosažení požadované účinnosti vybaveny ventilátorem, jehož porucha rovněž způsobuje pokles účinnosti kondenzátoru a dá se diagnostikovat sledováním tlaku.

Pro zjištění špatné funkce kondenzátoru je tedy důležité snímat tlak na výstupu kompresoru. Dále je důležité sledovat rozdíl teploty na vstupu a výstupu z kondenzátoru. Účinnost kondenzátoru je závislá na okolní teplotě. Aby nebyla falešně vyhodnocena špatná funkce kondenzátoru, je vhodné mít informace také o okolní teplotě.

Špatná funkce výparníku

Podobně jako nedostatečný odvod tepla v kondenzátoru má na celkovou účinnost zásadní vliv i špatný přenos tepla ve výparníku. Výparníky v chladicích zařízeních jsou konstruovány jako lamelové tepelné výměníky – pro přímé chlazení vzduchu, nebo jako deskové výměníky pro chlazení tekutiny.

U výparníků pro přímé chlazení vzduchu může docházet ke stejným problémům jako u kondenzátorů, neboť jsou v tomto případě konstruovány na stejném principu. Protože se však u chladicích zařízení na výparníku objevují nízké teploty, může se navíc žebrovaní ucpávat namrzající vlhkostí chlazeného vzduchu. Tyto výparníky proto při namrzání cyklicky přecházejí do stavu odmrazování, kdy jsou na krátký čas naopak zahřáty, aby námraza roztála a odtekla.

Výparník, konstruovaný jako deskový výměník, může při nedostatečném průtoku chlazené kapaliny zamrznout, což zamezí dalšímu přenosu tepla a vlivem tepelné roztažnosti může dojít k destrukci výparníku, potažmo až k úniku chladiva.

Zamrznutí nebo jiné ucpání výparníku se projevuje nízkým tlakem na vstupu kompresoru (tzv. podsáním). Pro odhalení chyby výparníku je tedy důležité snímat tlak na vstupu kompresoru. Také je vhodné sledovat rozdíl teploty chladiva na vstupu a na výstupu výparníku.

Ucpání filtru

Před expanzním ventilem je typicky umístěn filtr (případně filtrdehydrátor), který chrání ventil před nečistotami, které se mohou v chladicím oběhu objevit. Špatná průchodnost tohoto filtru snižuje množství chladiva, které je expandováno do výparníku, a má tedy negativní vliv na účinnost chladicího zařízení.

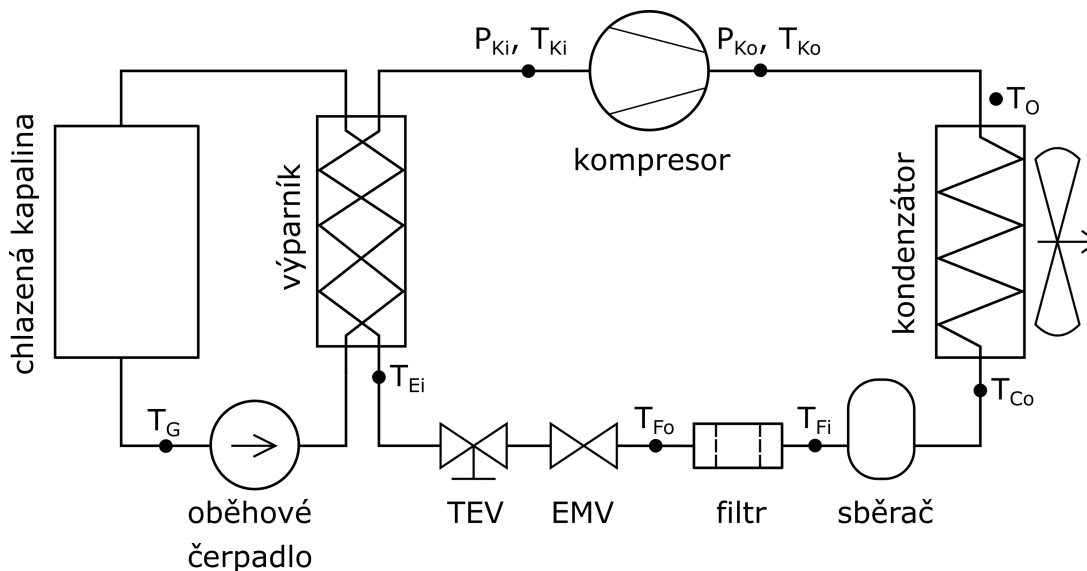
Ucpání filtru se dá zjistit rozdílem tlaku na vstupu a na výstupu filtru, který by měl být ideálně nulový. Pokud je proudění omezeno, je na vstupu filtru větší tlak než na výstupu. Měření tlaku vyžaduje zapojení čidla do chladicího oběhu, což je nepraktické a drahé. Díky skutečnosti, že na tlaku závisí teplota, je možné rozdíl tlaku odvodit měřením rozdílu teplot na vstupu a výstupu filtru (zanesený filtr v tu chvíli působí podobně jako expanzní ventil). Teplotu lze měřit i z venku chladicího oběhu, což je výrazně jednodušší.

3.3 Veličiny měřené v chladicím oběhu

Místa pro měření teplot a tlaků pro odhalování chybových stavů jsou vyznačena ve schématu skutečného kompresorového chladicího oběhu (který obsahuje i pomocné komponenty) na obrázku 3.2. Veličiny jsou indexovány tak, že první písmeno je zkratka komponentu oběhu a druhé písmeno udává, v jakém místě komponentu je hodnota měřena (Např. T_{Fo} je teplota měřená na výstupu z filtru). Kromě takto označených hodnot je ve schématu označena ještě teplota okolí T_O a teplota chlazené kapaliny T_G (většinou jde o směs vody a glykolu).

V chladicím oběhu budou pro odhalování chyb a sledování stavu chladicího zařízení umístěny dva senzory tlaku snímající tlak P_{Ki} a P_{Ko} , tedy tlak na vstupu kompresoru v nízkotlaké části oběhu a na výstupu kompresoru ve vysokotlaké části. Zároveň budou v těchto místech měřeny teploty T_{Ki} a T_{Ko} . Jelikož je kondenzátor přímo připojený na výstup kompresoru, za teplotu vstupu kondenzátoru je možné považovat T_{Ko} , neboť se bude od skutečné teploty na vstupu kondenzátoru lišit jen zanedbatelným poklesem (závisejícím na délce potrubí mezi kompresorem a kondenzátorem). Další měřená veličina bude teplota

na výstupu kondenzátoru T_{Co} . Pro odhalování poruch filtru je potřeba měřit teploty co nejbližně u jeho vstupu a výstupu, aby byly co nejméně zkreslené, bude tedy měřena teplota T_{Fo} na výstupu a teplota T_{Fi} na vstupu, i když se bude od teploty na výstupu kondenzátoru lišit jen minimálně (v případě oběhu bez vnitřní výměny tepla). Poslední teplotou měřenou v chladicím oběhu je teplota chladiva na vstupu do výparníku T_{Ei} . Teplota na výstupu výparníku bude v případě oběhu bez vnitřní výměny tepla téměř stejná jako teplota na vstupu kompresoru.



Obrázek 3.2: Místa měření veličin ve skutečném chladicím oběhu.

Kapitola 4

Architektura měřicího zařízení

Hlavní myšlenkou navrhovaného měřicího zařízení je mít řešení co nejvíce samostatné, nezávislé na řídicích prvcích chladicího zařízení. Je tedy potřeba zvolit vlastní senzory pro měření tlaků a teplot v chladicím oběhu. Dále je potřeba vyřešit komunikaci se serverem, na kterém budou data zaznamenávána a zobrazována. V neposlední řadě je třeba zvolit kontrolér, který bude vyčítat data ze senzorů a odesílat je pro další zpracování.

4.1 Mikrokontrolér

Informace pro zpracování této podkapitoly byly čerpány ze zdrojů [5, 14].

Pro vzdálené měření hodnot a další automatizace průmyslových zařízení obecně bývají používány tzv. **Programmable Logic Controller (PLC)**. Jsou to kontroléry vybavené několika digitálními a také analogovými vstupy a výstupy. Většina implementací disponuje podporou komunikace přes standardní rozhraní, jako je Ethernet nebo RS-485. Některé PLC je možné rozšiřovat pomocí modulů. Programování těchto kontrolérů často spočívá v jiných principech, než je sekvenčně definovaný zdrojový kód. Využívá se např. tzv. ladder logic nebo grafické jazyky s propojováním bloků. Samotné kontroléry je nutné doplnit vhodnými senzory a také – pokud je to vyžadováno – externím rozhraním pro jiný typ komunikace, než je vestavěná možnost.

Kvůli ceně programovatelných logických kontrolérů, která se pohybuje v tisících korun českých (v závislosti na počtu vstupů a vestavěných komunikačních rozhraní), se může takové řešení prodražit. Jako alternativa mohou dnes sloužit rychle se vyvíjející mikrokontroléry, jejichž cena se pohybuje i o dva řády níže – v desítkách českých korun.

Existuje velké množství levných mikrokontrolérů, které lze k tomuto účelu používat. Mezi ty známější patří například Arduino¹ nebo mikrokontroléry od společnosti Espressif². Většina je dostupná v tzv. development kitech (desky plošných spojů s mikrokontrolérem, programátorem a dalšími součástkami). Mikrokontroléry jsou nejčastěji programovány jazykem C, nově se však začínají rozšiřovat i development kity s jinými možnostmi programování, například MicroPythonem – odnože skriptovacího jazyka Python 3 právě pro mikrokontroléry.

¹Arduino home: <https://www.arduino.cc/>

²Espressif: <https://www.espressif.com/en>

ESP32

Pro účely této práce se vhodnou volbou jeví mikrokontrolér ESP32 právě díky nízké ceně a relativně velké komunitě vývojářů, kteří tyto mikrokontroléry používají. ESP32 jsou oblíbenými mikrokontroléry i díky své kompatibilitě s mnoha původními Arduino knihovnamí. Vnitřní logika tohoto mikrokontroléru pracuje na napěťové úrovni 3,3 V. ESP32 mikrokontroléry mají vyvedeno 34 **General Purpose Input/Output (GPIO)** pinů, některé z nich analogové – využívající interní dvanáctibitový A/D převodník.

Konkrétně model ESP32-WROVER-B je vybaven dvěma mikroprocesory Xtensa[®] LX6 s 32-bit architekturou a redukovanou instrukční sadou (**RISC**). SRAM paměť pro instrukce a data má kapacitu 520 KB a je umístěná přímo na čipu. Tento model integruje externí flash paměť o velikosti 4 MB, která je fyzicky umístěná přímo na desce ESP32-WROVER-B a je připojena přes **SPI** sběrnici. Procesor podporuje připojení i více externích flash pamětí přes **SPI** sběrnici. Více informací je možné dohledat v technické dokumentaci³.

4.2 Senzory

Při zpracování podkapitol o senzorech byly čerpány informace z knihy *Measurement, Instrumentation, and Sensors Handbook: Two-Volume Set* [26], z článku [25] a z přednášek magisterského předmětu SEM – Senzory a měření.

Senzor je zařízení, které snímá sledovanou fyzikální, chemickou nebo biologickou veličinu a dle určitého fyzikálního principu převodu ji transformuje na veličinu výstupní. Výstupní veličina je často elektricky kvantifikovatelná, neboť elektrické signály je jednoduché zpracovat, ukládat, přenášet a zobrazovat. Stav sledované veličiny snímá citlivá část senzoru označovaná jako čidlo a zpracovává jej vyhodnocovací obvod senzoru.

Elektrická hodnota může být vyjádřena přenosovou funkcí – matematickým modelem mezi signálem senzoru a snímanou veličinou. Ve spojitých systémech může být přenosová funkce lineární či nelineární. Lineární vztah lze vyjádřit rovnicí 4.1:

$$y = a \cdot x + b \tag{4.1}$$

kde:

y je výstupní elektrický signál,

x je snímaná fyzikální veličina,

a je sklon, též označovaný jako citlivost a

b je posun osy y udávající velikost výstupního signálu pro nulovou vstupní hodnotu.

V ideálním případě by tento vztah měl být konzistentní, bez chyb nebo nelinearit. Avšak v reálném měření se objevuje mnoho zdrojů chyb, se kterými je nutné počítat (např. nedokonalosti mechanických či elektrických komponentů, změna chování komponentů – vlivem stárnutí či změny okolních podmínek, ...).

Jakmile jsou získána elektrická data měření, je možné je několika způsoby interpretovat jakožto užitečnou informaci. Analýza dat může obsahovat statistické metody, jako je prokládání křivkou nebo výběrem podmnožiny dat.

Rozlišujeme senzory analogové, digitální, nebo kombinaci analogových a digitálních. Mnoho dnes produkovaných zařízení je digitálních, a to pro výhody, které nabízejí. Snímající části většiny zařízení jsou však stále analogové, neboť jejich čidla produkují analogové

³Datasheet ESP32-WROVER-B: https://www.espressif.com/sites/default/files/documentation/esp32-wrover-b_datasheet_en.pdf

hodnoty. Tyto jsou dále zpracovány vnitřními obvody a převedeny do digitálních hodnot. Vytvoří se však i senzory pracující čistě na základě digitálních principů. Pro příklad lze uvést inteligentní senzory, které mají čidlo integrované přímo na čipu s mikroelektronickou částí pro zpracování dat.

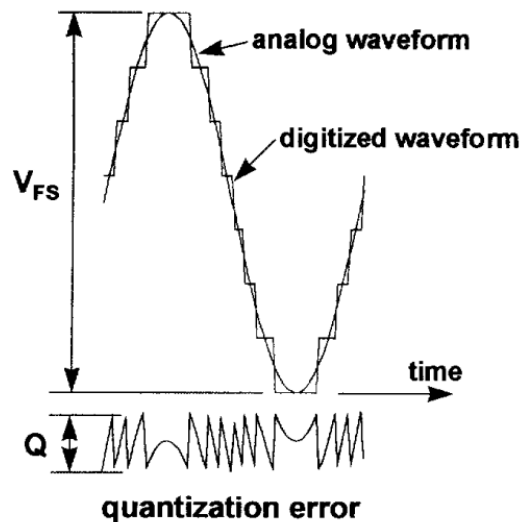
4.2.1 Analogové senzory

Tyto senzory jsou charakteristické spojitými signály. Čistě analogové systémy měří, ukládají, přenášejí i zobrazují data v analogové formě. Jako všechny systémy přenášející analogový signál i senzory se musí potýkat s nechtěnými signály vzniklými různými formami rušení. I proto je signál často upravován mnoha funkčními bloky, jako jsou zesilovače, filtry, modulátory, oscilátory, buffery a další.

Informace odrážející stav měřeného systému je kódována velikostí amplitudy signálu, nebo jeho frekvencí, případně kombinací těchto dvou vlastností. Pro další zpracování dat ze senzoru mikrokontrolérem je nutné analogový signál převést na digitální. K tomu slouží Analogově-Digitální převodník (ADC).

Převod spojitého analogového signálu do digitálního diskretního je prováděn ve třech fázích: vzorkování, kvantování a kódování. Ačkoliv teoreticky lze spojitý analogový signál přibližovat a pozorovat tak nekonečně malé detaily, kvůli omezené výpočetní kapacitě procesorů to v praxi v reálném čase není možné. Je potřeba zvolit vzorkovací frekvenci, a to takovou, aby platil Nyquistův vzorkovací teorém: „Počet vzorků za sekundu musí být alespoň dvakrát větší, než je nejvyšší frekvence ve spojitém signálu“. Jinak by mohlo dojít k úplnému a nenávratnému zkreslení signálu kvůli tzv. aliasingu (podle Shannonova teorému) [25].

S vhodnou frekvencí tedy rozdělíme – navzorkujeme – signál na časové úseky (na ose x). Dalším krokem je kvantizace, která udává rozlišení navzorkovaných bodů signálu. S rostoucím počtem bitů klesá kvantizační chyba (patrná z obrázku 4.1). Poslední fází je kódování, kdy jsou kvantované hodnoty převedeny do binárních čísel pro další digitální zpracování.



Obrázek 4.1: Příklad kvantizační chyby. V_{FS} je plný rozsah snímaného napětí, Q je velikost LSB (nejméně významného bitu) [25].

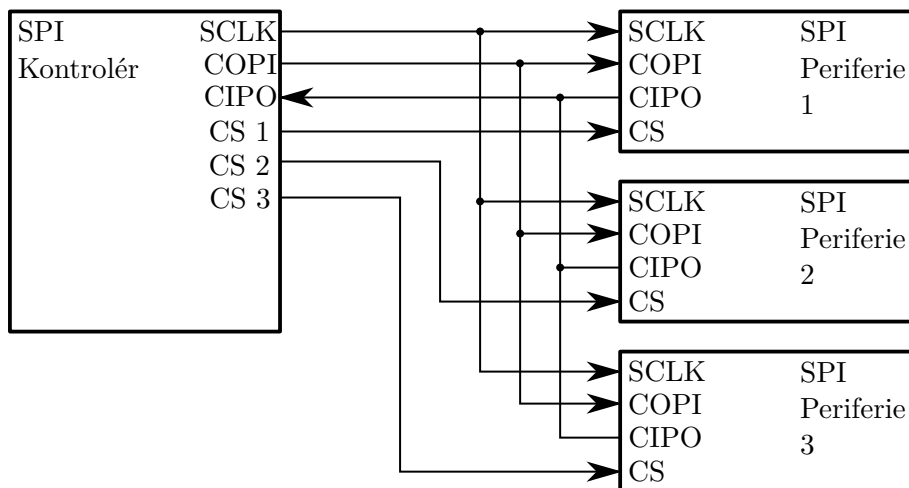
A/D převodníky se dělí na tři základní typy: komparační, kompenzační a integrační. Převodem analogového signálu na digitální je ztraceno velké množství informace. Tato ztráta je nevyhnutelná, neboť pro další zpracování je potřeba mít data ze senzoru v digitální podobě. Granularita převodu se však volí až v obslužném mikrokontroléru a je možné ji nastavit podle požadavků konkrétní aplikace.

