



**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 AUTOMATICKÉ ŘÍZENÍ A MONITOROVÁNÍ CENTRÁLNÍHO VYTÁPĚNÍ**

SYSTEM FOR AUTOMATED MONITORING AND CONTROL OF CENTRAL HEATING

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. MATĚJ ŠOC**

**VEDOUcí PRÁCE**

SUPERVISOR

**Ing. VÁCLAV ŠIMEK**

BRNO 2019

## Zadání diplomové práce



22166

Student: **Šoc Matěj, Bc.**  
Program: Informační technologie Obor: Počítačové a vestavěné systémy  
Název: **Systém pro automatické řízení a monitorování centrálního vytápění**  
**System for Automated Monitoring and Control of Central Heating**  
Kategorie: Vestavěné systémy

Zadání:

1. Seznamte se s existujícími systémy pro automatizaci a řízení vytápění. Zpracujte studii zaměřenou na dané téma.
2. Prostudujte problematiku bezdrátové komunikace sensorových prvků a jejich napájení. Diskutujte možnosti použitelné pro systém automatizace a řízení vytápění.
3. Vytvořte návrh architektury systému, který bude provádět automatické řízení centrálního vytápění. Systém bude obsahovat moduly pro ovládání topných těles, centrální řídicí prvek na platformě Raspberry Pi a modul pro ovládání kotle.
4. Na obvodové úrovni navrhnete prototypy systémových modulů pro ovládání ventilu topného tělesa a ovládání kotle. Tyto moduly budou vybaveny bezdrátovým komunikačním rozhraním.
5. Proveďte fyzickou realizaci jednotlivých modulů a implementujte obslužný firmware. Dále proveďte implementaci mobilní aplikace pro vzdálené ovládání a monitorování systému.
6. Demonstrujte funkčnost navrženého systému a zhodnoťte dosažené výsledky. Pokuste se navrhnout případná rozšíření či vylepšení.

Literatura:

- Dle pokynů vedoucího.

Při obhajobě semestrální části projektu je požadováno:

- Splnění bodů 1 až 3 zadání.

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

Vedoucí práce: **Šimek Václav, Ing.**  
Vedoucí ústavu: Sekanina Lukáš, prof. Ing., Ph.D.  
Datum zadání: 1. listopadu 2018  
Datum odevzdání: 22. května 2019  
Datum schválení: 9. května 2019

## Abstrakt

Tato diplomová práce se zabývá návrhem a následnou realizací bezdrátového systému pro automatické řízení a monitorování centrálního vytápění. Nejprve jsou ve stručnosti představena některá z existujících řešení a jejich vlastnosti. Dále je v této práci věnována pozornost analýze bezdrátových technologií a metodám napájení použitelných v oblasti automatizace domácnosti. S ohledem na získané poznatky byl proveden návrh a následná realizace systémových modulů s využitím miniaturního počítače Raspberry Pi, mikrokontroléru architektury ARM Cortex-M0+ a komunikačních modulů využívajících technologii ZigBee. Pro vytvořené moduly byl navrhnout a implementován obslužný firmware. Ovládání a monitorování systému lze provádět prostřednictvím mobilní aplikace pro platformu Android, jejíž návrh je rovněž popsán v této práci.

## Abstract

This thesis deals with design and implementation of wireless system for automated control and monitoring of central heating. Some of the existing solutions and their characteristics are briefly introduced at first. The attention in this thesis is subsequently given to the analysis of wireless technologies and power supply methods suitable for home automation. With regard to ascertained findings, the design and subsequent implementation of system modules using Raspberry Pi, the ARM Cortex-M0+ microcontroller and the ZigBee communication modules were performed. The firmware for created modules was designed and implemented. The designed system can be controlled and monitored using a mobile application for the Android platform, whose design is also described in this thesis.

## Klíčová slova

Automatizace, Raspberry Pi, Centrální vytápění, ZigBee

## Keywords

Automation, Raspberry Pi, Central heating, ZigBee

## Citace

ŠOC, Matěj. *Systém pro automatické řízení a monitorování centrálního vytápění*. Brno, 2019. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Václav Šimek

# System pro automatické řízení a monitorování centrálního vytápění

## Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Václava Šimka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Matěj Šoc  
20. května 2019

## Poděkování

Děkuji vedoucímu mé diplomové práce panu Ing. Václavu Šimkovi za odborné konzultace při návrhu a výrobě hardwarových modulů a užitečné připomínky k tvorbě práce. Rád bych také poděkoval mé rodině za podporu během celého studia.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>4</b>
<b>2</b>	<b>Existující řešení</b>	<b>5</b>
2.1	Honeywell Evohome . . . . .	5
2.2	FLEXIM . . . . .	6
<b>3</b>	<b>Bezdrátové technologie</b>	<b>8</b>
3.1	ZigBee . . . . .	8
3.1.1	Protokolový zásobník . . . . .	9
3.1.2	Topologie sítě . . . . .	10
3.2	Bluetooth . . . . .	12
3.2.1	Piconet a scatternet . . . . .	12
3.2.2	Protokolový zásobník . . . . .	13
3.2.3	Zabezpečení . . . . .	14
3.3	Wi-Fi . . . . .	14
3.3.1	Architektura sítě . . . . .	15
3.3.2	Protokolový zásobník . . . . .	16
3.3.3	Zabezpečení . . . . .	16
3.4	Shrnutí . . . . .	16
<b>4</b>	<b>Metody napájení bezdrátových senzorů</b>	<b>18</b>
4.1	Ukládání energie . . . . .	18
4.1.1	Akumulátory . . . . .	18
4.1.2	Superkondenzátory . . . . .	19
4.2	Energy harvesting . . . . .	20
<b>5</b>	<b>Návrh architektury systému</b>	<b>21</b>
5.1	Řídicí systém . . . . .	21
5.2	Centrální jednotka . . . . .	22
5.2.1	Raspberry Pi . . . . .	22
5.2.2	Schéma zapojení a deska plošných spojů . . . . .	23
5.3	Hlavice otopného tělesa . . . . .	23
5.3.1	Bezdrátové rozhraní . . . . .	23
5.3.2	Ovládání ventilu otopného tělesa . . . . .	25
5.3.3	Senzor teploty . . . . .	25
5.3.4	Ovládací prvky . . . . .	27
5.3.5	MCU . . . . .	28
5.3.6	Napájení . . . . .	29

5.3.7	Blokové schéma . . . . .	30
5.3.8	Schéma zapojení . . . . .	30
5.3.9	Deska plošných spojů . . . . .	32
5.4	Modul pro ovládání zdroje tepla . . . . .	32
5.4.1	Schéma zapojení . . . . .	33
5.4.2	Deska plošných spojů . . . . .	33
<b>6</b>	<b>Návrh a implementace řídicí aplikace</b>	<b>35</b>
6.1	Hlavní proces . . . . .	35
6.1.1	Úložiště dat . . . . .	36
6.1.2	MySQL API . . . . .	36
6.2	Vlákno pro komunikaci se ZigBee moduly . . . . .	36
6.2.1	Komunikační protokol . . . . .	37
6.2.2	Stanovení požadované teploty v místnosti . . . . .	38
6.3	Vlákno pro komunikaci s mobilní aplikací . . . . .	40
6.3.1	Komunikační protokol . . . . .	40
<b>7</b>	<b>Návrh a implementace firmware systémových modulů</b>	<b>42</b>
7.1	Hlavice otopného tělesa . . . . .	42
7.1.1	Princip činnosti . . . . .	43
7.1.2	Komunikace se ZigBee modulem . . . . .	43
7.1.3	Komunikace s centrální jednotkou . . . . .	44
7.1.4	Regulace vytápění . . . . .	45
7.1.5	Režim úspory elektrické energie . . . . .	45
7.1.6	Uživatelské rozhraní . . . . .	46
7.2	Modul pro ovládání zdroje tepla . . . . .	47
7.2.1	Princip činnosti . . . . .	48
<b>8</b>	<b>Návrh a implementace ovládací aplikace</b>	<b>50</b>
8.1	Technologie . . . . .	50
8.1.1	Struktura aplikace systému Android . . . . .	51
8.2	Backend . . . . .	51
8.2.1	Komunikace s centrální jednotkou . . . . .	51
8.2.2	Správa přihlašovacích údajů . . . . .	52
8.3	Frontend . . . . .	53
8.3.1	Přehled . . . . .	53
8.3.2	Nastavení . . . . .	55
<b>9</b>	<b>Použití systému</b>	<b>56</b>
9.1	Uživatelský test . . . . .	56
9.1.1	Průběh testu . . . . .	56
9.1.2	Shrnutí . . . . .	58
9.2	Spotřeba elektrické energie . . . . .	58
9.3	Optimalizace a rozšíření . . . . .	59
<b>10</b>	<b>Závěr</b>	<b>60</b>
	<b>Literatura</b>	<b>62</b>

<b>A</b>	<b>Obsah přiloženého paměťového média</b>	<b>64</b>
<b>B</b>	<b>Schémata systémových modulů</b>	<b>65</b>
<b>C</b>	<b>Snímky systémových modulů</b>	<b>68</b>
<b>D</b>	<b>Snímky prototypu hlavice otopného tělesa</b>	<b>70</b>
<b>E</b>	<b>ER diagram databáze</b>	<b>72</b>
<b>F</b>	<b>Snímky obrazovek mobilní aplikace</b>	<b>73</b>
<b>G</b>	<b>Naměřená data během uživatelského testu</b>	<b>75</b>

# Kapitola 1

## Úvod

Automatizace byla v minulosti zejména doménou průmyslu, ovšem postupně se rozšiřuje i do dalších oblastí, mezi které patří i vytápění domácnosti. Hlavní předností pokročilých systémů pro řízení centrálního vytápění je snížení nákladů a zvýšení uživatelského komfortu. Mezi další výhody patří schopnost nezávislé regulace teploty ve více místnostech současně, dynamická adaptace na aktuální potřeby uživatele či okolní prostředí a v neposlední řadě také možnost ovládání a monitorování systému prostřednictvím Internetu.

Cílem práce je návrh a realizace systému, který bude provádět automatické řízení centrálního vytápění. Systém bude rovněž umožňovat monitorování a konfiguraci prostřednictvím mobilní aplikace. Řízení centrálního vytápění je v kontextu této práce chápáno jako nezávislé ovládání otopných těles podle aktuální teploty okolí a uživatelského nastavení. Řízení vytápění zahrnuje také ovládání zdroje tepla, tedy plynového či elektrického teplovodního kotle, v závislosti na požadavcích systému či potřebách uživatele.

Kapitola 2 je věnována studii existujících řešení. Tato studie slouží jako výchozí bod návrhu systému. V následující kapitole 3 jsou popsány bezdrátové komunikační technologie použitelné pro komunikaci jednotlivých prvků systému. Navazující kapitola 4 pojednává o technikách získávání a ukládání elektrické energie v oblasti bezdrátových senzorů. Probrány jsou baterie a superkondenzátory, jelikož se jedná o nejrozšířenější prostředky pro ukládání elektrické energie. Nastíněno je také využívání alternativních zdrojů energie.

Návrh architektury systému a popis jednotlivých komponent je uveden v kapitole 5. V této kapitole je také popsána obvodová realizace modulů systému. V následujících kapitolách je pozornost věnována aspektům týkajících se implementace softwarového vybavení systému. Implementace řídicí aplikace je náplní kapitoly 6. Následující kapitola 7 pojednává o tvorbě obslužného firmware modulu hlavičky ventilu otopného tělesa a modulu pro ovládání zdroje tepla. Poslední z kapitol věnujících se implementačním záležitostem na úrovni obslužného firmware je kapitola 8, ve které je popsán návrh a stěžejní části kódu mobilní aplikace sloužící pro interakci systému s uživatelem.

Demonstrace funkčnosti a popis použití navrženého a realizovaného systému se nachází v kapitole 9.



## Kapitola 2

# Existující řešení

Tuzemský trh nabízí mnoho systémů pro řízení centrálního vytápění, které se navzájem liší po stránce implementovaných funkcí, rozšiřitelnosti, variability komponent a ceny. Populárním řešením jsou tzv. termostatické či elektronické hlavice instalované na ventily otopných těles. Tyto hlavice na základě okolní teploty, případně i času, otevírají či uzavírají ventil tělesa. Předností tohoto řešení je jednoduchost a nízká cena. Hlavice ovšem neumožňují řídit činnost zdroje tepla, proto se instalují spíše jako doplněk tradičního termostatu.

Další skupinou jsou systémy využívající centralizované řízení a sadu senzorů a aktivních prvků pro ovládání centrálního vytápění domácnosti. Ve zbylé části této kapitoly bude rozebráno několik zástupců této skupiny.

### 2.1 Honeywell Evohome

Systém Honeywell Evohome slouží ke komfortní regulaci vytápění domácnosti. Může být použit ve spojení s klasickými otopnými tělesy, podlahovým vytápěním či přímotopy. Systém Evohome je do jisté míry modulární a zahrnuje širokou škálu ovládacích a akčních prvků. Je ho tedy možné nakonfigurovat dle specifických požadavků uživatele. Tento systém umožňuje tzv. multizónovou regulaci teploty – nezávislá regulace teploty v jednotlivých zónách, respektive místnostech. Vytápění lze spravovat také vzdáleně prostřednictvím Internetu pomocí mobilního zařízení s operačním systémem Android či iOS.