4.2.2 Digitální senzory

V digitálních senzorech jsou data obvykle získána analogovým čidlem, avšak A/D převod je uskutečněn přímo v senzoru, a ten pak dále předává již zdigitalizovaná data. Digitální zařízení jsou s dalšími zařízeními propojována pomocí sběrnic. Přenos dat probíhá po sběrnicích podle standardizovaných protokolů. Informace k jejich následujícímu přehledu byly čerpány ze zdrojů: [15, 10, 3].

SPI

Sběrnice **Serial Peripheral Interface (SPI)** je tzv. single-controller rozhraní, tedy jedno centrální zařízení iniciuje komunikaci s připojenými periferiemi. **SPI** obsahuje čtyři vodiče: SCLK – hodinový signál, CS – Chip Select, tedy adresa zařízení, se kterým bude kontrolér komunikovat, a dva vodiče pro samotný přenos dat: COPI – Controller Out / Peripheral In a CIPO – Controller In / Peripheral Out⁴. Komunikace tedy probíhá obousměrně (full-duplex). Pokud chce kontrolér poslat (resp. vyžádat si) data jednomu z připojených zařízení, nastaví jemu odpovídající CS pin na logickou nulu a aktivuje hodinovou frekvenci tikající na SCLK vodiči. Kontrolér následně generuje signál na COPI vodiči, zatímco poslouchá na vodiči CIPO. Zapojení několika **SPI** zařízení je uvedeno na obrázku 4.2. Nevýhodou **SPI** sběrnice je s rostoucím počtem komunikujících zařízení také rostoucí počet vodičů (pro adresaci).



Obrázek 4.2: Diagram propojení zařízení v **SPI** rozhraní.

⁴V literatuře jsou **SPI** vodiče někdy značeny jako SS – Slave Select, MOSI – Master Out / Slave In a MISO – Master In / Slave Out. Dnes se však kvůli politické korektnosti doporučuje používat nové označení.

I²C

Inter-Integrated Circuit (I²C) (je na rozdíl od **SPI**) tzv. multi-controller rozhraní využívající (kromě napájení) pouze dva signálové vodiče: SDA – serial data a SCL – serial clock. Na tyto dva vodiče může být napojeno až 128 kontrolerů anebo periférií. Ty spolu komunikují podle jednoduchého protokolu, který obsahuje sedmibitovou unikátní adresu zařízení, data dělená do osmibitových bloků – Bytů a několik kontrolních bitů pro start, ukončení, směr a potvrzování komunikace. Sběrnice **I²C** obsahuje jen jeden datový vodič a jedná se tedy o jednosměrnou komunikaci (half-duplex). Rychlost přenosu dat musí být zvolena buď 100 kb/s (standard mode), 400 kb/s (fast mode) nebo 3,4 Mb/s (high speed mode). Některé varianty umožňují i rychlosti 10 kb/s (low speed mode) a 1 Mb/s (fast mode +). Právě rychlost je jedna z nejvýznamnějších nevýhod tohoto protokolu.

One-Wire

Další sériovou sběrnici populární převážně u levných jednoúčelových digitálních senzorů je One-Wire. Svým principem je podobný **I²C**, komunikace však může probíhat obousměrně. One-Wire posílá i přijímá data na jednom signálovém vodiči, navíc potřebuje minimálně jeden další vodič s nulovým potenciálem. Energie se v tomto tzv. parazitním módu získává z datového vodiče a pro potřebu odchozí komunikace je uchovávána ve vnitřním kondenzátoru periferie. Klasický mód zapojení přidává navíc vodič pro napájení periferního zařízení. Na jednom datovém vodiči může být připojených několik zařízení, která jsou adresována unikátním sériovým číslem.

UART

Tato zkratka vychází z anglického **Universal Asynchronous Receiver Transmitter** a jedná se o komunikační standard mezi dvěma zařízeními, využívající pouze dva vodiče: Tx (pro vysílání dat) a Rx (pro příjem dat). Slovo asynchronní v názvu znamená, že hodinový signál není přenášen a komunikující zařízení se na něm musí shodnout – udává jej tzv. baud rate. Přenos začíná start bitem, za kterým následuje osm bitů dat ukončených stop bitem, před kterým může být přidán ještě paritní bit. Přenos dat spočívá v propojení Tx vodiče prvního zařízení s Rx vodičem druhého a naopak, čímž je uzavřena komunikační smyčka. **UART** je implementován mimo jiné známými rozhraními RS-232 nebo RS-485, která jsou dodnes hojně využívána v průmyslových zařízeních.

Kromě rozhraní výše zmíněných existují další v průmyslu rovněž široce využívaná rozhraní. Pro svou spolehlivost a bezpečnost komunikace bývá v kritických aplikacích (automobilní průmysl) využívána sběrnice **CAN**, univerzálnost a snadnou rozšiřitelnost zase nabízí rozhraní **USB**.

4.2.3 Volba konkrétních senzorů

Senzory pro měření tlaku v hladinách, na kterých operuje chladicí zařízení na bázi kompresorového tepelného čerpadla (ve vysokotlaké části až 30 barů), nepatří mezi levná zařízení, neboť je kladen důraz na jejich konstrukci a spolehlivost. Tlakový senzor musí být přímo připojen do chladicího oběhu a jeho selhání může vést až k úniku chladiva, což se může velmi prodražit. Tlakové senzory jsou zpravidla analogové s napětovým či proudovým výstupem. Pro měření tlaku na nízkotlaké části chladicího oběhu bude použit senzor DIXELL PPR15, na vysokotlaké části bude použit senzor DIXELL PPR30. Jedná se o analogové senzory s

napěťovým výstupem v rozsahu 0,5 až 4,5 V. Sensory vyžadují napájení v rozmezí od 4,5 do 5,5 V, zvládají pracovat v teplotách mezi -40 °C a 135 °C, kdy je výrobcem deklarovaná přesnost 1,2% z celého rozsahu. Rozsah PPR15 je 0 až 15 barů, rozsah PPR30 je 0 až 35 barů. Oba tyto senzory jsou stejné konstrukce s vnitřním závitem.

Pro měření teploty byl vybrán digitální senzor DS18B20 komunikující po sběrnici One-Wire. Volbu tohoto senzoru podpořila jeho nízká cena a široký rozsah hodnot – od -55 °C do 125 °C. Vestavěný A/D převodník navíc umožňuje volbu počtu bitů pro konverzi, a tím pádem volbu přesnosti senzoru. Výhodou je také jednoduchost připojení i více senzorů na jeden digitální vstup/výstup z mikrokontroléru.

Pro umožnění budoucí změny volby jiného digitálního senzoru je plánováno z mikrokontroléru vyvést konektory pro snadné připojení jiných senzorů komunikujících po jiném typu sběrnice.

4.3 Komunikační rozhraní

Jakmile měřicí zařízení získá data ze senzorů, je nutné je přenést na server, na kterém budou zobrazována a ukládána. Pro jednodušší implementaci je přenos dat často řízen již definovanými protokoly. Využívají se převážně dva, a to **HTTP** a **MQTT**. Informace o nich byly čerpány ze zdroje [27], ve kterém lze najít i jejich porovnání z hlediska datové propustnosti a dalších významných kritérií.

HyperText Transfer Protocol (HTTP) je hojně používaný protokol pro přenos dat (velmi často serializovaných ve formátu **XML** nebo **JSON**) na internetu. Často se využívá tzv. **REST API (Representational State Transfer)** – s daty se pracuje voláním metod, což je jednoduché na implementaci, avšak pro **Internet of Things (IoT)** zařízení nevhodné kvůli velké režii, zvláště pokud je využíván transportní protokol **TCP** (který udržuje spojení a potvrzuje přijaté pakety). Režie **HTTP** při velkém množství malých datových bloků může způsobovat vážné problémy, jako jsou např. nepřiměřené zabírání síťových zdrojů nebo velká zpoždění.

Na rozdíl od synchronní **HTTP** komunikace je **Message Queueing Telemetry Transport (MQTT)** protokol asynchronní a využívá prostředníka, který je označován jako „broker“. Ten shromažďuje bloky dat, které jsou po síti posílány, a stará se o to, aby doputovaly tam, kam mají. Bloky dat jsou děleny do témat (*topics*) a jednotlivá zařízení se pak hlásí k odběru dat určitého tématu pomocí metody „Subscribe“ a nebo data určitého tématu odesílají metodou „Publish“. Témata mají hierarchickou stromovou strukturu, takže lze jedním dotazem získat data všech „podtémat“ tématu, o který žádáme. **MQTT** nabízí tři úrovně spolehlivosti přenosu dat – **Quality of Service (QoS)**. Při **QoS 0** je zpráva publikována pouze jednou a není zaručeno doručení příjemci ani brokeru. Při **QoS 1** je brokerem odesílateli potvrzeno přijetí dat, avšak poté broker pošle data všem zařízením přihlášeným pro jejich odběr již bez potvrzování. U **QoS 2** potvrzuje přijetí dat koncové zařízení a je zde tedy jistota doručení, avšak za cenu opět rostoucí režie.

MQTT broker je v podstatě síťová služba. Pro její využívání měřicím zařízením je potřeba se připojit k internetu, buď kabelovým spojením anebo pomocí bezdrátových standardů. Informace o následujících čtyřech standardech (Wi-Fi, ZigBee, SIGFOX a LoRa) byly čerpány ze zdrojů: [12, 20, 23], u dalších je zdroj uveden v textu.

Wi-Fi

Dnes nejrozšířenější bezdrátovou komunikací na krátké vzdálenosti je Wi-Fi – soubor IEEE 802.11 standardů, operujících nejčastěji na 2.4 a 5 GHz frekvenci (nejnovější šestá generace umožňuje komunikovat i na 6 GHz frekvenci a jsou snahy o další rozšiřování frekvenčního spektra). Datová propustnost se v nejnovější generaci pohybuje až v jednotkách gigabitů za sekundu [18]. Wi-Fi standard byl původně vytvořen pro datový přenos webového obsahu a videa, avšak pro svou popularitu se dnes velmi často využívá i v **IoT** aplikacích. V ideálních podmínkách je dosah Wi-Fi signálu až sto metrů, ale překážky (např. stěny) tento dosah výrazně snižují. Výhodou tohoto standardu je jeho rozšířenost, univerzalita a datová propustnost. Nevýhodou je relativně malý dosah signálu, přístupový bod, se kterým zařízení komunikuje a na kterém je závislé, musí být umístěn v blízkosti tohoto zařízení.

ZigBee

Dalším standardem spravovaným IEEE je ZigBee (IEEE 802.15.4) – standard pro **Wireless Personal Area Network (WPAN)** se zaměřením na monitoring, kontrolní a senzorické aplikace. Hlavní frekvencí, na které ZigBee operuje, je 2.4 GHz. Tento standard umožňuje zařízením s nízkou spotřebou energie komunikovat po síti, s tím že každé zařízení v této síti je schopné data přijmout a znovu odvysílat pro dosažení jejich požadované destinace, což umožňuje vytvořit velmi efektivní síť. ZigBee má dosah srovnatelný s Wi-Fi a šířka pásma se pohybuje od 20 do 250 kb/s.

SIGFOX

SIGFOX je technologie vyvinuta stejnojmennou francouzskou firmou využívající komunikaci na tzv. ISM rozhlasovém pásmu 868 MHz (resp. 902 MHz v USA). Zároveň se jedná o tzv. ultra narrowband technologii modulující data metodou **BPSK** – posunutím fáze nosné frekvence malých úseků spektra (**Binary-Phased Shift Keying**). Toto dodává příjemci signálu možnost naslouchat na malém úseku spektra, což zmírňuje efekt rušení. Tok dat je řízen základnovými stanicemi, prostřednictvím kterých spolu komunikují koncová zařízení vybavená levným rádiovým modulem. Myšlenkou tohoto standardu je vytvořit široce dostupnou síť pro nízkoobjemovou datovou komunikaci levných IoT zařízení, jako jsou např. chytré elektroměry, které automaticky odečítají a odesílají svůj stav. SIGFOX umožňuje obousměrnou komunikaci, avšak kapacita downlinku (tedy směru od základnové stanice ke koncovému zařízení) je omezena. Výhodou je velmi nízká spotřeba energie – jedná se o **LPWAN** síť (tedy nízkoenergetická širokopásmová síť) a velké pokrytí signálem (až 94% pokrytí území ČR⁵). Nevýhodou jsou relativně velké latence, které lze navíc těžko predikovat.

LoRa

LoRa využívá stejných rádiových frekvencí jak pro koncová zařízení, tak pro základnové stanice (v LoRa terminologii označované jako Gateway – vstupní brána) jako SIGFOX. Na rozdíl od něj se jedná o otevřený standard (po vstoupení do LoRa Alliance⁶). Kdokoliv si tak může vytvořit a plně spravovat vlastní síť (případně i vytvořit hardwarová zařízení).

⁵Podle oficiálních webových stránek společnosti SIGFOX: <https://sigfox.cz/cs>

⁶LoRa Alliance udržuje informace o společnostech vyvíjejících LoRa zařízení, o vstupních branách, jejich lokalizaci a podobně. <https://loro-alliance.org/>

Cena LoRa terminálů je vyšší než u SIGFOX, zdá se, že tyto dvě technologie sdílejí stejný cíl, a to stát se běžným standardem pro nízkoenergetickou širokopásmovou IoT komunikaci (LPWAN).

NB-IoT

Další technologií pro nízkoenergetickou širokopásmovou komunikaci je NB-IoT, kde NB označuje již zmiňovaný termín narrowband. Informace o tomto standardu byly čerpány ze zdroje [9].

Podobně jako dva předchozí standardy se NB-IoT soustředí na nízkou spotřebu a jednoduchost koncových zařízení. To umožňuje jejich levnou implementaci, avšak koncová zařízení zatím nejsou tolik rozšířena, neboť je tato technologie relativně nová (standardizovaná technickou specifikací 3GPP Release 13 vydanou 3GPP začátkem roku 2016). NB-IoT využívá nízkých frekvencí i frekvencí stejných jako 2G/3G/4G a 5G mobilní sítě, tyto standardy se však navzájem neovlivňují a je možná jejich koexistence.

GSM a GPRS

Standardy 2G mobilní sítě – **GSM** a **GPRS** – jsou v provozu už třetí desetiletí, jsou dobře známy a široce rozšířeny. Informace o nich jsou převzaty z technické specifikace: [1].

Global System for Mobile Communication (GSM) je spojení s přepínáním okruhů. Mobilní operátoři spravující **GSM** sítě účtují zákazníkům podle délky spojení, nikoliv objemu dat, který však stejně nebývá velký kvůli nízkým rychlostem v řádu kilobitů za sekundu.

Naproti tomu **General Packet Radio Service (GPRS)**, jak už z názvu vyplývá, je paketová síť s klasickou **IP** adresací a je proto vhodná pro datové spojení s internetem. Stejně jako SIGFOX, LoRa a **NB-IoT** je **GPRS** síť hvězdicové topologie – koncová zařízení komunikují rádiem prostřednictvím základnové stanice buď mezi sebou navzájem nebo dále do internetu. Výhodou **GPRS** je, že koncové zařízení může být stále „online“ a dostupné i pro zpětnou (downlink) komunikaci. Postupem času byly tyto standardy nahrazeny vyššími generacemi mobilních sítí, avšak jednoduchá **IoT** zařízení stále mobilní sítě druhé generace využívají.

V současné době probíhá vývoj 5G sítě, jejíž rádiová spojení operují na několika frekvenčních hladinách. Tato síť je navrhována tak, aby zvládla nápor komunikace mezi obrovským množstvím zařízení a aby umožňovala vysoce spolehlivou komunikaci s nízkými latencemi. V budoucnu se tedy dá očekávat postupné vypínání druhé generace mobilních sítí a přechod na generace vyšší. Tyto informace byly zjištěny z technické specifikace [2], kde lze najít i další podrobnosti.

Volba komunikačního rozhraní

Záměrem této práce je vytvořit zařízení co možná nejvíce nezávislé na svém umístění a okolí. Proto bylo potřeba zvolit komunikační standard, který nevyžaduje správu dalšího prvku jako je Wi-Fi nebo ZigBee přístupový bod či LoRa gateway. Jako vhodná volba se jeví SIGFOX, avšak v České republice je účtován podle počtu odeslaných zpráv, kterých umožňuje denně odeslat jen omezené množství. **NB-IoT** je vhodné pro nízkoenergetické aplikace, je to ale stále relativně nový standard a koncová zařízení zatím nejsou natolik levná jako koncová zařízení pro mobilní sítě druhé generace. S ohledem na skutečnost, že měřicí zařízení bude mít stálý přísun energie, byl nakonec zvolen standard **GPRS**.

4.4 Shrnutí návrhu

Měřicí zařízení bude vystavěno na základě vývojového kitu TTGO T-Call ESP32 SIM800L osazeného mikrokontrolérem ESP32-WROVER-B a modulem SIM800L. Naměřená data budou publikována na **MQTT** broker pomocí **GPRS** datového spojení.

K tomuto vývojovému kitu budou napojeny dva analogové senzory DIXELL PPR15 a PPR30 s napětovým výstupem pro měření tlaku, a to jeden na nízkotlaké a druhý na vysokotlaké části chladicího oběhu. Dále bude zařízení disponovat konektory sběrnic **I²C**, **SPI** a OneWire, což má za cíl modularitu a možnost změny typu teplotních senzorů. Pomocí zmíněných konektorů budou připojeny senzory pro měření teplot v místech chladicího oběhu určených na obrázku 3.2. Měření teplot bude realizováno senzory DS18B20, komunikujícími s mikrokontrolérem sběrnicí One-Wire.