Hlavním prvkem systému je řídicí jednotka. Řídicí jednotka slouží k ovládání regulace vytápění, rovněž se jejím prostřednictvím provádí nastavení programu vytápění jednotlivých zón v daný den a čas. Dle výrobce je možné takto nastavit program pro nejvýše dvanáct nezávislých zón. Již vytvořené programy lze zkopírovat a využít pro další zóny. Řídicí jednotka disponuje, kromě klasických programů vytápění, také pokročilými funkcemi, jako je například vyhodnocení tepelné setrvačnosti zón. Díky této funkci může být stanovena doba potřebná pro vytopení zóny na požadovanou teplotu, vytápění tedy může být spuštěno s optimální časovou rezervou. Řídicí jednotka komunikuje s ostatními prvky, a získává tak informace o aktuální teplotě v jednotlivých zónách. Získaná data vyhodnocuje a ovládáním akčních prvků reguluje vytápění. Komunikace s prvky probíhá bezdrátově na frekvenci 868 MHz s využitím proprietárního komunikačního protokolu. Jednotka je vybavena barevným dotykovým displejem, pomocí kterého probíhá interakce s uživatelem. Dále je vybavena Wi-Fi modulem pro připojení k Internetu. Napájení jednotky je řešeno prostřednictvím externího adaptéru

Mezi prvky systému Evohome patří i hlavice pro řízení ventilu otopných těles. Hlavice je vybavena LCD displejem, na kterém je zobrazována aktuální a nastavená teplota v dané zóně. Hlavice disponuje ovládacím prvkem v podobě otočného disku, který slouží k lokálnímu přenastavení teploty vytápění. Toto nastavení je deaktivováno s následující změnou v programu vytápění. Hlavice jsou schopny detekovat větrání (otevřené okno), kdy je v dané zóně pozastaveno vytápění za účelem úspory energie. Přítomna je také funkce ochrany otopných těles proti zamrznutí či zatuhnutí ovládacích ventilů. Hlavice jsou napájeny dvojicí článků typu AA. Výrobce uvádí, že doba provozu hlavice na jednu sadu článku může přesahovat 2 roky.

Dalším prvkem systému je spínací jednotka, jež může být použita k řízení zdroje tepla, řízení ohřevu teplé vody či spínání přímotopného tělesa. Činnost spínací jednotky je řízena centrální jednotkou. Spínací jednotka je napájena přímo ze sítě 230 V. Dalším, nikoli však posledním, prvkem systému Evohome je tzv. ovladač teploty, sloužící ke snímání a nastavování teploty v dané zóně. Tento ovladač je opět bezdrátový, napájený dvěma články typu AA [11, 10].

## 2.2 FLEXIM

Česká firma Softlink s.r.o. se pohybuje v oblasti měření a regulace za použití bezdrátových technologií. Poskytuje řešení pro vzdálené monitorování spotřeby vody a energií. Dále se věnuje vývoji vlastního SW a HW pro internet věcí (dále jen IoT) a v neposlední řadě také systému pro multizónovou regulaci vytápění FLEXIM. Systém FLEXIM je určen zejména pro administrativní budovy, školy či rodinné domy s centrálním vytápěním. Ovládání a monitorování systému je možné provádět pomocí webové aplikace v rámci lokální sítě či Internetu.

Mezi komponenty systému FLEXIM patří řídicí jednotka, komunikační brána, bezdrátový referenční teploměr, bezdrátový regulátor a regulační hlavice. Řízení systému zajišťuje centrální jednotka – jedná se o průmyslový počítač, na kterém je spuštěna řídicí aplikace. Připojení jednotky do lokální sítě, a tedy i do systému, je realizováno prostřednictvím rozhraní Ethernet. V případě potřeby může být řídicí jednotka suplována vzdáleným serverem dostupným z Internetu.

Dalším prvkem systému je komunikační brána, sloužící pro zprostředkování komunikace mezi řídicí jednotkou a bezdrátovými prvky systému. Brána je připojena do lokální sítě pomocí rozhraní Ethernet. Komunikace mezi bezdrátovými prvky probíhá prostřednictvím tzv. Wireless Automatic Collector (dále jen WACO) sítě. Jedná se o síť s mřížkovou topologií využívající frekvenční pásmo 868 MHz. Data jsou přenášena ve sběrnicovém formátu, díky čemuž lze bezdrátovou WACO síť snadno kombinovat s prvky využívajícími klasickou sběrnici – např. M-BUS nebo RS-485. Komunikační brána naslouchá na rozhraní WACO sítě a provádí kontrolu příchozích paketů. Zkontrolované pakety zapouzdřuje do UDP datagramů, které prostřednictvím lokální sítě adresuje řídicí jednotce.

Jedním z bezdrátových prvků je teploměr určený pro měření referenční teploty vzduchu v dané zóně (místnosti). Teploměr automaticky vysílá zprávy o aktuální teplotě na adresu broadcastu sítě v pětiminutových intervalech. Zaslání zprávy lze také vyžádat pomocí speciálního typu zpráv. Modul teploměru je založen na mikrokontroléru z řady MSP430<sup>1</sup> pracujícím na frekvenci 25 MHz. Napájení modulu je řešeno integrovaným lithiovým akumulátorem s kapacitou 2000 mAh, jehož životnost je dle výrobce nejméně 2 roky.

<sup>1</sup><http://www.ti.com/microcontrollers/msp430-ultra-low-power-mcus/overview.html>

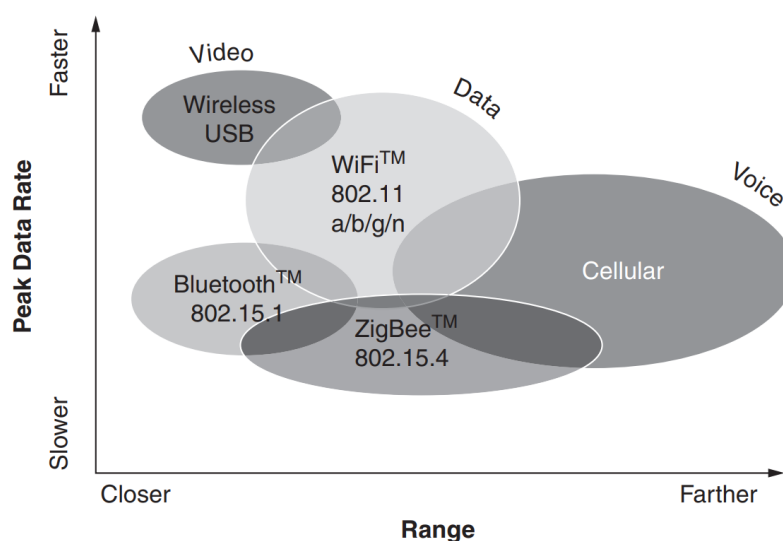
Dalším bezdrátově řízeným prvkem je regulátor. Regulátor prostřednictvím bezdrátového rozhraní přijímá příkazy od řídicí jednotky, na základě kterých ovládá regulační hlavice instalované na otopných tělesech. Regulátor má modulární koncepci, v základní podobě je osazen jedním výkonovým modulem, který je vybaven čtyřmi digitálními výstupy pro řízení až čtyř nezávislých zón. V případě potřeby je možné regulátor rozšířit pomocí tzv. expandérů, kdy každý expandér disponuje opět čtyřmi výstupy. Na každém výstupu regulátoru mohou být připojeny až 4 regulační hlavice. Pracovní napětí hlavic je 24 V, při ztrátě napájení se hlavice vrací do výchozího stavu [20, 21].

## Kapitola 3

# Bezdrátové technologie

Bezdrátové technologie umožňují komunikaci dvou a více zařízení bez nutnosti jejich fyzického spojení. Tyto technologie nejčastěji využívají jako nosné médium rádiové vlny. Rádiové vlny umožňují vytvářet spojení dlouhá od několika metrů až po miliony kilometrů. Mezi další používaná média patří například světelné záření či mechanické vlnění. Bezdrátové technologie nalezy uplatnění takřka ve všech odvětvích, kde doplňují nebo úplně nahrazují fyzické spoje.

Tato kapitola pojednává o bezdrátových technologiích použitelných v oblasti bezdrátových senzorů a automatizace. Probrány budou technologie Wi-Fi, Bluetooth a ZigBee. Srovnání jednotlivých technologií z hlediska dosahu a přenosové rychlosti je uvedeno na obrázku 3.1.



Obrázek 3.1: Srovnání bezdrátových technologií [7]

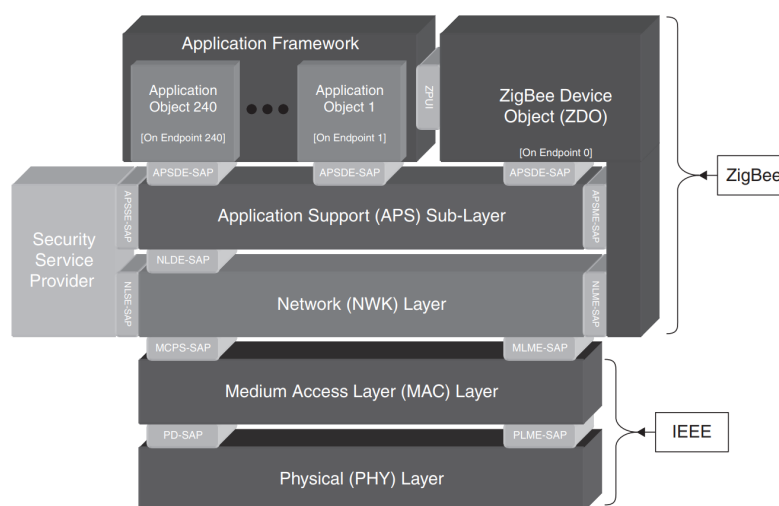
### 3.1 ZigBee

ZigBee je bezdrátový komunikační standart zaměřený na oblast automatizace, řízení a monitorování zařízení s nízkým příkonem. ZigBee umožňuje komunikaci mezi vysokým počtem zařízení na středně vysoké vzdálenosti. Oproti jiným bezdrátovým technologiím, jako napří-

klad Wi-Fi nebo Bluetooth, ZigBee necílí na co možná nejvyšší přenosovou rychlost. Naopak je zaměřen na jednoduchost, spolehlivost, bezpečnost a nízkou energetickou náročnost. Tyto vlastnosti činí ze ZigBee ideální technologii pro průmyslovou či domácí automatizaci, spotřební elektroniku a IoT. ZigBee je navržen pro aplikace, ve kterých je kladen důraz na velmi nízkou spotřebu elektrické energie, a kde jednotlivé prvky potřebují přenášet pouze malé objemy dat. ZigBee pracuje v bezlicenčních pásmech ISM, jejichž frekvence je závislá na lokalitě. Pro globální použití je určeno pásmo na 2,4 GHz – k dispozici je 16 kanálů a maximální přenosová rychlost je 250 kb/s. Současně se jedná o nejčastěji komerčně využívané pásmo. Pro Ameriku je určeno pásmo 915 MHz s maximální přenosovou rychlostí 40 kb/s a 10 kanály. Poslední pásmo je určeno pro Evropu. V evropském pásmu je k dispozici pouze jediný kanál a maximální přenosová rychlost je 20 kb/s [7].

### 3.1.1 Protokolový zásobník

Standard ZigBee je postaven na vrstvách PHY a MAC, které jsou definovány standardem IEEE 802.15.4. Tyto vrstvy zajišťují funkcionalitu fyzické a linkové vrstvy OSI modelu. Nad těmito vrstvami standard ZigBee definuje síťovou vrstvu NWK a aplikační vrstvu APL [9]. Protokolový zásobník ZigBee je zobrazen na obrázku 3.2.



Obrázek 3.2: Protokolový zásobník ZigBee [7]

### Aplikační vrstva APL

Aplikační vrstva se skládá z pomocné aplikační podvrstvy APS, ZigBee objektů ZDO a aplikačních objektů specifikovaných výrobci. Hlavní funkcí aplikační podvrstvy je udržování vazebních tabulek, které umožňují propojení dvou koncových zařízení. Rovněž zajišťuje předávání zpráv ze síťové vrstvy aplikačním objektům na základě jejich identifikátoru. Další funkcí podvrstvy APS je správa bezpečnostních klíčů či zajištění skupinové komunikace zařízení.

ZigBee objekt zajišťuje správu zařízení a služeb v síti – určuje roli daného zařízení v síti, vytváří zabezpečené spojení mezi dvěma síťovými zařízeními a odesílá či přijímá žádosti o spojení. ZDO spolupracuje také se síťovou vrstvou. Určuje, kdy se má zařízení připojit k síti a kdy ji opustit. Poskytuje rovněž aplikační rozhraní síťové vrstvy [9].

## Síťová vrstva NWK

Hlavní funkcí síťové vrstvy je sestavení sítě a poskytnutí funkčního rozhraní mezi aplikační vrstvou a MAC vrstvou. Tato vrstva zajišťuje směrování a přepínání paketů, vyhledávání přímých sousedů a ukládání informací o nich. V určitých případech také zajišťuje komunikaci a přiděluje adresy novým zařízením v síti. Pro udržování funkční sítě musí být zajištěno rozesílání broadcast paketů a hledání optimální cesty mezi dvěma komunikujícími uzly pro přeposílání unicast paketů. Zabezpečení komunikace rovněž probíhá na síťové vrstvě, kompletní obsah NWK rámce je šifrován [9].

## Vrstva MAC a fyzická vrstva PHY

Vrstva MAC (Medium Access Control) slouží jako rozhraní mezi síťovou a fyzickou vrstvou, tedy řídí přístup k fyzické vrstvě. Zajišťuje synchronizaci, zabezpečuje přístup na radiový kanál, kontroluje platnost rámce a potvrzuje příjem rámce. Dále řídí spojení a provádí generování a rozpoznávání adres. Pro komunikaci jsou definovány celkem čtyři typy rámců, tyto rámce mohou být datové nebo řídicí.