Mikrokontrolér bude také snímat kontaktní výstupy řídicích prvků chladicího zařízení udávající stav těchto prvků. Informace o těchto stavech bude rovněž odesílat obslužné webové aplikaci pomocí protokolu **MQTT**.

Kapitola 5

Technická realizace měřicího zařízení

Základem měřicího zařízení je vývojový kit TTGO T-Call, s mikrokontrolérem ESP-32 WROVER-B a **GPRS** modulem SIM800L. Realizace se dá rozdělit do dvou hlavních částí: sestavení hardware zařízení a jeho obslužný program – firmware.

5.1 Hardware

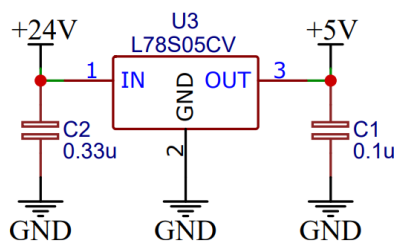
Pro návrh schématu zapojení měřicího zařízení i pro tvorbu nákresu desky plošných spojů byl použit online webový nástroj EasyEDA¹, který má velkou uživatelskou základnu a obsahuje rozsáhlou knihovnu součástek definovaných jak samotnými výrobci, tak uživateli.

Typické umístění měřicího zařízení bude uvnitř skříně rozvaděče chladicího zařízení s ostatními řídicími prvky. Měřicí zařízení je proto umístěno do krabičky, kterou je možné přichytit na standardní nosnou (DIN) lištu rozvaděče. Krabička má šířku čtyř DIN modulů a její horní i spodní stranu přesahuje deska plošných spojů se svorkovnicemi pro připojení napájení, senzorů a digitálních vstupů či výstupů. Deska je dvouvrstvá s rozlitou zemí na spodní straně. Její návrh je vyobrazen na obrázku 5.2. Jedná se o druhou verzi desky plošných spojů. Při první verzi se projevila chyba v návrhu spočívající v připojení analogových senzorů na piny **GPIO** 32 a **GPIO** 33, které však rovněž využívá modul SIM800L. Ověření funkčnosti pinů bylo prováděno bez inicializace SIM800L modulu, po jeho inicializaci však modul interferoval se signálem z analogových čidel tlaku, který tím byl znehodnocen. Tento problém byl vyřešen úpravou schématu zapojení a tvorbou nové desky plošných spojů. Výsledná deska je navíc doplněna o dutinkové lišty po obou stranách vývojového kitu pro zpřístupnění všech pinů pro případné další použití. Navíc jsou na desce umístěny sběrnice **SPI** a **I²C** pro případné další senzory či vstupně-výstupní periferie (display, externí paměť, ...). Tímto je docíleno modularity a snadného případného rozšíření měřicího zařízení.

Napájecí obvod

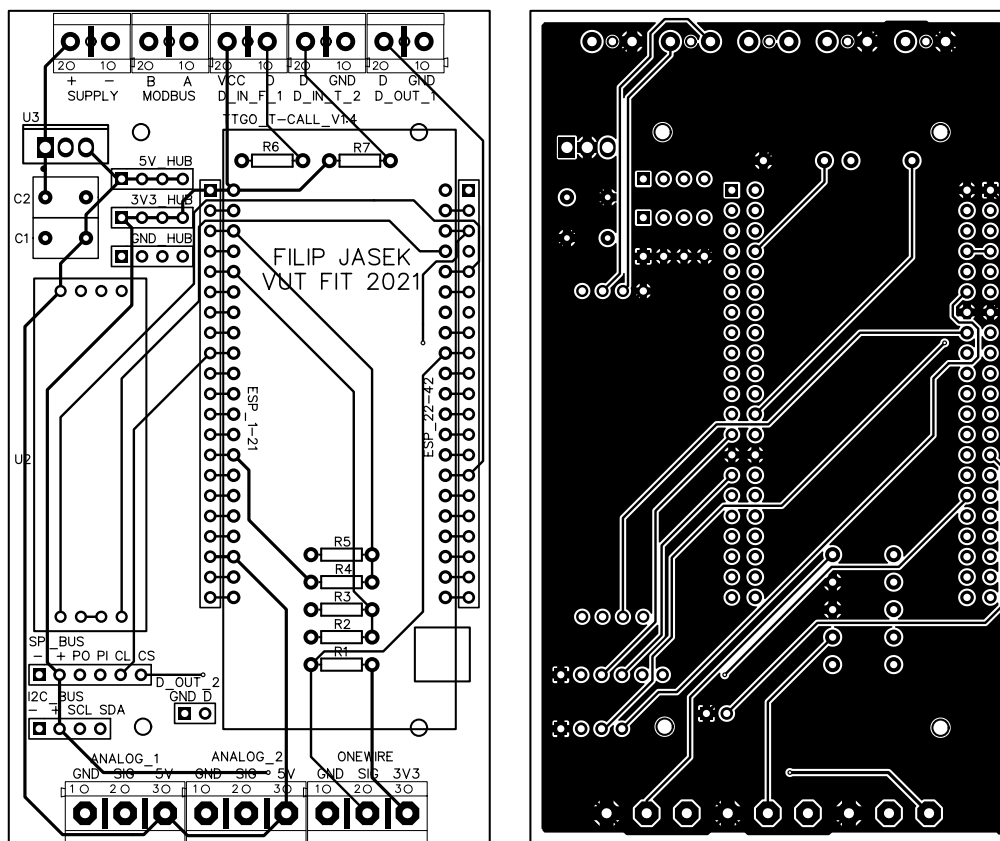
Rozvaděče chladicích zařízení obvykle obsahují transformátor s výstupem 12 V nebo 24 V stejnosměrného napětí pro napájení řídicích prvků. Tento transformátor je vhodné použít i pro napájení měřicího zařízení. Na desce měřicího zařízení je proto umístěn lineární stabilizátor napětí L78S05CV, který je schopen vstupní napětí o velikosti až 35 V stabilizovat na

¹EasyEDA: <https://easyeda.com/>



Obrázek 5.1: Schéma napájecího obvodu.

hodnotu 5 V. Podle doporučení výrobce tohoto stabilizátoru je mezi zem a svorku vstupního napětí přidán kondenzátor o kapacitě 330 nF a mezi zem a svorku výstupního napětí kondenzátor o kapacitě 100 nF , jak je patrné ze schématu napájecího obvodu na obrázku 5.1. Ztrátový výkon stabilizátoru napětí se přeměňuje na teplo. Toto teplo je odváděno měděným plechem, který plní funkci pasivního chladiče, a zároveň je k němu uchycena anténa GPRS modulu.



Obrázek 5.2: Deska plošných spojů.

Výslednou podobu hardwarové části měřicího zařízení a jeho umístění na DIN lištu rozvaděče zachycuje obrázek 5.3.



Obrázek 5.3: Výsledná podoba měřicího zařízení.

5.2 Firmware

Firmware zařízení je zapsán v odnoži jazyka C++ pro programování mikrokontrolérů označované *Wiring*. Využita byla platforma pro vývoj vestavěných systémů Platformio².

Program po startu procesoru nejprve provede funkci `setup()`, ve které je definované nastavení módu pinů, inicializovaná komunikace s modulem SIM800L a výstupním sériovým monitorem. Rovněž je zde inicializovaná knihovna pro práci s nevolatilní pamětí. Následně je v nekonečném cyklu prováděna pracovní smyčka, kterou definuje funkce `loop()`. V této smyčce je zajištěn opětovný pokus o připojení do GPRS sítě i na MQTT broker po daném časovém intervalu – pokud je komunikace přerušena. Kromě intervalů opětovného připojení jsou v nekonečném cyklu stopovány ještě další dva intervaly. Jeden pro vyčtení a následné publikování hodnot všech senzorů a binárních vstupů a druhý (dále označovaný jako „delta interval“) rovněž pro vyčtení všech hodnot, avšak publikovány jsou pouze ty hodnoty, které se změny alespoň o stanovený minimální krok. Tím je docíleno zaznamenání více hodnot v případě rychlé změny průběhu sledované veličiny. Představíme-li si sledovanou veličinu jako graf závislosti její hodnoty na čase, pak hodnoty v místech tohoto grafu s derivací blížící se nule budou zaznamenávány méně často než hodnoty v místech s vyšší absolutní hodnotou derivace křivky takového grafu. Délky intervalů udávané v milisekundách je možné upravovat za běhu zařízení pomocí MQTT zpráv. Stejně tak lze za běhu upravit i minimální změny teplot a tlaků pro publikování.

Protože měření nejsou pouze v pravidelných intervalech, je nutné řešit otázku záznamu času u každé publikované hodnoty. Nabízí se zaznamenávat čas až na straně serveru, který bude MQTT zprávy přijímat a zapisovat do databáze. Měřicí zařízení však zaznamenává hodnoty, i když je offline a po obnovení komunikace odešle všechny záznamy najednou. Je proto nutné časovou značku hodnotě přiřadit v době měření. Mikrokontrolér ESP32 neob-

²Platformio: <https://platformio.org/>

sahuje modul poskytující reálný čas (**RTC** modul). Je však možné použít funkci `millis()`, která vrací čas od spuštění mikrokontroléru v milisekundách. Informace o reálném čase je získána ze serveru pomocí dotazu na čas, který uplynul od Unixové epochy (1. ledna 1970) v milisekundách. Po přijetí odpovědi je od tohoto času odečtena aktuální hodnota funkce `millis()` a takto získaná časová značka je uložena do proměnné. Při každém měření je pak k této proměnné přičtena aktuální hodnota funkce `millis()` a takto získaná časová značka je připojena k měřené hodnotě se znakem středníku jako oddělovačem.

5.2.1 Komunikační rozhraní

Komunikace probíhá pomocí **GPRS** mobilní sítě protokolem **MQTT**. **GPRS** spojení zajišťuje knihovna `TinyGsmClient.h`. Po inicializaci `SIM800L` modulu je tento modul zaregistrován do mobilní sítě metodou `gprsConnect()`. Tato metoda jako parametr přijímá tzv. **Access Point Name (APN)**. To určuje, který mobilní operátor má modul do sítě zaregistrovat. Protože komunikace protokolem **MQTT** je nenáročná na datový objem, je využíváno služeb operátora T-Mobile, který nabízí levnou **SIM** kartu s datovým plánem účtovaným po jednotkách megabytů.

Komunikaci s **MQTT** brokerem zajišťuje knihovna `PubSubClient.h`. Pro připojení k brokeru je nejprve nutné specifikovat jeho adresu (nebo doménové jméno) a port, na kterém očekává příchozí komunikaci. Klient se k **MQTT** brokeru připojuje metodou `connect()` ze zmíněné knihovny. Této funkci jsou jako parametry předány údaje pro autentizaci vůči brokeru. Před samotným připojením je ještě specifikována tzv. callback funkce, která se provede při každém přijetí zprávy relevantního tématu. Relevantní témata, která klient odebírá, určuje metoda `subscribe()`, volána po úspěšném připojení právě jednou pro každé odebírané téma.

Logika programu je koncipovaná tak, aby se zařízení bylo schopné po výpadku spojení s **MQTT** brokerem nebo i po výpadku signálu **GPRS** sítě zotavit. Při neúspěšném odeslání zprávy na **MQTT** broker je zkontrolován stav připojení **GPRS** sítě. Pokud metoda `isGprsConnected()` vrátí hodnotu `false`, provede se pokus o obnovení připojení. Je-li neúspěšný, bude tento pokus opakován za daný časový interval (implicitně nastavený na 10 minut). Pokud je mobilní síť úspěšně připojena, je zkontrolováno spojení s **MQTT** brokerem. I zde je v případě potřeby opět proveden pokus o připojení a při jeho neúspěchu je tento pokus opakován opět za daný časový interval. Data jsou ze senzorů snímána i pokud je zařízení offline, avšak místo odesílání jsou ukládána do nevolatilní flash paměti.

5.2.2 Práce s nevolatilní pamětí

Pokud nastane výpadek spojení s **MQTT** brokerem a zařízení se nepodaří komunikaci obnovit, jsou měřená data zapisována do nevolatilní flash paměti o velikosti 4 MB přímo na desce ESP mikrokontroléru. Data v této paměti přetrvávají i po restartu zařízení.

Zápis dat do paměti lze realizovat několika způsoby. Jeden z nich je knihovna `EEPROM.h`, která obsahuje funkce pro čtení a zápis dat po jednotlivých bajtech. Ačkoliv je to funkční řešení, práce s touto knihovnou není pro tento případ užití vhodná, neboť je nutný převod dat na jednotlivé bajty, oddělování jednotlivých záznamů, následné postupné čtení a další funkcionality, které musí zajišťovat sám programátor.

Další způsob práce s daty na flash paměti nabízí knihovna `SPIFFS.h` (Serial Peripheral Interface Flash File System), která s flash pamětí pracuje, jako by to byl klasický souborový systém počítače (avšak samozřejmě zjednodušený). Umožňuje správu souborů v pseudo-stromové struktuře adresářů – soubory jsou fyzicky uloženy na stejné úrovni, avšak jejich

názvy mohou obsahovat prefix „adresářů“ oddělených lomítkem od názvu souboru. Otevření souboru (a i případné vytvoření souboru, pokud neexistuje) zajišťuje metoda `open()` s parametrem určujícím mód „append“, tedy připojení dat na konec souboru. Následně je možné data zapisovat podobně jako jsou odesílána na sériový monitor – tedy metodou `print()` či `println()`. Po dokončení zápisu je vhodné soubor zavřít metodou `close()`.

Jednotlivé záznamy jsou do souboru ukládány vždy na nový řádek, a to jako **MQTT** téma a zpráva oddělená středníkem. Po obnovení komunikace s brokerem je ve funkci `publishFile()` soubor otevřen v režimu pro čtení a v cyklu čtení řádek po řádku. Každý řádek je na pozici středníku rozdělen na **MQTT** téma a zprávu. Tím jsou získány všechny informace potřebné k publikování zprávy do daného tématu. Jakmile jsou publikovány všechny záznamy ze souboru, je soubor z paměti odstraněn.

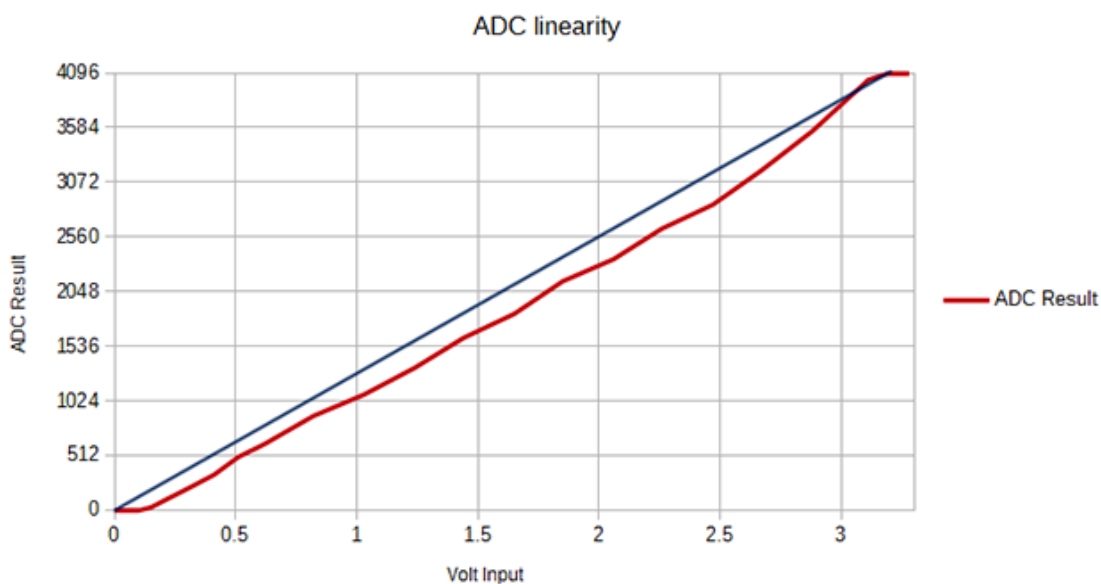
5.2.3 Měření hodnot a A/D převod

Měření tlaku je realizováno analogovými senzory DIXELL PPR15 pro nízkotlakou část a PPR30 pro vysokotlakou část chladicího oběhu. Na výstupu senzorů je možno snímat napětí v lineární závislosti na snímaném tlaku. Výstup senzoru je přes napěťový dělič připojen na analogový vstup mikrokontroléru ESP32 a čtení funkcí `analogRead()`. Čtení probíhá třikrát za sebou s rozmezím 100ms a jako výsledná hodnota je považován aritmetický průměr těchto tří čtení. Mikrokontrolér ESP využívá k převodu analogových hodnot na digitální hodnoty interní dvanáctibitový A/D převodník, který však v krajních mezích trpí na nelinearitu, jak je patrné z grafu 5.1. V některých případech je vhodné tuto nelinearitu kompenzovat vyhledávací tabulkou či přenosovou funkcí [16]. Z tohoto důvodu byla pro oba senzory provedena kontrolní měření závislosti výstupní hodnoty A/D převodníku na skutečný tlak snímaný senzorem zapojeným do systému s plynným dusíkem o tlaku kontrolovaném digitálním manometrem. Výsledky tohoto měření jsou zaneseny do grafu 5.2. Díky tomu, že výstup tlakových senzorů je v rozsahu 0,5 až 4,5 voltů, není využit celý rozsah A/D převodníku a nelinearita je tak patrná až ve vysokých hodnotách tlaků. V chladicím oběhu monitorovaných zařízení však k takto vysokým tlakům nikdy nedochází (dříve je kompresor zařízení vypnut ochranným presostatem). Proto bylo usouzeno, že kompenzovat drobnou nelinearitu vysokých hodnot A/D převodníku není v tomto případě nutné.

Teploty jsou snímány digitálními senzory DS18B20, komunikujícími po sběrnici One-Wire. Senzory umožňují dva typy zapojení – klasický mód a tzv. parazitní mód, kdy je senzor napájen po datovém vodiči, a kromě tohoto stačí přivést k senzoru zem (GND). V tomto případě jsou senzory zapojené klasicky, tedy kabelem se třemi vodiči. Pro minimalizaci rušení potenciální indukci byl zvolen stíněný kabel. O obsluhu teplotních senzorů se stará knihovna `DallasTemperature.h`. Metodou `requestTemperatures()` je všem senzorům odeslán požadavek na provedení měření, následně je od každého senzoru získána naměřená hodnota ve stupních Celsia metodou `getTempC()`. Té je jako parametr předána unikátní 64bitová adresa, kterou má každý senzor od výroby přidělenou.

5.2.4 Význam jednotlivých MQTT témat

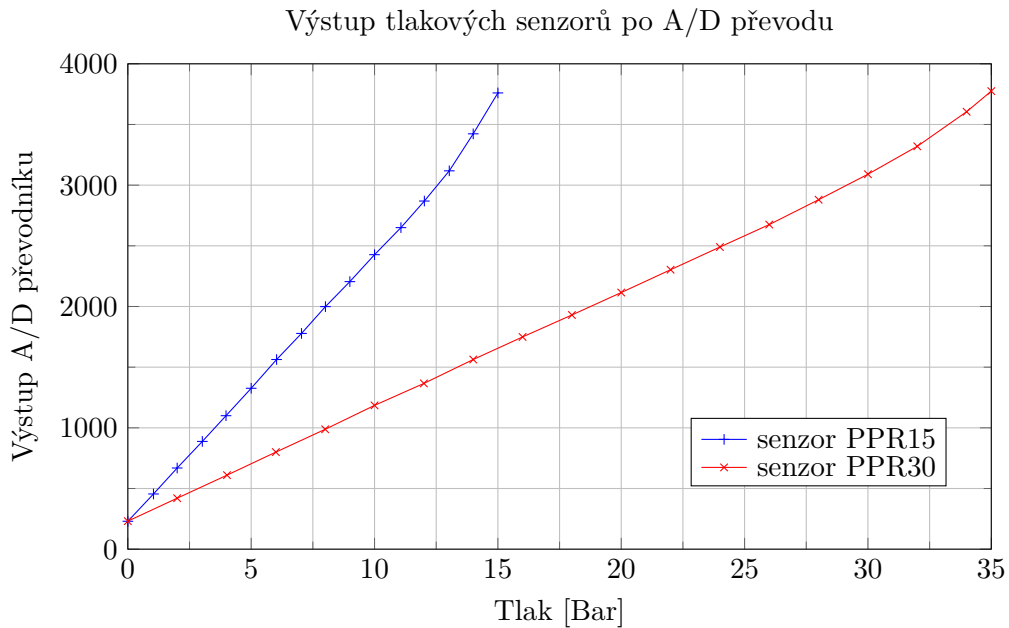
Pro měřicí zařízení představují **MQTT** témata a zprávy jakési **API**, díky kterému je schopno komunikovat s webovou aplikací zobrazující data uživateli. Z pohledu měřicího zařízení lze rozlišovat témata jako odebíraná a odchozí – do kterých jsou data publikována. Kvůli identifikaci zařízení má každé téma prefix daného zařízení, který bude v následujícím seznamu vynechán.



Graf 5.1: Závislost hodnoty z A/D převodníku na vstupním napětí [6].

Témata pro publikování:

- `t_o` – Téma pro odesílání hodnot okolní teploty.
- `t_g` – Téma pro odesílání hodnot teploty chlazené látky (nejčastěji glykolu).
- `t_ki` – Téma pro odesílání hodnot teploty na vstupu kompresoru.
- `t_ko` – Téma pro odesílání hodnot teploty na výstupu kompresoru.
- `t_co` – Téma pro odesílání hodnot teploty na výstupu kondenzátoru.
- `t_fi` – Téma pro odesílání hodnot teploty na vstupu filtru.
- `t_fo` – Téma pro odesílání hodnot teploty na výstupu filtru.
- `t_ei` – Téma pro odesílání hodnot teploty na vstupu výparníku.
- `p_ki` – Téma pro odesílání hodnot tlaku na vstupu kompresoru.
- `p_ko` – Téma pro odesílání hodnot tlaku na výstupu kompresoru.
- `d_f1` – Téma pro odesílání stavu digitálního vstupu `D_IN_F_1`.
- `d_t2` – Téma pro odesílání stavu digitálního vstupu `D_IN_T_2`.
- `livenessProbeReq` – Odpověď v případě obdržení zprávy tématu `livenessProbeReq` pro ověření stavu měřicího zařízení.
- `timeSynceReq` – Vyžádání si informace pro časovou synchronizaci měřicího zařízení s reálným časem.



Graf 5.2: Závislost výstupní hodnoty z A/D převodníku na tlaku snímaném senzory.

Odebíraná témata:

- `requestData` – Jedná se o vyžádání si okamžitých hodnot všech senzorů a binárních vstupů bez ohledu na intervaly měření. Obdrží-li zařízení zprávu `true` v tomto tématu, provede měření a odeslání všech hodnot do příslušných témat.
- `setDeltaInt` – Nastavení délky delta intervalu v milisekundách pro publikování hodnot, které se změní alespoň o stanovený minimální práh.
- `setPubInt` – Nastavení délky intervalu v milisekundách pro publikování všech hodnot.
- `setMinTempStep` – Nastavení minimálního prahu teplot pro publikování při uplynutí delta intervalu.
- `setMinPresStep` – Nastavení minimálního prahu tlaků pro publikování při uplynutí delta intervalu.
- `livenessProbeReq` – Pro přijetí požadavku na odeslání zprávy ověřující stav měřicího zařízení.
- `timeSynceRep` – Pro přijetí odpovědi po odeslané zprávě na téma `timeSynceReq`.

Kapitola 6

Webová aplikace pro obsluhu systému

Měřicí zařízení odesílá na **MQTT** broker sesbíraná data ve formě zpráv obsahujících časové značky (milisekundy uběhlé od Unixové epochy) a hodnoty měřených veličin do příslušných témat. Sama o sobě tato data nemají příliš velkou vypovídající hodnotu. Klíčový aspekt tak představuje jejich prezentace obslužnou aplikací, díky které může uživatel získat potřebné informace o stavu chladicího zařízení. Aby bylo možno k těmto informacím získat snadno přístup jak z počítače, tak z mobilních zařízení, byly pro tvorbu aplikace zvoleny webové technologie. Při návrhu aplikace bylo od počátku počítáno i s mobilními zařízeními a je proto plně responzivní.

6.1 Návrh aplikace

Účelem webové aplikace je primárně zaznamenávat a zobrazovat údaje zjištěné měřicím zařízením. Pro možnost interakce s aplikací z širší škály zařízení je vhodné pro tvorbu aplikace využít webové technologie a při implementaci dbát na responzivnost výsledného řešení.

Aplikace jsou zpravidla děleny na dvě hlavní části: front-end a back-end. Z pohledu webových aplikací je back-endem myšlena ta část aplikace, která je vykonávána pouze na serveru, front-endem pak část vykonávána v prohlížeči uživatele. Pro snazší udržitelnost a případnou znovupoužitelnost je vhodné tyto části od sebe oddělovat již při návrhu architektury aplikace. K tomu jsou využívány architektonické vzory, jako je **Model View Controller (MVC)**. Tento vzor vzájemně odděluje perzistentní data ukládaná v databázi (Model), definici grafického uživatelského rozhraní (View) a logiku aplikace (Controller) [19].

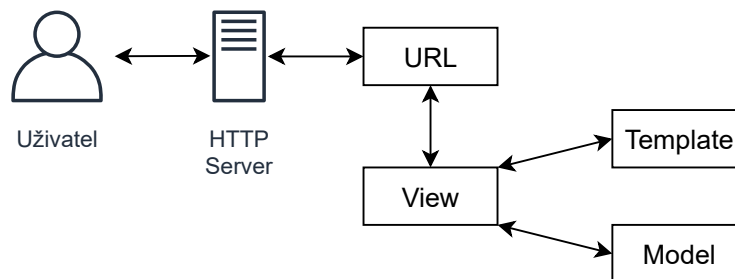
Zvolené technologie

Následující text shrnuje zvolené technologie pro tvorbu webové aplikace. Informace o nich byly čerpány vždy z oficiální dokumentace, která je v jednotlivých případech dostupná formou poznámek pod čarou.

Pro tvorbu obslužné aplikace byl zvolen open source framework *Django*¹. Jedná se o framework založený na programovacím jazyce Python. Výhodou tohoto frameworku je zejména

¹Framework Django: <https://www.djangoproject.com/>

skutečnost, že integruje často používané funkcionality (například správa uživatelů, tvorba a zpracování **HTML** formulářů atp.) a zjednodušuje tak práci vývojářům, kteří se mohou soustředit na tvorbu specifických funkcionalit vlastní aplikace. Pro dosažení modularity a ulehčení znovupoužitelnosti je výsledný projekt složen z modulů, které jsou označovány jako „apps“. Django klade také důraz na bezpečnost a zjednodušuje (v některých případech přímo implementuje) obranu proti častým bezpečnostním nedostatkům (např. **SQL** injection, cross-site scripting nebo clickjacking). Django respektuje **MVC**, avšak častěji se zde pro stejný architektonický vzor užívá označení **Model Template View (MTV)**. Těmto pojmům pak odpovídají i názvy souborů v rámci jednoho „app“ modulu. Definice typů dat, které mají být ukládány do databáze, a metod, jak s nimi pracovat, jsou v souboru `models.py`. Rozhraní pro uživatele je definováno prostřednictvím tzv. templates, což jsou soubory ve značkovacím jazyce **HTML**, ve kterých je definováno uživatelské rozhraní (**GUI**). Django implementuje vlastní template **API** – tzv. **Django Template Language (DTL)**, které umožňuje např. vyhodnocovat podmínky, iterovat přes kolekce objektů nebo upravovat formát zobrazení objektu, a to až přímo v **HTML** template. V souboru `views.py` je pak zapsána veškerá logika pracující s daty definovanými modelem. Tato data pak předává front-endu pomocí kolekce `context` a tzv. „renderuje“ (vykresluje) příslušnou template. Zobrazení správné stránky na základě požadavku uživatele zajišťuje směrovač. Ten na základě **Uniform Resource Locator (URL)**, což je adresa zdrojů (například jednotlivých stránek aplikace nebo obrázků), volá příslušné funkce z vrstvy view, které řeší aplikační logiku požadavku. Ty jsou volány podle předpisů zapsaných v souboru `urls.py`. Směrovač zpracovává vše, co je v rámci dané aplikace zadáno do **URL** řádku prohlížeče. Django **MTV** architekturu ilustruje obrázek 6.1. Pro správu projektu je součástí frameworku Django skript `manage.py`, díky kterému je možné přidávat do projektu vlastní app moduly nebo spustit vestavěný testovací server.



Obrázek 6.1: Diagram Django **MTV** architektury.

Jako systém řízení báze dat byl zvolen *SQLite*². Jeho transakce respektují princip **ACID**, jsou tedy atomické (transakce se provede celá nebo vůbec), konzistentní (transakce převádí databázi mezi dvěma konzistentními stavy), izolované (dílčí operace uvnitř transakce jsou vnějším operacím skryty) a trvalé (změny, které se během transakce provedou, jsou skutečně zapsány do paměti). Díky tomu je systém schopen se zotavit i po pádu během práce s daty. I přes to všechno se může databáze dostat do nedefinovaného stavu, neboť je **SQLite** databáze na disk uložena formou jednoho souboru. Tento soubor může být poškozen například zásahem jiného procesu. **SQLite** byl zvolen pro své snadné nasazení i proto, že má ve frameworku Django velmi dobrou podporu. Komunikace s tímto systémem probíhá za pomoci dotazovacího jazyka **SQL**, avšak tuto komunikaci zajišťuje samotný framework

²SŘBD **SQLite**: <https://www.sqlite.org/>

Django na pozadí. Aplikace tedy s databází interaguje pouze prostřednictvím metod modelu, čímž je dosažena požadovaná abstrakce a je možné snadno přemigrovat aplikaci na jiný systém řízení báze dat.

Tvorbu grafického uživatelského rozhraní ulehčují sady nástrojů umožňující využití šablon založených na značkovacím jazyce **HTML** s vizuální podobou definovanou **CSS** kaskádovými styly. Velmi populární open source sadou nástrojů je *Bootstrap*³. Ten umožňuje použít předdefinované prvky stránky, jako jsou např. tlačítka, tabulky nebo navigační panely. Definuje způsoby rozložení prvků na stránce tak, aby byla responzivní - tedy přizpůsobující se velikosti displaye (případně okna prohlížeče), ve kterém je zobrazována. Pro využití této sady nástrojů mají vývojáři možnost stažení souborů s **CSS** styly, které do svého projektu přidají, a nebo na styly pouze odkázat. Pomocí **Content Delivery Network (CDN)**, což je síť vzájemně propojených počítačů pro dosažení velké dostupnosti dat a spolehlivosti, se pak aplikují styly z této sítě a není nutné je stahovat a ukládat ve vlastním projektu.

Složitější konstrukce uživatelského rozhraní obsluhuje jazyk JavaScript s využitím open source knihovny *jQuery*⁴, která se zaměřuje na interakci mezi JavaScriptem a **HTML**. Integrace jQuery do projektu je provedena opět pomocí **CDN** odkazu. Tato knihovna usnadňuje například obsluhu událostí, animace nebo přístup k elementům stránky pomocí **Document Object Model (DOM)**. **DOM** je standard pro udržování stromové struktury **HTML** elementů a přístup k nim, umožňující jejich modifikaci.

6.2 Implementace aplikace

Webová aplikace je implementována v jazyce Python za použití frameworku Django. Základní nastavení projektu, jako je umístění statických souborů (**CSS**, JavaScript, případně obrázky), nastavení časové zóny, volba databázového systému, určení cesty k dalším důležitým souborům (umístění templates souborů, `urls.py`, `wsgi.py`, ...) a také seznam apps modulů, které jsou součástí projektu, je zapsáno v souboru `settings.py`.

Protože tradiční webové servery nejsou stavěné pro aplikace psané v jazyce Python, využívá se tzv. **Web Server Gateway Interface** – oddělený proces, kterému předává požadavky web server, na němž je aplikace umístěna. **WSGI** požadavek zpracuje a vrátí serveru odpověď v podobě **HTML**. Soubor `wsgi.py` slouží jako výchozí bod Django aplikace pro **WSGI** server. V tomto souboru je rovněž spuštěn **MQTT** klient: `mqtt.client.loop_start()`, který následně komunikuje s měřicím zařízením.

Django projekt obsahuje dva „app“ moduly: *login*, který se stará o správu uživatelů, a *main*, který zabezpečuje veškerou další funkcionalitu – správu chladicích zařízení a záznam měřených veličin. Blokové schéma celé aplikace je ilustrováno na obrázku 6.2.

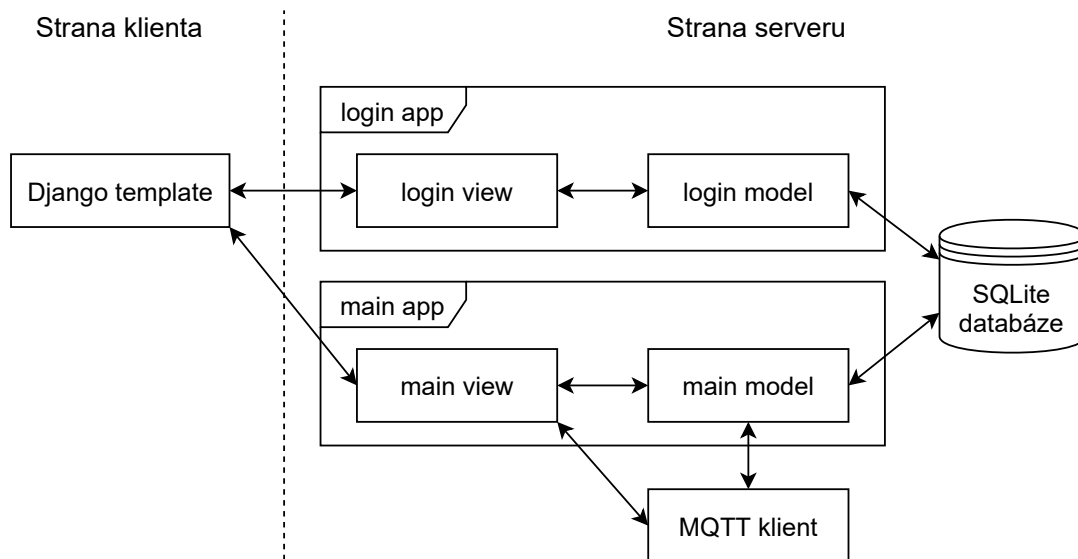
6.2.1 Správa zařízení a záznamu hodnot

Hlavní funkcionalitu aplikace zajišťuje modul *main*. Ten v souboru `main/models.py` definuje tři modelové třídy, které odrážejí tabulky v databázi. Tvorbu těchto tabulek zajišťuje Django po spuštění skriptu `manage.py` s parametry `makemigrations` a `migrate`.