- *Data Frame* – pomocí tohoto rámce jsou realizovány veškeré datové přenosy. Prostřednictvím jednoho datového rámce může být přeneseno až 104 B dat.
- *Acknowledgment Frame* – slouží pro odeslání potvrzení o úspěšném přijetí rámce. Potvrzující pakety jsou zasílány ihned po přijetí rámce v nevyužitém čase mezi pakety.
- *MAC Command Frame* – rámeček sloužící k nastavení a řízení klientských zařízení v síti.
- *Beacon Frame* – slouží zejména k synchronizaci zařízení v síti. V tzv. *beacon-enabled* sítích je používán koordinátorem sítě pro uvedení klientských zařízení do režimu spánku, za účelem minimalizace spotřeby energie.

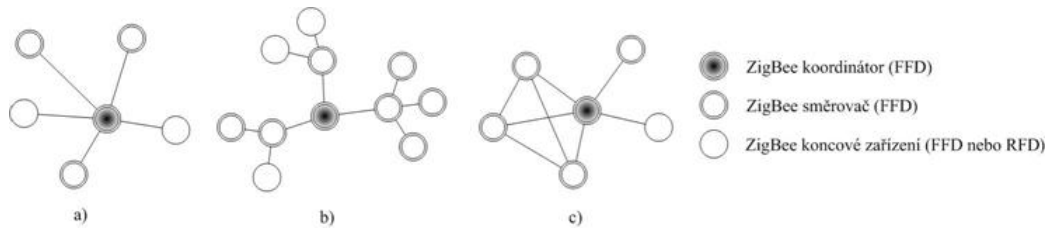
Fyzická vrstva vytváří rozhraní mezi MAC vrstvou a radiovým kanálem, a tedy zajišťuje vlastní přenos datových jednotek. Fyzická vrstva rovněž provádí vyhodnocení kvality spojení a výběr komunikačního kanálu. Komunikace probíhá, jak již bylo uvedeno výše, v některém ze tří bezlicenčních pásem: 868 MHz, 915 MHz, 2,4 GHz. Pro přístup ke sdílenému médiu je na této vrstvě implementována technika CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) [12].

### 3.1.2 Topologie sítě

Jednotlivá zařízení v síti ZigBee jsou adresována pomocí binárních kódů o délce 64 b v případě dlouhého kódu a 16 b v případě zkráceného kódu. Délka adresovacích kódů určuje maximální počet zařízení v síti. Rovněž každá síť má svůj jednoznačný 16b identifikátor PAN ID. Tento identifikátor je používán pro rozlišení komunikace v případech, kdy je v jednom prostoru provozováno více sítí využívajících standardu IEEE 802.15.4.

V síti ZigBee se mohou vyskytovat tři typy zařízení (uzlů): koordinátor, směrovač a koncové zařízení. Zařízení lze dále rozdělit na plně funkční zařízení (FDD) a redukováná zařízení (RFD). Plně funkční zařízení může zastávat funkci koordinátora, směrovače i koncového zařízení. Redukovaná zařízení může fungovat pouze v roli koncového zařízení. Každá síť ZigBee je řízena koordinátorem. ZigBee standardem jsou podporovány topologie typu hvězda, strom a síť (obrázek 3.3). V topologii typu hvězda komunikují všechna koncová zařízení přímo s koordinátorem. V topologii typu strom a síť mohou rovnocenná zařízení

komunikovat samostatně bez účasti koordinátora. Koordinátor typicky spouští komunikaci a stanovuje parametry sítě. Sít je možné rozšířit pomocí směrovačů [12].



Obrázek 3.3: Topologie sítě ZigBee [12]

## Zabezpečení

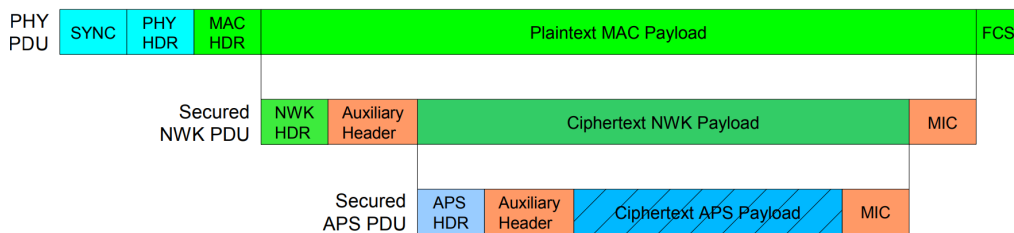
Technologie ZigBee zajišťuje integritu a důvěrnost přenášených dat. Zabezpečení je implementováno na síťové a aplikační vrstvě. Dostupné jsou celkem tři úrovně zabezpečení – nezabezpečený přístup, přístup na základě práv a zabezpečený přístup. Pokud je použit přístup na základě práv, jsou z komunikace vyfiltrovány rámce od neznámých zařízení. Pro zabezpečený přístup je více dostupných služeb, které mohou zařízení použít.

Standard ZigBee využívá model distribuce klíčů pomocí důvěryhodné autority, tzv. *trust center*. Bezpečnost je založena na symetrických klíčích (odesílatel i příjemce sdílí stejný bezpečnostní klíč), které přímo slouží k šifrování komunikace. Před začátkem komunikace musí obě komunikující strany disponovat bezpečnostním klíčem. Existují tři techniky získávání bezpečnostních klíčů. První možnost je předem nainstalovaný klíč, není tedy nutné klíče žádným způsobem přenášet. Druhou možností je distribuce klíčů od *trust center* ke komunikujícím zařízením. U této techniky je nutné také vhodným způsobem zabezpečit přenášený klíč. Poslední technikou je domluva komunikujících stran s *trust center*, na základě které jsou stanoveny klíče. Klíče není nutné přenášet.

V síti ZigBee jsou používány tři typy klíčů: *master klíč*, linkový klíč a síťový klíč. *Master klíče* jsou distribuovány zařízením *trust center* a slouží ke stanovení linkového klíče mezi dvěma zařízením. Rovněž je používán ke vzájemné autentizaci zařízení. Síťové klíče jsou používány zejména pro správu sítě. Mohou být nainstalovány předem, nebo je instaluje *trust center*. Síťové klíče sdílí všechna zařízení v síti. Linkové klíče jsou nastavovány na základě domluvy komunikujících stran, případně mohou být nastaveny pomocí *trust center* na žádost zařízení [9].

## Zabezpečení rámců

Standard ZigBee umožňuje zabezpečení rámců jak na síťové, tak i aplikační vrstvě. Zabezpečení se skládá ze dvou částí – šifrování a zajištění integrity. K zabezpečení je využíván algoritmus AES-CCM, pomocí kterého jsou šifrována data přenášená rámcem. Integrita je zajištěna přidáním speciální položky MIC (Message Integrity Code) na konec přenášených dat. Na MIC lze pohlížet jako na kontrolní součet přenášených dat, příjemce si tedy může ověřit, že data nebyla během přenosu neoprávněně modifikována. MIC zajišťuje také autentizaci komunikujících stran, jelikož je svázán s adresou zařízení [9]. Struktura zabezpečeného rámce je zobrazena na obrázku 3.4.