Třída *Device* udržuje informace o chladicích zařízeních, jako je název a popis zařízení, **MQTT** id (to je použito při rozlišování zpráv), skupiny uživatelů mající práva přistupovat k danému zařízení, hodnoty intervalů pro vyčítání (delta interval a interval odeslání

³Sada nástrojů Bootstrap: <https://getbootstrap.com/>

⁴Kniozna jQuery: <https://jquery.com/>

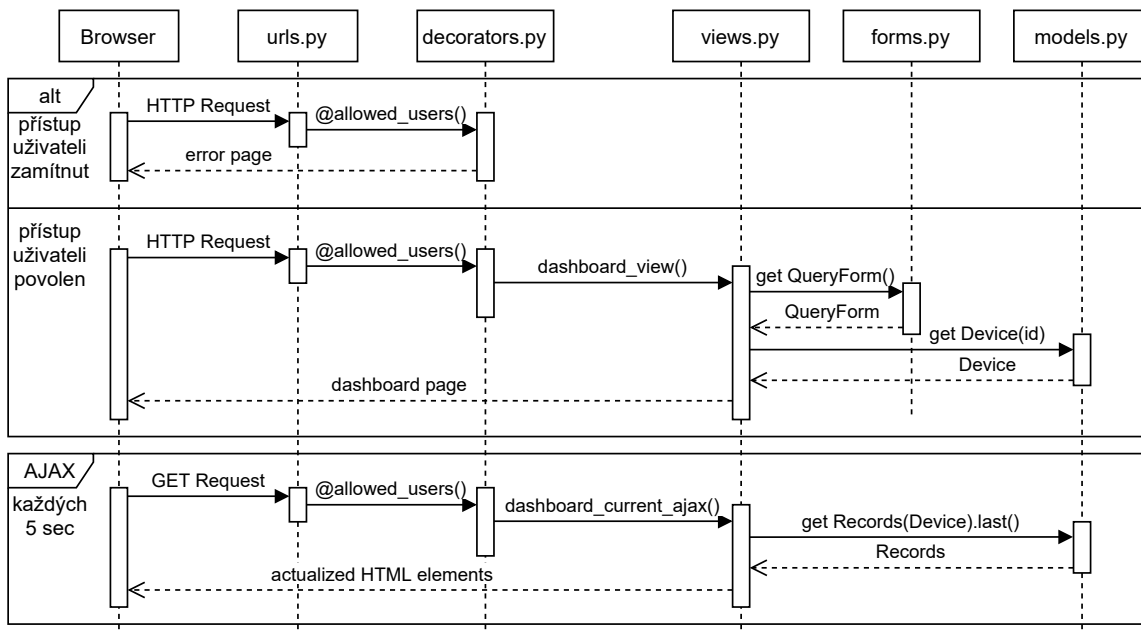


Obrázek 6.2: Blokové schéma aplikace.

všech hodnot) a minimální kroky tlaků a teplot pro odesílání. Tato třída obsahuje také metodu `get_absolute_url()`, která vrací **URL** cestu k vybrané instanci této třídy pro zobrazení detailních informací o instanci – konkrétnímu chladicímu zařízení. Dalším modelem je třída `Record`, která udržuje informace o jednotlivých záznamech měření: odkaz na zařízení, kterého se měřená hodnota týká (formou cizího klíče), měřenou veličinu, její hodnotu, časovou značku a příznak, zda je hodnota mimo povolené limity. Posledním modelem je třída `Thresholds` udržující informace o limitních hodnotách. Pro konkrétní zařízení (opět odkazováno cizím klíčem) a pro měřenou veličinu definuje minimální a maximální povolenou hodnotu.

Modul `main` obsluhuje tři případy užití aplikace: *přehled* (zobrazení aktuálních hodnot ze všech chladicích zařízení, ke kterým má přihlášený uživatel právo přistupovat), *detail* (všechny informace o vybraném zařízení, prohlížení historie měřených hodnot v tabulkách a grafech) a *úprava údajů* chladicího zařízení (změna hodnot uložených v databázi – název, popis, intervaly a minimální kroky). Logika každého z nich je zapsána v minimálně jedné view funkci.

Přehled všech zařízení (který je domovskou stránkou pro přihlášené uživatele) je z pohledu template tvořen elementem `<div class='card'>`, tedy kartou poskytovanou sadou nástrojů Bootstrap, pro každé zařízení. Obsah karty tvoří poslední hodnoty teplot a tlaků příslušného zařízení a také stavy digitálních vstupů (stav oběhového čerpadla a kompresoru), jak je vidět na obrázku 6.4. Aby byly hodnoty aktuální bez nutnosti opětovného načítání celé stránky, je technologií **AJAX** na pozadí periodicky odeslán GET dotaz back-endu pro návrat aktuálních hodnot. View funkce, která tento dotaz zpracovává, vyčte z databáze nejnovější hodnoty a vrátí vykreslenou sadu **HTML** elementů obsahující karty jednotlivých zařízení s aktuálními hodnotami. Za pomoci JavaScriptu je přistoupeno k **DOM** objektu obsahu s původními (již neaktuálními) kartami, které jsou jQuery metodou `replaceWith()` nahrazeny navracenými aktualizovanými kartami z view. **Asynchronous JavaScript and XML (AJAX)** umožňuje dynamicky měnit obsah stránek díky asynchronní komunikaci se serverem na pozadí bez nutnosti zásahu do chování zobrazované stránky.



Obrázek 6.3: Sekvenční diagram vykreslení stránky detailu zařízení.

Nejdůležitějším případem užití je zobrazení detailu zařízení s historií hodnot. Vzájemnou interakci funkcí pro vykreslení stránky s detailem ukazuje sekvenční diagram na obrázku 6.3 (vykreslení před odesláním POST požadavku na historii hodnot). Tato stránka je rozdělena na dvě karty (elementy třídy `card`). První je velmi podobná kartám z obrazovky *prehled*, avšak navíc obsahuje tlačítka pro odeslání hodnoty `true` či `false` (zapnutí a vypnutí) pro každý ze dvou binárních výstupů měřicího zařízení a také tlačítko které, odešle požadavek pro změření a odeslání všech hodnot. Akce prováděné všemi těmito tlačítky jsou odchyceny JavaScriptem, aby nedošlo k jejich výchozí akci (což je odeslání dotazu a znovunačtení stránky). Následně je na pozadí odeslán **AJAX** dotaz typu POST. Ten je odchycen funkcí `request_data_ajax()` definované v souboru `main/views.py`, která dotaz zpracuje a na základě jeho parametru odešle příslušnou **MQTT** zprávu měřicímu zařízení. Hodnoty v této kartě jsou opět aktualizovány pomocí asynchronního dotazu na pozadí (**AJAX**). Druhá karta umožňuje uživateli prohlížení historie naměřených hodnot. Obsahuje formulář pro výběr požadované veličiny a zadání časů, od kterého a do kterého chce uživatel měřené hodnoty zobrazit. Pole pro zadání časů využívají tzv. `datepicker` přidány ve verzi **HTML5** pro uživatelsky příjemný výběr. Tato pole i pole pro výběr veličiny jsou definovány v souboru `main/forms.py`. Po stisku tlačítka „Vybrat“ je tento formulář odeslán. View funkce `dashboard_view()`, která se stará o logiku této stránky, formulář přijme a provede opětovné vykreslení stránky s kolekcí požadovaných dat, která je předána parametrem `context`. Pomocí **DTL** jsou předaná data postupně zpracovávána a následně řádek po řádku vykreslena do tabulky, přičemž ty s hodnotou mimo povolené limity jsou červeně podbarveny. Pro snazší orientaci v datech a jejich uvedení do kontextu jsou hodnoty všech veličin naměřených ve vybraném časovém intervalu zakresleny do grafu pomocí JavaScriptové knihovny *Highcharts*⁵, která je importována opět pomocí **CDN**. Highcharts přijímá data ve formě JavaScriptového pole dvojic čas a hodnota. Data jsou tedy nejprve převedena do této formy a následně předána k vykreslení. Graf obsahuje dvě osy `y` – jednu pro

⁵Knihovna Highcharts: <https://www.highcharts.com/>

Testovací zařízení	
Teploty:	Tlaky:
Teplota okolí (t_o): 10.81 °C	Nízkotlak (p_ki): 4.68 Bar
Teplota glykolu (t_g): 3.81 °C	Vysokotlak (p_ko): 11.80 Bar
Teplota vstupu kompresoru (t_ki): 6.50 °C	
Teplota výstupu kompresoru (t_ko): 35.63 °C	Stavy komponent:
Teplota výstupu kondenzátoru (t_co): 23.38 °C	Kompresor (d_f1): Neběží
Teplota vstupu filtru (t_fi): 23.81 °C	Oběhové čerpadlo (d_t2): Běží
Teplota výstupu filtru (t_fo): 24.44 °C	
Teplota vstupu výparníku (t_ei): 4.37 °C	

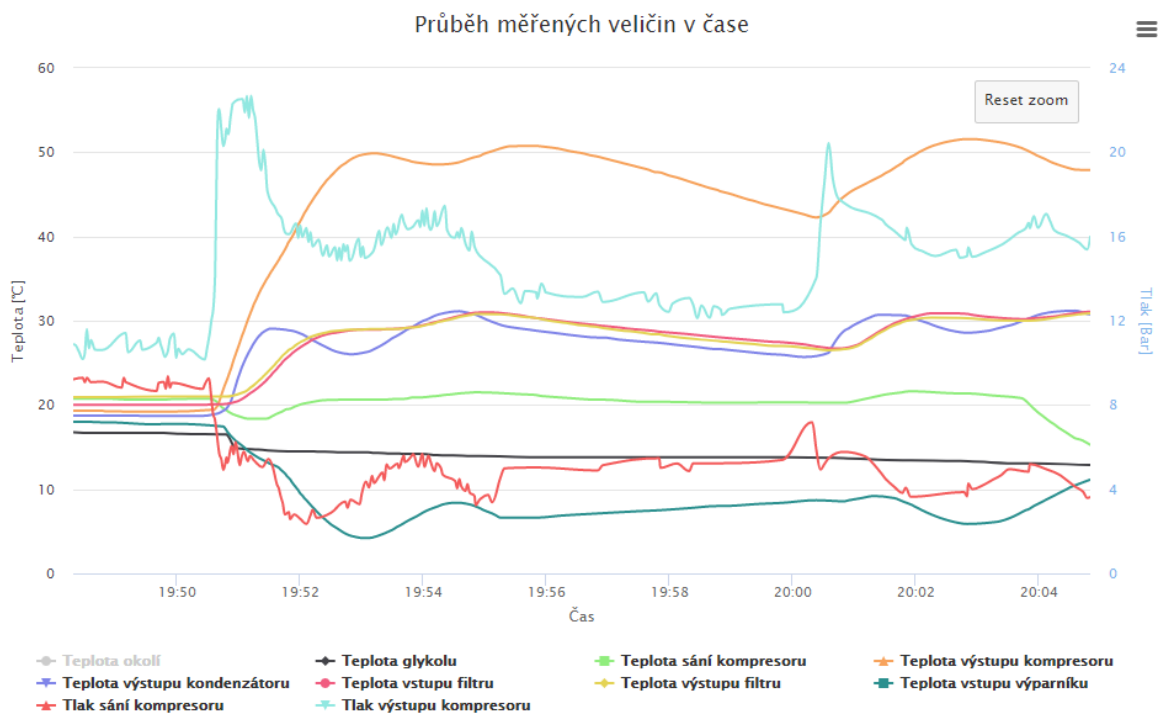
Obrázek 6.4: Karta jednoho chladicího zařízení (výřez snímku obrazovky aplikace).

teplotu a druhou pro tlak. Osa x grafu představuje čas od první po poslední naměřenou hodnotu z vybraného intervalu. Ukázka grafu aplikace je na obrázku 6.5. Highcharts umožňuje uživateli graf přiblížit na vybraný interval osy x, volit veličiny pro zobrazení v grafu, nebo přejetím myši prohlížet přesné naměřené hodnoty na křivce. Tato knihovna rovněž umožňuje export grafu jako obrázek (PNG, JPEG, PDF či SVG) nebo jako CSV či XLS soubor pro další zpracování hodnot.

Úpravu údajů zařízení v databázi zajišťuje view funkce `device_edit_view()`, která vykresluje HTML formulář podle definice v souboru `main/forms.py`. Kromě úpravy údajů v databázi jsou upravovány i parametry měření (hodnoty intervalů vyčítání dat a minimální změny hodnot pro odeslání). Po ověření validity dat a jejich uložení do databáze jsou do příslušných **MQTT** témat publikovány ty parametry měření, které byly změněny. Případné další úpravy zařízení a jeho nastavení jsou povoleny pouze administrátorům a lze je provádět využitím vestavěného administrátorského prostředí.

6.2.2 Správa uživatelů

Modul login využívá vestavěné metody pro přihlašování a odhlašování uživatelů. Jejich použití je zapsáno v souboru `login/views.py`. Kromě toho obsahuje ještě view funkci pro změnu hesla uživatele, která využívá třídu definovanou v `login/forms.py` a rozšiřuje vestavěnou třídu pro formulář změny hesla. Nové heslo je před přijetím zkontrolováno validátory hesel definovanými v souboru `settings.py`. Všechny view funkce pak vykreslují příslušné HTML soubory z adresáře `templates/`. Pro dosažení požadovaného vzhledu formulářů je využíván app modul třetí strany jménem `crispy_forms`. Ostatní úpravy údajů uživatele jsou povoleny pouze administrátorům a jsou prováděny pomocí vestavěného Django administrátorského prostředí. Uživatelé mohou být přiřazeni do skupin, na základě kterých je řešeno oprávnění přístupu k jednotlivým chladicím zařízením.



Obrázek 6.5: Graf historie měřených hodnot (výřez snímku obrazovky aplikace).

6.2.3 Restrikce přístupu a MQTT klient

Protože informace o chladicích zařízeních nejsou dostupné komukoliv, je nutné ověřovat přihlášení uživatele. Django toto řeší elegantně pomocí dekorátorů. Ty stačí přidat před definici funkce, které se daná restrikce týká. Příkladem takového dekorátoru je `@login_required`. V případě aplikace pro sledování stavů chladicích zařízení toto však není dostatečné, protože restrikci přístupu je potřeba řešit na úrovni jednotlivých zařízení (aplikace může spravovat několik zařízení různých uživatelských skupin). Proto byly do souboru `decorators.py` vytvořeny dva vlastní dekorátory. Jeden, který povoluje přístup pouze nepřihlášeným uživatelům, dekoruje view funkci pro přihlášení uživatele – již přihlášený uživatel nemá důvod na tuto stránku přistupovat. Druhý dekorátor zajišťuje přístup pouze těm uživatelům, kteří mají skutečně právo k dekorovaným view funkcím přistupovat. Ve vnitřní funkci dekorátoru je z databáze vybraného zařízení (jehož ID je předáno parametrem view funkce) získán seznam skupin uživatelů, kteří mají povolený přístup k tomuto zařízení. Pokud ID předáno není, je seznam povolených skupin získán z parametru dekorátoru. Následně je zkontrolováno, zda aktuálně přihlášený uživatel patří do některé z povolených skupin. Pokud ano, je vnitřní funkcí dekorátoru navržena příslušná view funkce. Pokud ne, je uživatel přesměrován na jednoduchou template `error.html`, kde je informován o skutečnosti, že nemá oprávnění přístupu k vybranému zařízení. Ukázka tohoto dekorátoru je uvedena ve výpisu 6.1.

Funkci MQTT klienta aplikace zajišťuje balíček *Paho*⁶, práce s ním je zapsána v souboru `mqtt.py`. Po připojení je klient přihlášen pro odběr všech témat týkajících se chladicích zařízení. Následně je definována callback funkce `on_message()`. Protože klient je přihlášen

⁶Python MQTT klient paho: <https://pypi.org/project/paho-mqtt/>

```

1 def allowed_users(allowed_groups = []):
2     def decorator(view_function):
3         def inner(request, id=None, *args, **kwargs):
4             if id is None:
5                 allowed = allowed_groups
6             else:
7                 allowed = Device.objects.get(id=id).groups
8
9             if request.user.groups.exists():
10                if request.user.groups.get().name in allowed:
11                    return view_function(request, id, *args, **kwargs)
12
13                return redirect('error')
14            return inner
15        return decorator

```

Výpis 6.1: Dekorátor pro restrikcí přístupu uživatelů k zařízení na základě skupin.

k odběru všech zpráv, přijímá i ty zprávy, které sám odešle. Aby nedocházelo k nedefinovaným stavům, je přijatá zpráva zpracována v tzv. try bloku, ve kterém se zjistí, jaké zařízení na jaké téma odeslalo jakou zprávu. Klient zpracovává dva typy zpráv: požadavek na synchronizaci času – jako odpověď odešle aktuální časovou značku, a také parametry měření (intervaly a minimální změny hodnot pro odeslání). Druhým typem zpráv jsou pak naměřené hodnoty. Při jejich přijetí je volána funkce `check_faults()`, která (pokud běží kompresor) porovná přijatou hodnotu s jejími limity. Nespadá-li hodnota do povoleného intervalu, je uložena do databáze s příznakem `fault=True`, jinak je uložena s příznakem `fault=False`. Jedná-li se o první překročení takového limitu v řadě, bude vestavěnou Django funkcí `send_mail()` odeslán notifikační e-mail o této skutečnosti všem uživatelům, kteří mají k tomuto zařízení přístup. E-maily však bude možné odesílat až po nasazení aplikace na produkční server se službou **SMTP**.

Kapitola 7

Demonstrace funkčnosti a návrhy rozšíření

Pro otestování celého systému, bylo měřicí zařízení nasazeno na kompresorové tepelné čerpadlo typu voda/vzduch, které teplo z chlazené vody v nádrži o objemu $1m^3$ odvádí do okolního vzduchu (obrázek 7.1). Chladicí oběh tohoto tepelného čerpadla odpovídá schématu na obrázku 3.2. Teploty a tlaky jsou měřeny v místech zakreslených v tomto schématu.