Obrázek 3.4: Zabezpečený rámec [4]

## 3.2 Bluetooth

Standard Bluetooth je určen pro bezdrátovou komunikaci mobilních i stacionárních zařízení na krátké vzdálenosti. Podporovány jsou hlasové i datové přenosy. Technologie Bluetooth je velmi spolehlivá a odolná vůči rušení – pro datové přenosy je používáno potvrzování paketů zaručující spolehlivý přenos dat, hlasové přenosy probíhají bez potvrzování. Bluetooth operuje v bezlicenčním pásmu ISM, konkrétně na frekvenci 2,4 GHz. Pro komunikaci je k dispozici celkem 79 kanálů. Dosah signálu se odvíjí od třídy zařízení definované standardem.

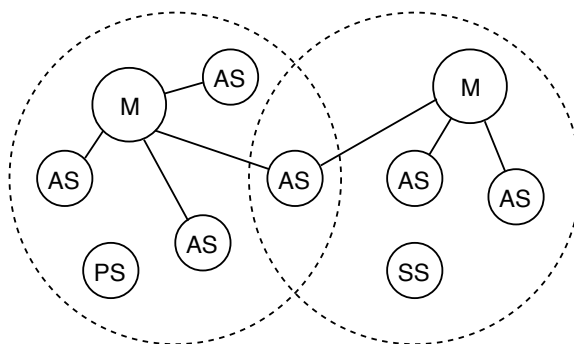
Pomocí Bluetooth lze vytvářet jak spojení typu *point-to-point* tak i spojení typu *point-to-multipoint*, které odpovídá komunikačnímu modelu *master-slave*. Síť složená ze zařízení komunikujících pomocí Bluetooth se nazývá piconet. V každé takové síti musí být přítomno právě jedno zařízení typu *master*. Jednotlivé sítě piconet lze propojovat a rozšiřovat tak výslednou síť.

### 3.2.1 Piconet a scatternet

Piconet je síť zařízení komunikujících pomocí Bluetooth. Síť piconet má topologii tvaru hvězda, v každém piconetu je tedy právě jedno zařízení v roli *master*, ke kterému jsou připojena další zařízení v roli *slave*. V síti se mohou vyskytovat také zařízení v režimu *parked* a *standby*. Zařízení v režimu *parked* jsou neaktivní, ovšem ostatní zařízení v síti o těchto zařízeních vědí, a v případě potřeby mohou být velmi rychle aktivována. Aktivováním přejde takovéto zařízení do role *slave*. Zařízení v režimu *standby* se žádným způsobem nepodílí na komunikaci v rámci daného piconetu, ale jsou v komunikačním dosahu. Adresování v rámci sítě piconet je prováděno pomocí binárních kódů (adres). Zařízení v roli *master* a *slave* jsou adresována 3b kódem, v jednom piconetu tedy může pracovat až osm aktivních zařízení, jedno zařízení typu *master* a sedm zařízení v roli *slave*. Neaktivní zařízení, tedy *parked* a *standby*, jsou identifikována 8b kódem, v rámci jednoho piconetu jich může být až 256.

Jednotlivé sítě piconet lze propojovat a vytvořit tak tzv. scatternet. Propojení piconetů může být realizováno dvěma různými způsoby. Prvním způsobem je propojení prostřednictvím zařízení, jež je v rámci obou piconetů v roli *slave*. Druhým způsobem je využití zařízení, které v rámci jednoho piconetu vystupuje jako *master* a ve druhém jako *slave*. Možnost, kdy by jedno zařízení vystupovalo jako *master* ve více než jednom piconetu, je nepřijatelná. Komunikace pak probíhá tím způsobem, že spojující zařízení přepíná mezi piconety [2]. Na obrázku 3.5 je uveden příklad scatternetu tvořeného propojením dvou piconetů.





Obrázek 3.5: Scatternet

### 3.2.2 Protokolový zásobník

Protokolový zásobník standardu Bluetooth poskytuje služby pro lokalizaci, navázání spojení a přenos dat. Dále umožňuje provozování různých typů aplikací s využitím různých kombinací protokolů. Všechny aplikace mají společnou fyzickou a linkovou vrstvu. Protokoly technologie Bluetooth lze rozdělit do tří skupin: transportní, *middleware* a aplikační. Protokoly transportní skupiny umožňují lokalizaci a spojení zařízení, zajišťují synchronní i asynchronní přenosy pro hlasovou komunikaci, spravují fyzické spojení mezi zařízeními, čímž vytváří rozhraní pro protokoly vyšších vrstev. Protokoly v *middleware* vrstvě umožňují fungování aplikací skrze Bluetooth spojení. Tyto protokoly nejsou definovány standardem Bluetooth. Aplikační vrstva obsahuje již samotné aplikace, které využívají Bluetooth spojení [19]. Kompletní architektura je zobrazena na obrázku 3.6. Nyní budou popsány nejvýznamnější části architektury Bluetooth.

#### Link Manager

Link Manager (dále jen LM) je systém zajišťující řízení spojení mezi dvěma zařízeními. Řídí proces vytvoření, autentizaci a konfiguraci spojení. Dále také rozhoduje o ukončení spojení, přepínání mezi jednotlivými rolemi a v neposlední řadě také řídí párování zařízení. LM jednoho zařízení vyhledává LM ostatních zařízení, se kterými následně komunikuje prostřednictvím LMP protokolu. Aby mohl LM korektně fungovat, využívá služeb Link Controlleru (dále jen LC). LC se vyskytuje přímo nad Baseband vrstvou a je zodpovědný za samotné vytvoření, udržování a ukončení spojení [19].

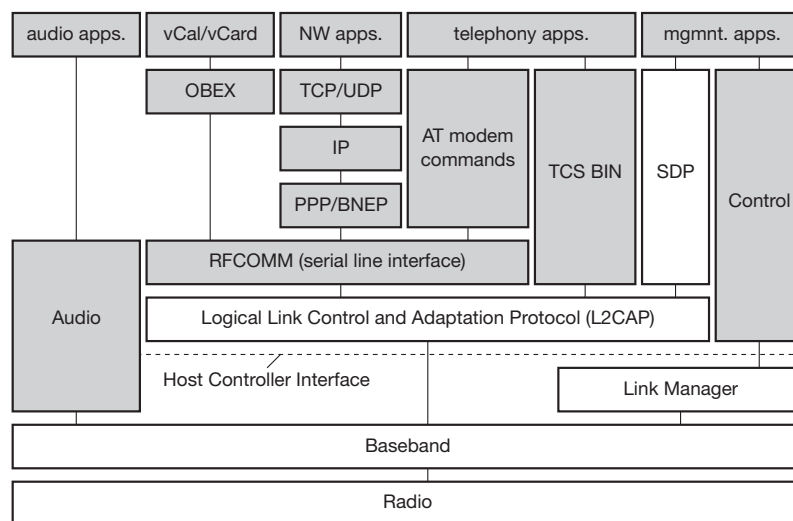
#### L2CAP (Logical Link Control and Adaptation Protocol)

Protokol L2CAP vytváří rozhraní mezi LM a vyššími vrstvami, které umožňuje současné provozování více logických spojení mezi dvěma zařízeními. Provádí fragmentaci a opětovné sestavení paketů protokolů vyšších vrstev na jednotky menší délky, aby mohly být přeneseny skrze bezdrátové spojení. Dále také implementuje zajištění kvality služeb (QoS) a skupinové adresování zařízení [19].