Obrázek 7.1: Proces nasazení měřicího zařízení na tepelné čerpadlo.

Pro umožnění komunikace byl nasazen vlastní **MQTT** broker na **Virtual Private Server (VPS)** za použití Docker kontejneru. To umožňuje jednoduše přemigrovat **MQTT** broker v kontejneru na jiný fyzický server. Rovněž je to výhodné pro snadnou možnost přesměrovat

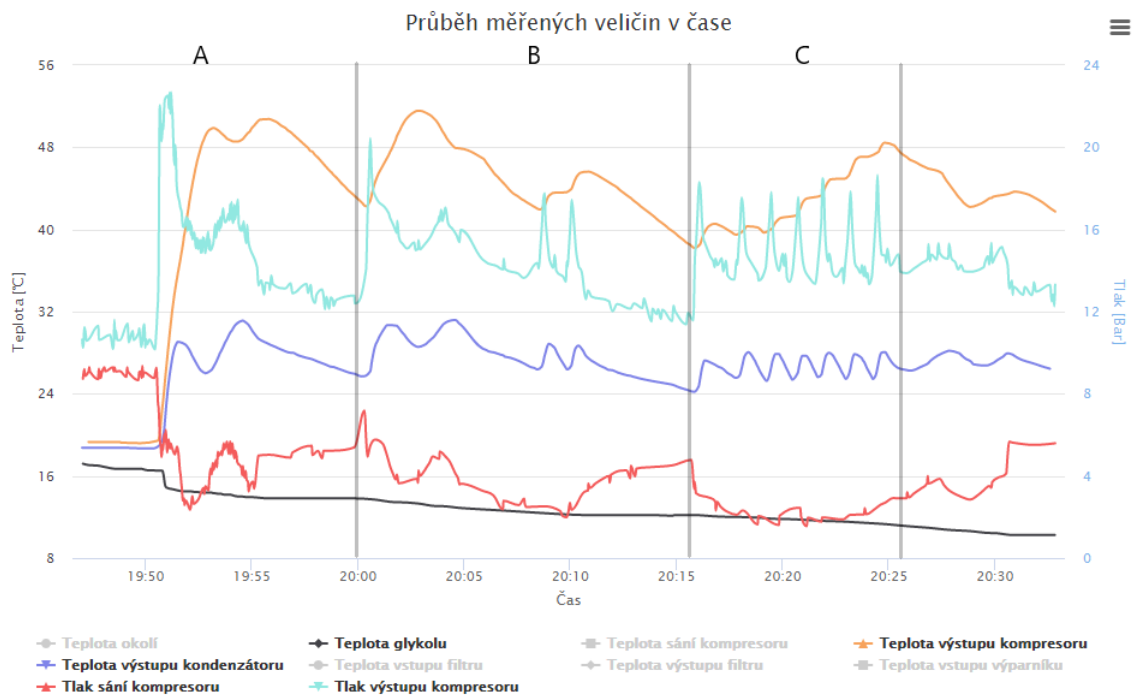
portové číslo, na kterém broker poslouchá. To umožňuje komunikovat na jiném než výchozím portu a vyhnout se tak většímu množství kybernetických útoků. Pro nasazení byl použit Docker image `eclipse-mosquitto`. MQTT broker je tedy dostupný na IP adrese serveru s portem 11883 a přístup je umožněn pouze po autentizaci uživatelským jménem a heslem, které je na serveru uloženo v šifrované podobě.

7.1 Testování za provozu chladicího zařízení

Testování probíhalo čtyři dny, během kterých zařízení provedlo několik cyklů chlazení kapaliny. Byly provedeny také simulace chybových stavů za účelem ověření korektního odhalení problému vyvíjeným systémem.

7.1.1 Korektní chod tepelného čerpadla

Při běžném chodu chladicího zařízení se jeho řídicí prvky snaží udržovat tlak ve vysokotlaké části na určité úrovni. S rostoucí teplotou ve vysokotlaké části oběhu roste i tlak. Snížení tlaku lze docílit zvýšením účinnosti přenosu tepla v kondenzátoru, a tedy snížením teploty vysokotlaké části oběhu. Průběh měřených veličin běžného chodu ilustruje graf na obrázku 7.2.



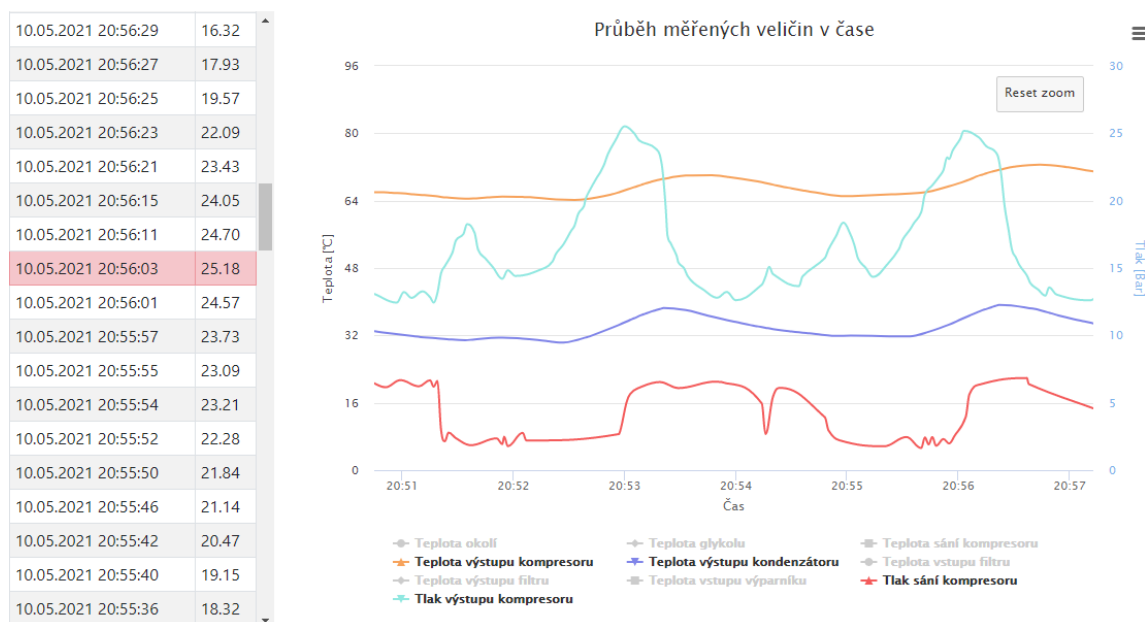
Obrázek 7.2: Záznam měření při běžném chodu tepelného čerpadla (upravený výřez snímku obrazovky aplikace).

V prvním úseku grafu (A) je zaznamenán start chladicího zařízení. Teploty jsou před startem vyrovnány. Pro větší přehlednost grafu byly některé veličiny skryty (například teplota vstupu a výstupu filtru je téměř shodná s teplotou výstupu kondenzátoru, protože filtr není znečištěn). Po startu se prudce změní hodnoty sacího i výtlačného tlaku. Postupně narůstají i hodnoty teplot vstupu a výstupu kondenzátoru. Tlaková čidla jsou zařazena přímo

do chladicího oběhu, změna tlaku je tedy detekována okamžitě. Naproti tomu u teplotních čidel je patrné zpoždění změny hodnoty. To je způsobeno tepelnou setrvačností samotných čidel i potrubí, na které jsou přiložena. Teplotní spád, tedy rozdíl teplot vstupu a výstupu kondenzátoru, který je téměř 20 °C, ukazuje jeho správnou funkci. Pokles tlaku výstupu kompresoru je způsoben zapnutím ventilátoru kondenzátoru, který zvýší jeho účinnost. Ventilátor sledovaného tepelného čerpadla nemá proporcionální řízení otáček. Je-li zapnut, pracuje na plný výkon. Jakmile tlak výstupu kompresoru poklesne, je ventilátor řídicím regulátorem tepelného čerpadla vypnut. To způsobuje cyklování nárůstu a poklesu tlaku, jak je patrné z úseku C popisovaného grafu.

7.1.2 Simulace špatné funkce kondenzátoru

Při špatné funkci kondenzátoru není odváděno přebytečné teplo do okolí s dostatečnou účinností, proto roste tlak ve vysokotlaké části chladicího oběhu až do bodu, kdy ochranný presostat kompresor vypne. Pokles účinnosti může být způsoben znečištěním lamel kondenzátoru, kterými pak neproudí vzduch a teplo jím není odváděno. Rovněž může dojít k poruše ventilátoru, který zajišťuje proudění vzduchu kondenzátorem.

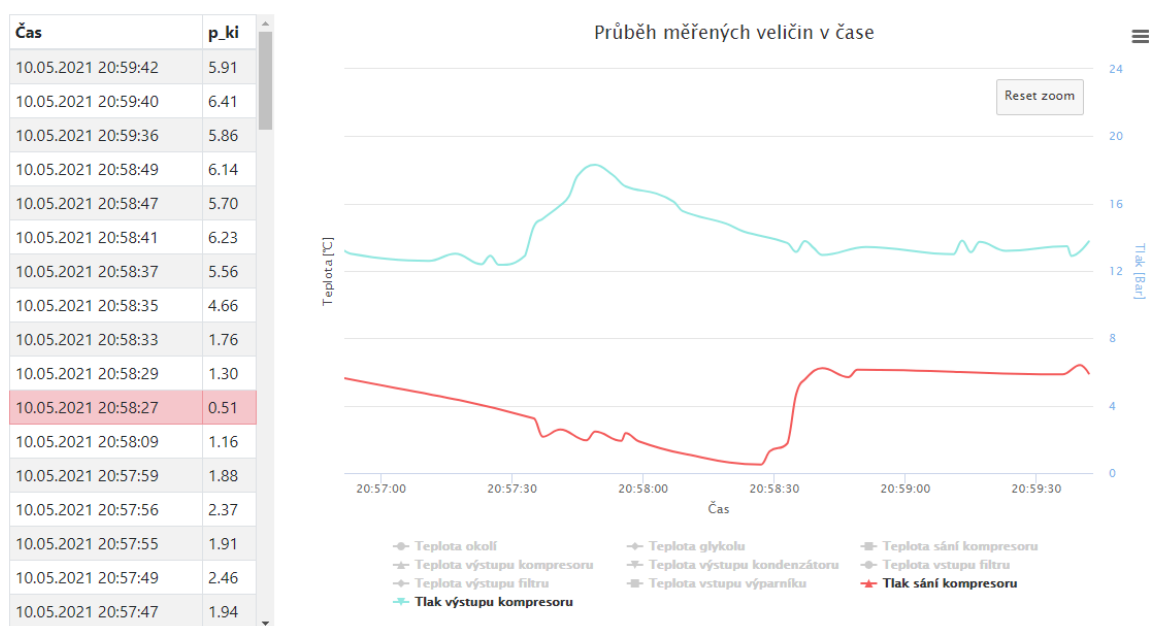


Obrázek 7.3: Záznam měření při simulaci špatné funkce kondenzátoru (výřez snímku obrazovky aplikace).

Simulace špatné funkce kondenzátoru byla provedena vypnutím ventilátoru chvíli po startu chladicího zařízení, a to dvakrát po sobě. Z křivky tlaku výstupu kondenzátoru na obrázku 7.3 je patrný nárůst tlaku, až nad hodnotu 25 barů, která je pro toto zařízení nastavená jako limitní. Po zapůsobení vysokotlaké ochrany je kompresor vypnut a v grafu je patrný strmý pokles tlaku výstupu kompresoru a nárůst tlaku sání (neběží-li kompresor, tlaky se vyrovnávají). V grafu jsou kromě tlaků zobrazeny také teploty vstupu a výstupu kondenzátoru. Na obou je patrný nárůst teploty, což odpovídá teoretickým předpokladům (tlak je fyzikálně svázán s teplotou).

7.1.3 Simulace špatné funkce vstřikovacího ventilu či výparníku

Pokud vstřikovací ventil neotevře přístup kapalnému chladivu do výparníku, kompresor vysaje chladivo z výparníku až na hodnotu tlaku, při které zapůsobí nízkotlaký ochranný presostat. Ten kompresor vypne. Před vstřikovacím ventilem bývá typicky umístěný elektromagnetický ventil, který když neotevře přístup chladiva, rovněž dojde k poklesu tlaku sání až na limitní hodnotu. Stav, kdy tlak sání klesne pod povolenou mez, je označován jako podsání kompresoru. Podsání může být způsobeno i zamrznutím nebo jiným ucpáním výparníku (případně selháním ventilátoru u chladicích zařízení typu vzduch/vzduch). Výparník pak nemá dostatečnou účinnost a chladivo se nevypařuje dost rychle. Z tohoto důvodu klesá sací tlak až na limitní hodnotu.



Obrázek 7.4: Záznam měření při simulaci podsání kompresoru (výřez snímku obrazovky aplikace).

Podsání bylo dosaženo přivřením tzv. rotalocku. To je speciální ventil na vstupu kompresoru, který umožňuje přerušit chladicí oběh a připojení servisního manometru. Postupný pokles tlaku je patrný z obrázku 7.4. Limitní minimální hodnota tlaku sání testovaného chladicího zařízení je nastavena na 1 bar. Krátce po překročení této hranice ochranný presostat kompresor vypne. Aplikace správně zaznamenala problém s tlakem pod povoleným limitem a řádek s jeho hodnotou vykreslila do tabulky červeně.

7.2 Ověření zápisu hodnot do paměti

Pokud dojde k výpadku spojení s MQTT brokerem nebo s GPRS sítí, zapisuje měřicí zařízení data do souboru ve vnitřní flash paměti. Po obnovení komunikace jsou data ze souboru odeslána. Díky tomu, že se časové značky zapisují pro každou hodnotu samostatně, jsou naměřené hodnoty uloženy do databáze se správným časem jejich sejmutí měřicím zařízením. Aplikace tedy získá data v podobě stejné, jako by k žádnému výpadku nedošlo. Tato funkcionality byla ověřena ve dvou scénářích.

Během měření byl zastaven Docker kontejner poskytující **MQTT** službu na **VPS** serveru. Měřicímu zařízení se nepodařilo odeslat měřenou hodnotu a pokusilo se proto opětovně připojit. Když tento pokus selhal, byly hodnoty zapisovány do paměti. **MQTT** služba byla opět spuštěna. Za definovaný časový interval (v tomto případě 10 minut) byl pokus o připojení opakován. Zařízení se podařilo komunikaci obnovit a odeslat naměřená data z paměti.

Podobně bylo ověřeno i selhání **GPRS** spojení. Během měření byla odpojována anténa **GPRS** modulu. Měřicí zařízení bylo během tohoto testování připojeno k počítači, aby mohl být čten výstup sériového monitoru. Ten je ve zkrácené podobě uveden ve výpisu 7.1. Z výpisu je patrné, že při nepovedeném odesílání hodnoty bylo detekováno přerušení spojení s brokerem i s mobilní sítí. Měřicí zařízení se pokusilo spojení obnovit, avšak neúspěšně. Byl nastaven časový interval, za který bude pokus opakován. Mezitím byly měřené hodnoty připojovány na konec souboru v paměti ESP. Anténa byla připojena zpět. Po uplynutí časového intervalu zařízení opětovně provedlo pokus o připojení nejprve k mobilní síti a poté i k **MQTT**. Pokus byl úspěšný, a tak byla veškerá data ze souboru postupně publikována. Měřicí zařízení poté pokračovalo ve své funkci obvyklým způsobem. Měřená data tedy nejsou zapisována do paměti, ale rovnou publikována.

```
16:49:45.997 -> Requesting temperatures...
16:49:46.635 -> Temperature t_o: 12.94
16:49:46.635 -> MQTT connection lost...
16:49:46.635 -> GPRS connection lost...
16:49:46.635 -> Initializing GPRS modem...
16:49:52.248 -> Modem Info: SIM800 R14.18
16:49:52.248 -> Connecting to APN: internet.t-mobile.cz FAILED
16:49:53.710 -> Connection to GPRS Failed, trying again in [ms]: 600000
16:49:53.710 -> Connecting to 176.102.65.218 FAILED with given error code: -2
16:49:57.753 -> Trying again in [ms]: 600000
16:50:01.755 -> Connection lost. Saving data to memory.
16:50:01.970 -> Appending: chj/filip_esp/t_o;1621003779640;12.94
16:50:01.970 -> Temperature t_g: 14.81
16:50:01.970 -> Appending: chj/filip_esp/t_g;1621003794985;14.81
...
16:59:43.513 -> Requesting temperatures...
16:59:48.632 -> Initializing GPRS modem...
16:59:59.036 -> Modem Info: SIM800 R14.18
16:59:59.036 -> Connecting to APN: internet.t-mobile.cz SUCCEEDED
17:00:10.854 -> Connecting to 176.102.65.218 SUCCEEDED
17:00:11.627 -> Publishing file content...
17:00:11.627 -> Topic: chj/filip_esp/t_o Message: 1621003779640;12.94
17:00:11.627 -> Topic: chj/filip_esp/t_g Message: 1621003794985;14.81
...
17:00:14.791 -> Topic: chj/filip_esp/d_t2 Message: 1621004266848;0.00
17:00:14.791 -> Requesting temperatures...
17:00:14.791 -> Temperature t_o: 13.06
```

Výpis 7.1: Zkrácený výstup sériového monitoru ESP při výpadku **GPRS**.

7.3 Návrhy na rozšíření

Během implementace systému pro záznam a hlášení chybových stavů chladicích zařízení vyvstalo několik návrhů rozšíření. Hardware měřicího zařízení byl navržen tak, aby byl připraven na snazší zavedení některých z možných rozšíření.