#### RFCOMM (Radio Frequency Communication)

Vrstva RFCOMM emuluje virtuální sériové rozhraní typu RS-232. Jedná se sadu transportních protokolů implementovaných nad L2CAP protokolem, která umožňuje propojení libovolné aplikace používající sériové rozhraní s technologií Bluetooth. Vrstva RFCOMM

je nezbytná například pro aplikace spojené s telefoníí pro přenos AT příkazů. Celkem je možné provozovat až 60 logických spojení mezi dvěma zařízeními současně [18].



Obrázek 3.6: Protokolový zásobník Bluetooth [19]

### 3.2.3 Zabezpečení

Mezi bezpečnostní služby, které standard Bluetooth zajišťuje, patří autentizace, důvěrnost a autorizace. Jiné bezpečnostní služby standard nedefinuje, ale v případě potřeby mohou být implementovány odděleně. První fází zabezpečení je tzv. párování, k této proceduře dochází pouze pokud spolu daná zařízení ještě nekomunikovala. Párování může být prováděno na základě PIN kódu nebo s využitím techniky *Secure Simple Pairing*.

Standard Bluetooth definuje celkem čtyři bezpečnostní režimy, ve kterých mohou zařízení pracovat. Tyto režimy specifikují, v jaké fázi komunikace je nasazeno zabezpečení, neudávají tedy používané bezpečnostní funkce. V případě režimu 1 nikdy nedochází k iniciaci zabezpečení. Při použití režimu 2 dochází k iniciování zabezpečení po úplném sestavení spojení, ke generování klíče spojení je využíván PIN. V režimu 3 je zabezpečení aktivováno ještě před úplným sestavením fyzického spojení zařízení. Zařízení v tomto režimu nemohou operovat bez autentizace, šifrování a autorizace. Posledním režimem je režim 4, který je velmi podobný režimu 2 s tím rozdílem, že pro generování klíče spojení je využívána metoda *Secure Simple Pairing*. Každé zařízení musí pracovat právě v jednom z těchto režimů [13].

## 3.3 Wi-Fi

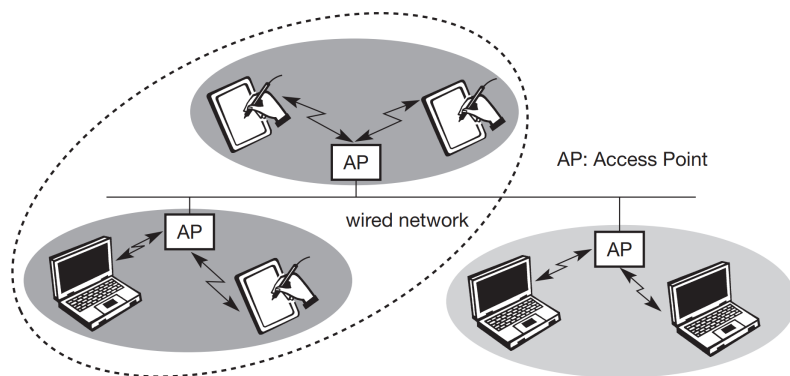
Wi-Fi je technologie pro vytvoření lokální bezdrátové sítě WLAN (Wireless Local Area Network) vycházející ze standardu 802.11 organizace IEEE. Tato technologie byla vyvinuta jako náhrada za drátové spoje, stala se tak prostředkem pro bezdrátový přístup k síti Internet a nabídla větší flexibilitu pro tzv. ad-hoc komunikaci. Hlavními výhodami jsou flexibilita, možnost vytváření ad-hoc sítí a robustnost [19]. Standard Wi-Fi definuje několik verzí lišících se dosahem, používaným frekvenčním pásmem a přenosovou rychlostí. V současnosti nejpoužívanější verze 802.11b využívá frekvenční pásmo 2,4 GHz. Standard 802.11b poskytuje přenosovou rychlost až 11 Mb/s. Verze 802.11a pracuje ve frekvenčním pásmu

5 GHz a poskytuje přenosovou rychlost až 54 Mb/s. Usilování o vyšší přenosovou rychlost standardu 802.11b mělo za následek vznik verze 802.11g. Verze 802.11g pracuje v pásmu 2,4 GHz a poskytuje přenosovou rychlost až 54 Mb/s. Verze 802.11n pracuje v obou zmiňovaných pásmech a umožňuje dosahovat přenosové rychlosti 600 Mb/s. Posledním komerčně využívaným standardem je 802.11ac, který využívá frekvenční pásmo 5 GHz a poskytuje přenosovou rychlost přesahující 1 Gb/s. Existují také další verze standardu Wi-Fi, které ale nejsou komerčně používány [14].

### 3.3.1 Architektura sítě

Sít vybudovaná za použití Wi-Fi technologie je vždy identifikována pomocí SSID. SSID je řetězec ASCII znaků o maximální délce 32 B. Přístupový bod, pokud není nastaven jinak, v pravidelných intervalech vysílá identifikátor na adresu broadcastu sítě, klientská zařízení tak mají informaci o dostupných sítích. Standard Wi-Fi podporuje dva základní typy sítí, infrastrukturní a ad-hoc.

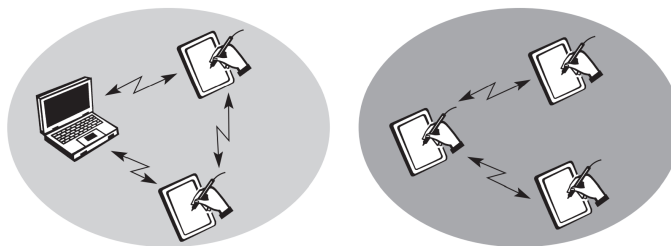
Infrastrukturní síť umožňuje přístup do jiných sítí, zajišťuje přeposílání a směrování paketů a také řízení přístupu ke sdílenému médiu. Zařízení v infrastrukturní síti nekomunikují přímo mezi sebou, ale využívají přístupový bod, který zajišťuje přeposílání komunikace. Přístupový bod může rovněž sloužit jako most do dalších drátových či bezdrátových sítí, je tedy možné spojovat více sítí do logické sítě, jež je větší než samotný dosah radiového signálu. Na obrázku 3.7 je uveden příklad infrastrukturní sítě tvořené třemi bezdrátovými sítěmi za pomoci tří přístupových bodů. Infrastrukturní sítě jsou obvykle jednodušší, jelikož většinu funkcí zajišťuje přístupový bod, a klientská zařízení tak mohou být relativně jednoduchá [19].