Přidání periférií mikrokontroléru

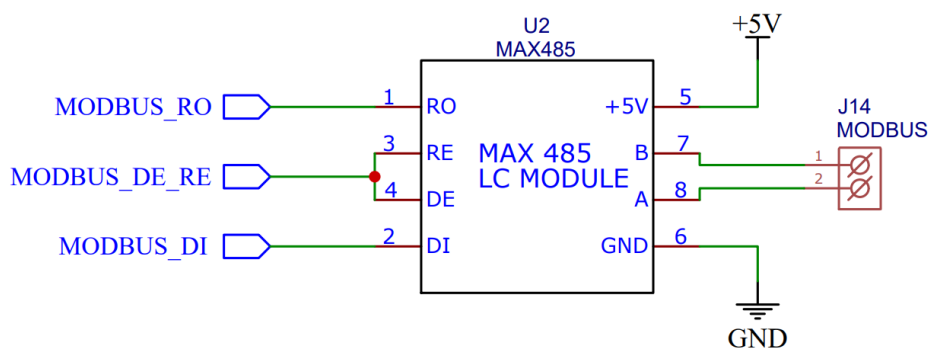
Díky vyvedení sběrnic **SPI** a **I²C** je možné měřicí zařízení rozšířit dalšími perifériemi. Při ztrátě konektivity měřicí zařízení zapisuje snímaná data do vnitřní flash paměti na **SPI** sběrnici pomocí knihovny SPIFFS. Zařízení je možno doplnit modulem pro microSD kartu, který bude s mikrokontrolérem komunikovat právě po sběrnici **SPI**. Firmware měřicího zařízení pak vyžaduje jen drobnou úpravu v podobě změny knihovny zajišťující práci s paměťovým médiem připojeným na **SPI** sběrnici. Větší kapacita paměťového média by umožnila zapisovat data do paměti, i pokud je zařízení online a data jsou korektně odesílána. Pak by šlo snáze přenést naměřená data do počítače pro další zpracování. Avšak díky využití knihovny Highcharts je možné exportovat naměřená data z webové aplikace ve formátu CSV nebo rovnou jako XLS tabulkový dokument. Pro uživatele je tato cesta získání dat pro další zpracování příjemnější než zpracování souboru se surovými daty.

Pro zpřístupnění základních informací o stavu chladicího zařízení a případně i zobrazení hodnot vybraných veličin je možné měřicí zařízení rozšířit o LCD display komunikující po **I²C** sběrnici.

Vzdálená konfigurace řídicích prvků

Chladicí zařízení jsou řízena regulátory, které lze často kromě vestavěné klávesnice konfigurovat také protokolem MODBUS po poloduplexní sériové lince RS-485. MODBUS je široce používaný komunikační protokol v průmyslových aplikacích. Podporuje komunikaci mezi několika zařízeními, tu však iniciuje vždy jen jedno zařízení označované jako *master*. Ostatní zařízení odpovídají na jeho požadavky a příkazy.

Díky obousměrné komunikaci měřicího zařízení s obslužnou aplikací je možné přenášet MODBUS požadavky a příkazy pro řídicí prvky přes **MQTT** zprávy. Měřicí zařízení by pak plnilo funkci jakéhosi prostředníka, který by předával MODBUS příkazy přijaté z obslužné aplikace řídicímu prvku a jeho odpovědi opět předával obslužné aplikaci prostřednictvím **MQTT** zpráv.



Obrázek 7.5: Modul pro komunikaci po sběrnici RS-485.

Aby bylo měřicí zařízení na toto rozšíření připraveno, byl do návrhu hardwarového zapojení přidán modul MAX 485 (obrázek 7.5), který umožňuje komunikaci po sériové lince RS-485. Mikrokontrolér s tímto modulem komunikuje prostřednictvím **UART** sběrnice. Obslužnou aplikaci by pak bylo nutné rozšířit o jakýsi terminál, který by umožňoval odeslat požadavek řídicímu prvku a následně přijmout a zobrazit jeho odpověď.

Náhrada vývojového kitu a úpravy firmware

Při prvotním návrhu bylo počítáno s vývojovým kitem TTGO T-Call. Ačkoliv je použití tohoto kitu pro vyvíjené zařízení vhodné, neposkytuje tolik volnosti v návrhu hardwarové části měřicího zařízení. Některé **GPIO** vývody procesoru nemohou být pro měření hodnot využity, neboť jsou vývojovým kitem obsazeny jinou (byť nepoužívanou) funkcí, která však znemožňuje plnohodnotné použití těchto vývodů. Návrh vlastního zapojení s odděleným procesorem a SIM800L modulem by tento nedostatek eliminoval.

Protože použitý mikrokontrolér ESP32 WROVER-B obsahuje dva mikroprocesory, mohou na mikrokontroléru běžet dva procesy simultánně. Toho by se mohlo využít při pokusech o znovupřipojení v případě ztráty konektivity **GPRS** sítě či komunikace s **MQTT** brokerem. Zatímco se jeden procesor pokouší obnovit komunikaci s obslužnou aplikací, druhý by mohl pokračovat v měření v definovaných časových intervalech se zápisem do paměti mikrokontroléru. V současné implementaci nelze během pokusu o obnovení komunikace získávat data ze senzorů. Není to však příliš velký nedostatek, protože pokus o obnovu spojení trvá přibližně 20 sekund.

Kapitola 8

Závěr

V rámci této diplomové práce byl navržen, implementován a otestován systém pro záznam a hlášení chybových stavů chladicích zařízení.

V první kapitole je uveden přehled chladicích zařízení a principů jejich funkce. Zvláštní důraz klade práce na nejčastější typ – chladicí zařízení na bázi kompresorového tepelného čerpadla. Je zde tedy detailně popsán princip činnosti tohoto zařízení.

Na základě získaných poznatků principu činnosti chladicího okruhu kompresorového tepelného čerpadla byly stanoveny podmínky, za kterých mohou vznikat chybové stavy, a definovány způsoby jejich odhalování. V chladicím oběhu byla vytipována místa pro měření teplot a tlaků, která jsou pro odhalování chyb relevantní.

Další část práce je věnována návrhu hardwarové části monitorovacího systému – měřicímu zařízení. Pro jeho obsluhu byl zvolen mikrokontrolér ESP32 WROVER-B vsazený do vývojového kitu, který rovněž obsahuje modul pro **GPRS** datovou komunikaci.

Podle vytvořeného návrhu bylo měřicí zařízení implementováno. Byla vyrobena deska plošných spojů a na ní sestaven hardware měřicího zařízení. Jeho obslužný firmware zapsaný v odnoži jazyka C++ byl vyvíjen za pomoci nástroje PlatformIO. Mikrokontrolér sleduje hodnoty tlaků vysokotlaké a nízkotlaké části chladicího oběhu analogovými senzory DIXELL PPR30 a PPR15. Teploty ve vytipovaných místech jsou snímány digitálními senzory DS18B20, komunikujícími po One-Wire sběrnici. Získaná data jsou za použití mobilní sítě **GPRS** odesílána na **MQTT** broker, odkud jsou následně odebírána obslužnou aplikací.

Pro záznam měřených hodnot, odhalování a hlášení chybových stavů sledovaných chladicích zařízení byla navržena obslužná webová aplikace. Ta byla implementována v jazyce Python za pomoci frameworku Django. Obslužná aplikace přijímá, vyhodnocuje a ukládá hodnoty měřené v chladicím oběhu vyvinutým měřicím zařízením. Zpracované hodnoty prezentuje uživateli formou tabulek a interaktivních grafů vykreslených za pomoci knihovny Highcharts. Aplikace umožňuje monitorovat několik měřicích zařízení současně. Přístup ke každému z nich je individuálně omezen na základě příslušnosti uživatelů do skupin.

Implementovaný systém byl nasazen do zkušebního provozu na reálné kompresorové chladicí zařízení. Toto zařízení bylo řízeně uváděno do chybových stavů, aby byla ověřena správná funkce vyvíjeného systému. Systém korektně vyhodnotil a zaznamenal nastalé chybové stavy.

Sestavený hardware je navržen s ohledem na možná rozšíření, která jsou uvedena v poslední kapitole této práce. Vyvinutý systém je připraven pro nasazení do ostrého provozu k monitorování chladicích zařízení.

Literatura

- [1] 3GPP. *General Packet Radio Service (GPRS); Service description; Stage 2*. Technical Specification (TS) 23.060. 3rd Generation Partnership Project (3GPP), leden 2015.
- [2] 3GPP. *System architecture for the 5G System (5GS)*. Technical Specification (TS) 23.501. 3rd Generation Partnership Project (3GPP), prosinec 2020.
- [3] AWTREY, D. *Transmitting Data and Power over a One-Wire Bus*. 1997. [Online; navštíveno 10.1.2021]. Dostupné z: http://www.datsi.fi.upm.es/docencia/Micro_C/dallas/1wirebus.pdf.
- [4] BRÁZDIL, M. *Peltierův článek pro chlazení malých výkonů*. Brno, CZ, 2009. Bakalářská práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství. Dostupné z: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=17231.
- [5] CHEN, J.-Y., TAI, K.-C. a CHEN, G.-C. Application of Programmable Logic Controller to Build-up an Intelligent Industry 4.0 Platform. *Procedia CIRP*. 2017, sv. 63, s. 150 – 155. DOI: <https://doi.org/10.1016/j.procir.2017.03.116>. ISSN 2212-8271. Dostupné z: <http://www.sciencedirect.com/science/article/pii/S2212827117302627>.
- [6] CHEUNG, H. *ESP32-adc-calibrate*. [Online; navštíveno 5.5.2021]. Dostupné z: <https://github.com/e-tinkers/esp32-adc-calibrate>.
- [7] DANIEL, M. *Chladicí jednotka domácího pivovaru*. Praha, CZ, 2015. Bakalářská práce. České vysoké učení technické, Fakulta strojní. Dostupné z: https://dspace.cvut.cz/bitstream/handle/10467/63456/F2-BP-2015-Daniel-Marek-BP_daniel.pdf?sequence=1&isAllowed=y.
- [8] DVOŘÁK, Z. *Tepelná čerpadla*. Vyd. 1. Praha: SNTL - Nakladatelství technické literatury, 1987.
- [9] GSMA. *3GPP Low Power Wide Area Technologies*. 2016. [Online; navštíveno 8.1.2021]. Dostupné z: <https://www.gsma.com/iot/wp-content/uploads/2016/10/3GPP-Low-Power-Wide-Area-Technologies-GSMA-White-Paper.pdf>.
- [10] IRAZABAL, J.-M. a BLOZIS, S. *AN10216-01 I²C MANUAL*. 2003. [Online; navštíveno 1.1.2021]. Dostupné z: <https://www.nxp.com/docs/en/application-note/AN10216.pdf>.
- [11] JAŠEK, F. *Návrh tepelného čerpadla voda-voda*. Brno, CZ, 2012. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství. Dostupné z: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=56558.

- [12] KEERTIKUMAR, M., SHUBHAM, M. a BANAKAR, R. M. Evolution of IoT in smart vehicles: An overview. In: *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*. 2015, s. 804–809. DOI: 10.1109/ICGCIoT.2015.7380573. ISBN 978-1-4673-7910-6. Dostupné z: <https://ieeexplore.ieee.org/abstract/document/7380573>.
- [13] KOUBKOVÁ, L. *Vytápění rodinného domu s využitím tepelného čerpadla*. Brno, CZ, 2008. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství. Dostupné z: <https://dspace.vutbr.cz/bitstream/handle/11012/13690/final-thesis.pdf?sequence=8>.
- [14] KOVAŘÍK, V. *IoT systém pro domácnost*. Brno, CZ, 2020. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Dostupné z: <https://www.fit.vut.cz/study/thesis-file/23115/23115.pdf>.
- [15] LEENS, F. An introduction to I²C and SPI protocols. *IEEE Instrumentation Measurement Magazine*. 2009, sv. 12, č. 1, s. 8–13. DOI: 10.1109/MIM.2009.4762946. Dostupné z: <https://ieeexplore.ieee.org/document/4762946>.
- [16] MAIER, A., SHARP, A. a VAGAPOV, Y. Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things. In: *2017 Internet Technologies and Applications (ITA)*. 2017, s. 143–148. DOI: 10.1109/ITECHA.2017.8101926. ISBN 978-1-5090-4816-8. Dostupné z: <https://ieeexplore.ieee.org/abstract/document/8101926>.
- [17] MCLINDEN, M. O. a HUBER, M. L. (R)Evolution of Refrigerants. *Journal of Chemical & Engineering Data*. 2020, sv. 65, č. 9, s. 4176–4193. DOI: 10.1021/acs.jced.0c00338. Dostupné z: <https://doi.org/10.1021/acs.jced.0c00338>.
- [18] MITCHELL, B. *Learn Exactly How „Fast“ a Wi-Fi Network Can Move*. 2020. [Online; navštíveno 7.1.2021]. Dostupné z: <https://www.lifewire.com/how-fast-is-a-wifi-network-816543>.
- [19] PECINOVSKÝ, R. *Návrhové vzory*. Vyd. 1. Brno: Computer Press, 2007. ISBN 978-80-251-1582-4.
- [20] SAMUEL, S. S. I. A review of connectivity challenges in IoT-smart home. In: *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*. 2016, s. 1–4. DOI: 10.1109/ICBDSC.2016.7460395. ISBN 978-1-4673-9584-7. Dostupné z: <https://ieeexplore.ieee.org/abstract/document/7460395>.
- [21] SRDEČNÝ, K. *Tepelná čerpadla*. 1. vyd. Brno: ERA, 2005. 21. století. ISBN 80-7366-031-8.
- [22] STRAKA, M. *Chladicí jednotka domácího pivovaru*. Brno, CZ, 2016. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství. Dostupné z: <https://dspace.vutbr.cz/bitstream/handle/11012/60864/final-thesis.pdf?sequence=6>.
- [23] TARDY, I. C. R., AAKVAAG, N., MYHRE, B. a BAHR, R. Comparison of wireless techniques applied to environmental sensor monitoring. *SINTEF Digital*. 2017, č. 1. Dostupné z: <https://sintef.brage.unit.no/sintef-xmlui/handle/11250/2436270>.

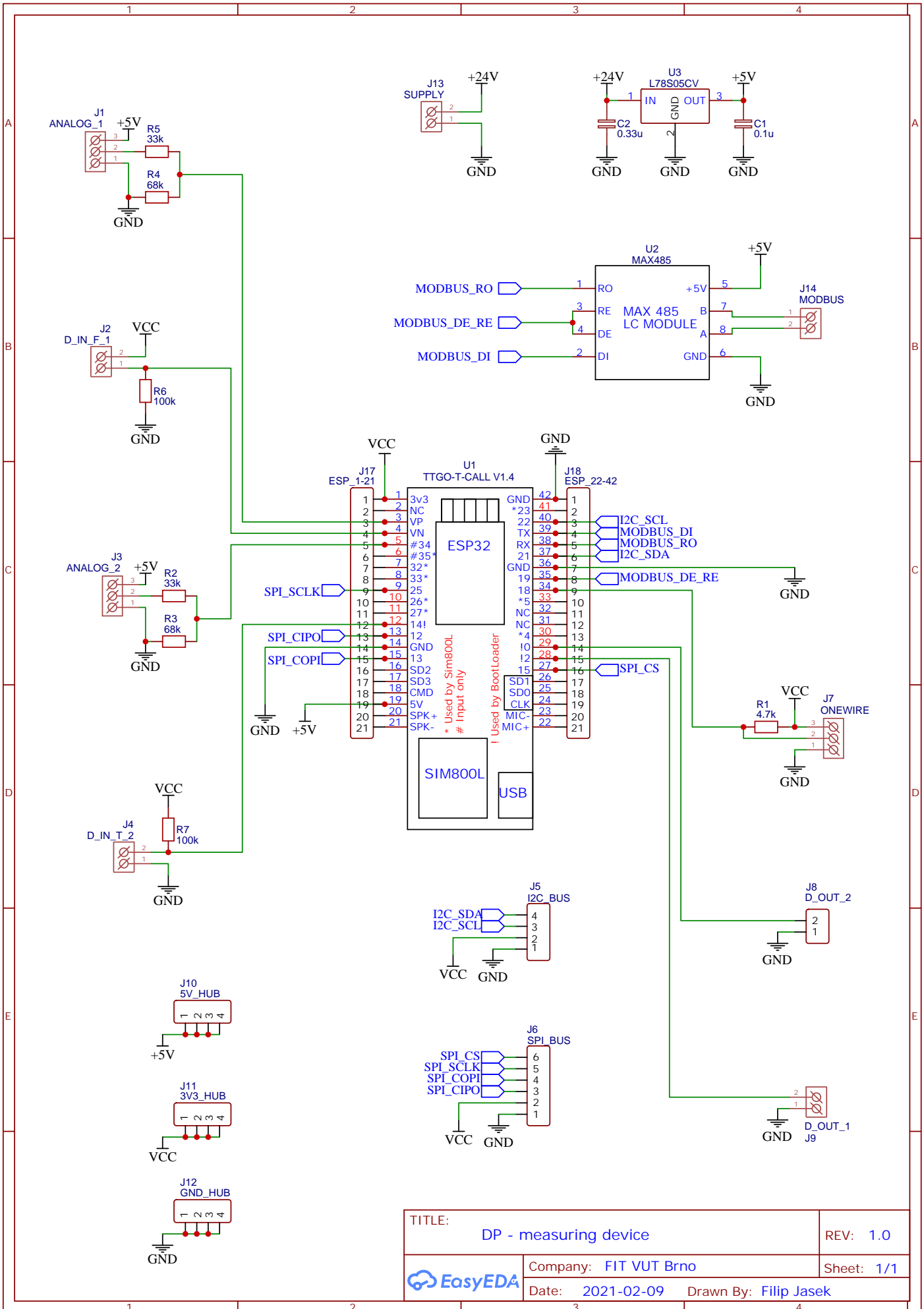
- [24] VRTEK, M. Tepelná čerpadla. *Automa*. Duben 2012, s. 34–37. ISSN 1210-9592. Dostupné z: https://automa.cz/Aton/FileRepository/pdf_articles/9542.pdf.
- [25] WALDEN, R. H. Analog-to-digital converter survey and analysis. *IEEE Journal on Selected Areas in Communications*. 1999, sv. 17, č. 4, s. 539–550. DOI: 10.1109/49.761034. Dostupné z: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=761034>.
- [26] WEBSTER, J. G. a EREN, H. *Measurement, Instrumentation, and Sensors Handbook: Two-Volume Set*. CRC Press, 2014. ISBN 9781439848883.
- [27] YOKOTANI, T. a SASAKI, Y. Transfer protocols of tiny data blocks in IoT and their performance evaluation. In: *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*. 2016, s. 54–57. DOI: 10.1109/WF-IoT.2016.7845442. ISBN 978-1-5090-4130-5. Dostupné z: <https://ieeexplore.ieee.org/abstract/document/7845442>.
- [28] ZALBA, B., MARÍN, J. M., CABEZA, L. F. a MEHLING, H. Free-cooling of buildings with phase change materials. *International Journal of Refrigeration*. 2004, sv. 27, č. 8, s. 839 – 849. DOI: <https://doi.org/10.1016/j.ijrefrig.2004.03.015>. ISSN 0140-7007. Dostupné z: <http://www.sciencedirect.com/science/article/pii/S0140700704000623>.
- [29] ŠÍMA, J. *Noční chlazení budov v podmínkách České Republiky*. Brno, CZ, 2016. Disertační práce. Vysoké učení technické v Brně, Fakulta stavební. Dostupné z: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=156840.

Příloha A

Kusovník a schéma zapojení měřicího zařízení

Kusovník (BOM list)

ID	Název	Označení	Otisk	Popis	Mn.
1	C 0.1u	C1	RAD-0.2	Kond. 100n/100V 10%	1
2	C 0.33u	C2	RAD-0.2	Kond. 330n/50V 20%	1
3	ARK500/3	J1, J3, J7	ARK500/3	Svorkovnice 3pól	3
4	ARK500/2	J2, J4, J9, J13, J14	ARK500/2	Svorkovnice 2pól	5
5	HDR-F-2.54_1X2	J8	HDR-F-2.54_1X2	Dutinková lišta 1x2pin	1
6	HDR-F-2.54_1X4	J5, J10, J11, J12	HDR-F-2.54_1X4	Dutinková lišta 1x4pin	4
7	HDR-F-2.54_1X6	J6	HDR-F-2.54_1X6	Dutinková lišta 1x6pin	1
9	HDR-F-2.54_1x21	J17,J18	HDR-F-2.54_1X21	Dutinková lišta 1x21pin	2
10	R 4.7k	R1	R_AXIAL-0.3	Rezistor 4k7 0,6W 1%	1
11	R 33k	R2,R5	R_AXIAL-0.3	Rezistor 33k 0,4W 1%	2
12	R 68k	R3,R4	R_AXIAL-0.3	Rezistor 68k 0,4W 1%	2
13	R 100k	R6,R7	R_AXIAL-0.3	Rezistor 100k 0,6W 1%	2
14	TTGO-T-CALL	U1	TTGO-T-CALL	TTGO-T-CALL V1.3	1
15	MAX485	U2	MAX485-LC	Modul RS-485	1
16	L78S05CV	U3	TO-220-3	Lin. stabilizátor 5V, 2A	1

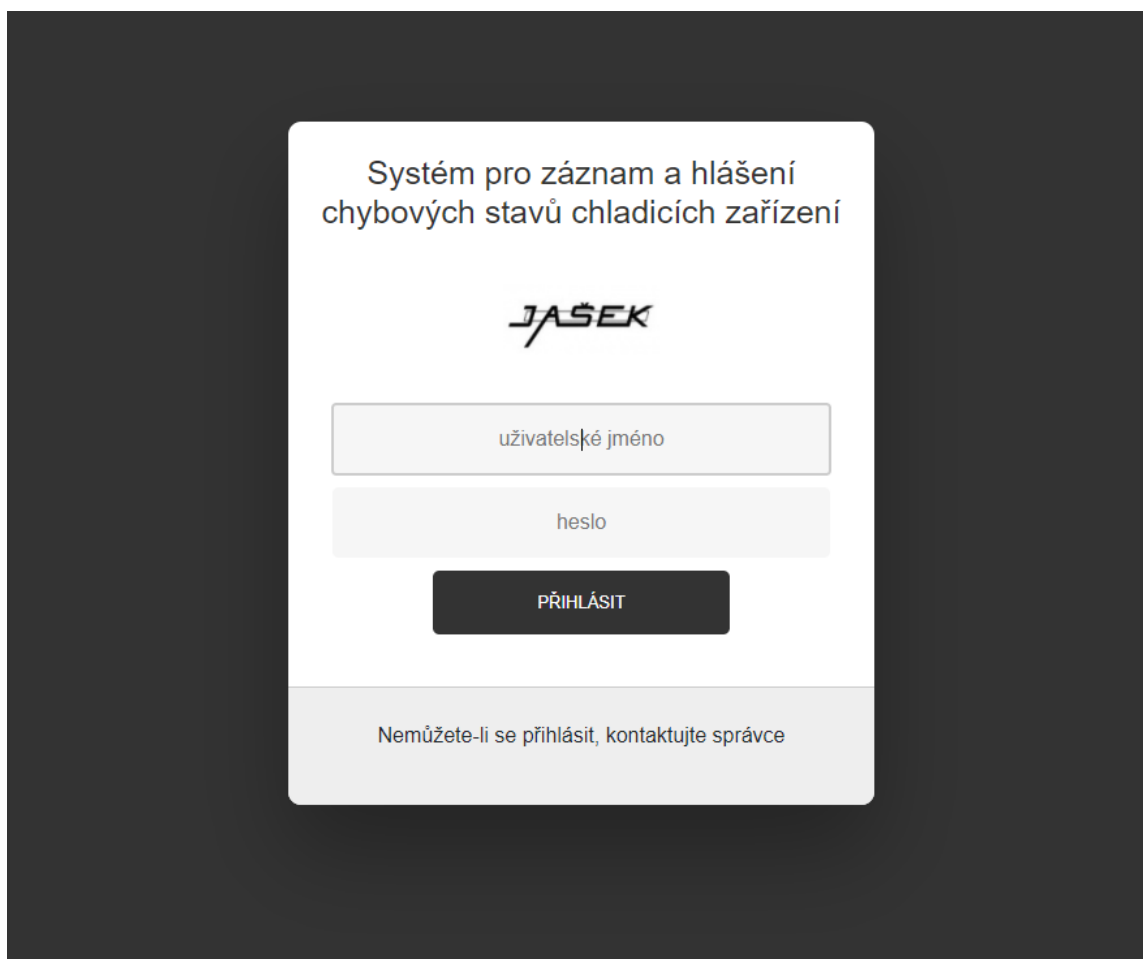


TITLE: DP - measuring device		REV: 1.0
Company: FIT VUT Brno		Sheet: 1/1
Date: 2021-02-09	Drawn By: Filip Jasek	



Příloha B

Snímky některých obrazovek aplikace

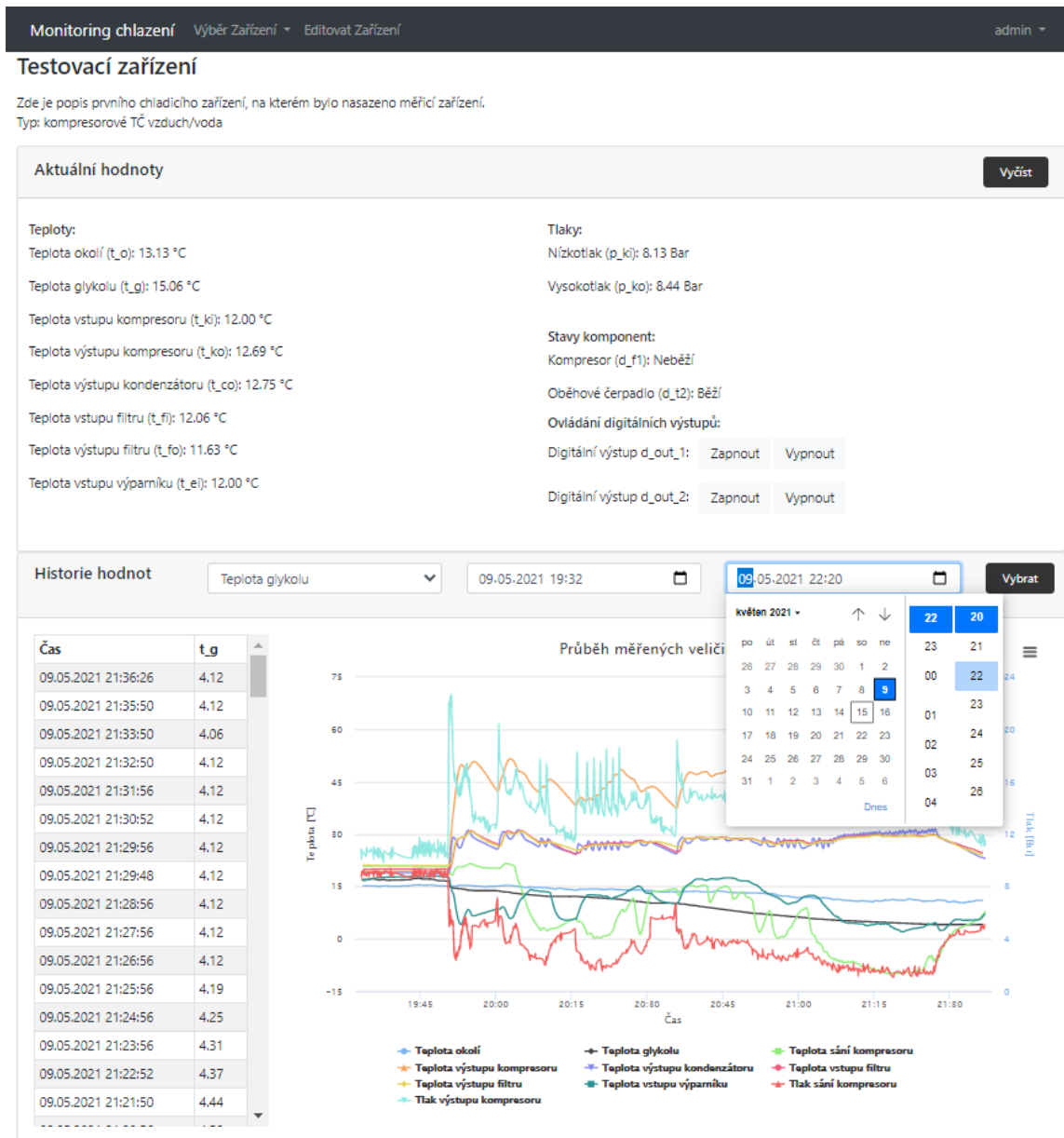


Obrázek B.1: Přihlašovací obrazovka

Přehled sledovaných chladicích zařízení

Testovací zařízení	
Teploty: Teplota okolí (t_o): 13.13 °C Teplota glykolu (t_g): 15.06 °C Teplota vstupu kompresoru (t_ki): 12.00 °C Teplota výstupu kompresoru (t_ko): 12.69 °C Teplota výstupu kondenzátoru (t_co): 12.75 °C Teplota vstupu filtru (t_fi): 12.06 °C Teplota výstupu filtru (t_fo): 11.63 °C Teplota vstupu výparníku (t_ei): 12.00 °C	Tlaky: Nízkotlak (p_ki): 8.13 Bar Vysokotlak (p_ko): 8.44 Bar Stavy komponent: Kompresor (d_f1): Neběží Oběhové čerpadlo (d_t2): Běží
Druhé testovací.	
Teploty: Teplota okolí (t_o): 15.00 °C Teplota glykolu (t_g): 23.00 °C Teplota vstupu kompresoru (t_ki): -2.56 °C Teplota výstupu kompresoru (t_ko): 53.60 °C Teplota výstupu kondenzátoru (t_co): 42.60 °C Teplota vstupu filtru (t_fi): 42.30 °C Teplota výstupu filtru (t_fo): 41.48 °C Teplota vstupu výparníku (t_ei): 6.32 °C	Tlaky: Nízkotlak (p_ki): 5.89 Bar Vysokotlak (p_ko): 21.33 Bar Stavy komponent: Kompresor (d_f1): Neběží Oběhové čerpadlo (d_t2): Neběží

Obrázek B.2: Základní obrazovka – přehled všech sledovaných zařízení.



Obrázek B.3: Detail vybraného zařízení s možností prohlížení historie měření.

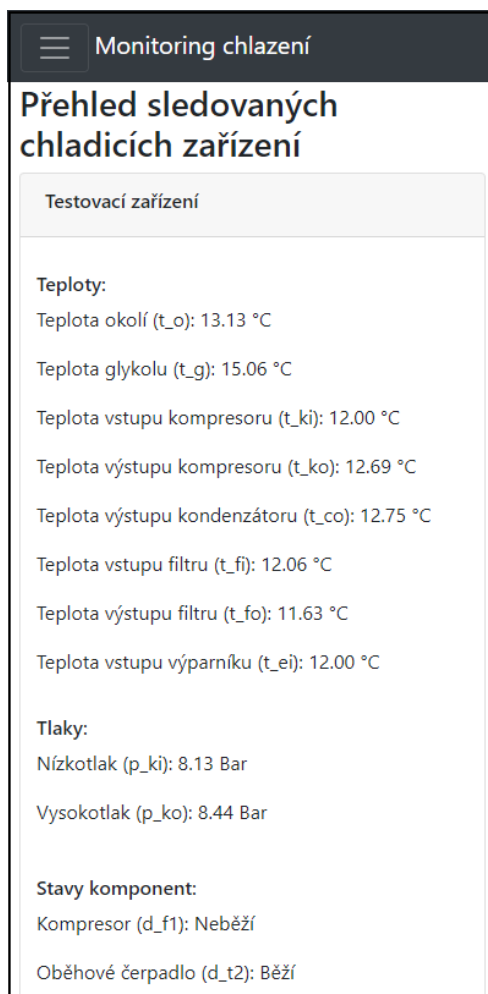
Monitoring chlazení Výběr Zařízení ▾ Editovat Zařízení admin ▾

Úprava zařízení Testovací zařízení

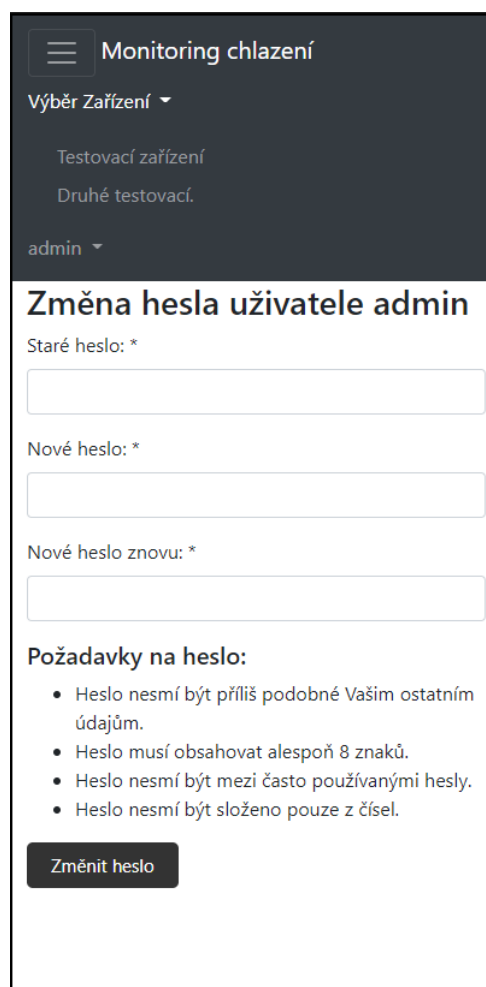
Název zařízení *	Popis zařízení (poznámky, atp...)
<input type="text" value="Testovací zařízení"/>	<div style="border: 1px solid #ccc; padding: 5px;">Zde je popis prvního chladicího zařízení, na kterém bylo nasazeno měřicí zařízení. Typ: kompresorové TČ vzduch/voda</div>
Interval zápisu všech hodnot [ms] *	
<input type="text" value="120000"/>	
Interval výpočtu delt [ms] *	
<input type="text" value="4000"/>	
Minimální krok teploty pro zápis *	
<input type="text" value="0,20"/>	
Minimální krok tlaku pro zápis *	
<input type="text" value="0,50"/>	

Ad. Interval výpočtu delt - interval měření hodnot; při změně větší, než je minimální krok (Minimální krok teploty (resp. tlaku) pro zápis), bude hodnota zapsána.

Obrázek B.4: Úprava údajů vybraného zařízení



Základní obrazovka



Menu a změna hesla

Obrázek B.5: Snímky aplikace zobrazené na mobilním zařízení

Příloha C

Obsah přiloženého CD

/	
doc/	Zdrojové L ^A T _E X soubory této práce.
firmware/	Soubory PlatformIO projektu.
src/	Zdrojový soubor firmware zařízení.
web_app/	Soubory Django projektu.
src/	Zdrojové soubory webové aplikace.
demo.mkv	Video s ukázkou práce v aplikaci.
thesis.pdf	Text této práce ve formátu PDF.