Obrázek 3.7: Infrastrukturní síť [19]

Ad-hoc sítě ke svému fungování nepotřebují žádnou infrastrukturu, zařízení mohou komunikovat přímo mezi sebou. Komunikace zařízení je možná pouze za předpokladu, že se nachází v dosahu radiového signálu. Absence přístupových bodů v ad-hoc sítích má za následek vyšší složitost klientských zařízení, jelikož musí implementovat funkce jako řízení přístupu ke sdílenému médiu či zajištění kvality služeb. Na obrázku 3.8 jsou zobrazeny příklady dvou ad-hoc sítí.

WLAN sítě nemusí být vždy striktně infrastrukturní nebo ad-hoc, jednotlivé přístupy lze v případě potřeby kombinovat.



Obrázek 3.8: Ad-hoc síť [19]

### 3.3.2 Protokolový zásobník

Standard 802.11 definuje pouze fyzickou vrstvu PHY a MAC (Medium Access Control) vrstvu. Vyšší vrstvy a jejich protokoly jsou shodné přes všechny 802.x standardy, a tedy z pohledu těchto vrstev není mezi drátovým a bezdrátovým spojením žádný rozdíl. Fyzická vrstva PHY se skládá z podvrstev PLCP (Physical Layer Convergence Protocol) a (Physical Medium Dependent). PLCP zajišťuje nosný signál pro přenášená data a zajišťuje služby fyzické vrstvy, které jsou nezávislé na technologii přenosu (rádiové vlny, infračervené vlnění). Podvrstva PMD zajišťuje modulaci, kódování a dekódování signálu. Vrstva MAC je zodpovědná za řízení přístupu ke sdílenému médiu, zajišťuje fragmentaci datových jednotek, šifrování přenášených dat a také řídí autentizaci [19].

### 3.3.3 Zabezpečení

U sítí LAN není implementováno žádné hardwarové či softwarové zabezpečení – jelikož se jedná o fyzické spojení zařízení, za dostatečnou ochranu je považováno omezení fyzického přístupu ke spoji. WLAN sítě touto vlastností však nedisponují, bylo tedy nutné implementovat zabezpečovací techniky. Bezpečnostní architektura standardu 802.11 poskytuje autentizaci, důvěrnost a integritu.

Základním bezpečnostním standardem používaným WLAN sítěmi je WEP (Wired Equivalent Privacy), tento standard byl však roku 2001 prolomen a v současnosti je považován za nedostatečný. Slabinami tohoto standardu je absence protokolu pro dohodnutí/zřízení klíčů, použití ne příliš vhodného synchronního šifrovacího algoritmu a také krátký inicializační vektor. Nástupcem standardu WEP je WPA (Wi-Fi Protected Access). Hlavní změnou je použití protokolu TKIP (Temporal Key Integrity Protocol), který slouží pro správu klíčů mezi přístupovým bodem a klientem. WPA využívá sice stejný šifrovací algoritmus jako WEP, ale díky protokolu pro distribuci klíčů a generování inicializačních vektorů je mnohem bezpečnější. Zdokonalen je také algoritmus pro zajištění integrity dat [3].

Nástupcem WPA je technologie zabezpečení WPA2, která plně implementuje standard 802.11i a v současnosti je považována za bezpečnou. Komunikace je zabezpečena protokolem CCMP používajícím blokovou šifru AES. Pro vytvoření bezpečné komunikace a distribuci klíčů slouží sada bezpečnostních protokolů EAP (Extensible Authentication Protocol).

## 3.4 Shrnutí

Technologie ZigBee je vhodná pro aplikace, ve kterých je kladen důraz na úsporu energie a jednoduchost zařízení. Mezi další rysy této technologie patří nízké náklady, zabezpečená

komunikace a schopnost tvorby spolehlivé síťové struktury. Oproti technologii Bluetooth či Wi-Fi poskytuje ZigBee o několik řádů nižší přenosovou rychlost.

Technologie Bluetooth umožňuje spolehlivé a bezpečné spojení různých typů zařízení na krátké vzdálenosti. Podporovány jsou hlasové i datové přenosy. Standard Bluetooth umožňuje vytvářet spojení typu *point-to-point* a také složitější spojení *point-to-multipoint*. Důležitými vlastnostmi této technologie je bezpečnost komunikace a nízká cena.

Wi-Fi technologie je určena pro vytváření lokálních bezdrátových sítí. Umožňuje vytvářet přímé spojení mezi dvěma zařízeními a rovněž rozsáhlejší infrastrukturní sítě. Výhodou je vysoká přenosová rychlost a dosah signálu. Nevýhodou může být cena a spotřeba elektrické energie.

## Kapitola 4

# Metody napájení bezdrátových senzorů

Způsob napájení bezdrátových senzorů je zásadním faktorem ovlivňujícím nejen dobu provozu, ale také rozměry, cenu a v neposlední řadě také četnost provádění údržby daného zařízení. Vývoj CMOS technologie má za následek neustálou miniaturizaci elektronických obvodů. Zmenšování obvodů ovšem dalece překonalo zvyšování energetické hustoty zdrojů elektrické energie, což má za následek skutečnost, že zdroje elektrické energie se stávají největšími a nejdražšími komponentami bezdrátových senzorů.

V této kapitole budou popsány techniky ukládání energie využitelné v oblasti bezdrátových senzorů. Rovněž bude probírána technika získávání energie z alternativní zdrojů známá jako energy harvesting.

### 4.1 Ukládání energie

Tradičním a současně nejpoužívanějším prostředkem pro uchování energie jsou baterie, respektive akumulátory. V senzorových sítích nalézají své uplatnění také tzv. superkondenzátory. Existují také alternativní řešení jako miniaturní palivové články či tepelné stroje. Tato řešení ovšem nejsou vhodná pro bezdrátové senzory z důvodu velmi malé efektivity přeměny energie a velkých rozměrů.

#### 4.1.1 Akumulátory

Akumulátory jsou nejrozšířenějším prostředkem pro ukládání elektrické energie. Jsou dostupné ve velkém množství variant lišících se kapacitou, rozměry, hmotností či způsobem nabíjení. Akumulátory se typicky vyznačují vysokou kapacitou a vysokou energetickou hustotou. Rozdíl napětí plně nabitého a vybitého akumulátoru je pouhým zlomkem jeho nominálního napětí, pokud tedy zařízení nevyžaduje stabilizovaný zdroj napětí, může být napájeno přímo z akumulátoru bez dalších obvodů. Mohou tak být sníženy rozměry výsledného zařízení a minimalizována spotřeba energie napájecí elektroniky. Nevýhodou (např. oproti kondenzátorům) je potřeba specifického průběhu nabíjení akumulátorů, což vyžaduje dodatečnou nabíjecí elektroniku [16].

Akumulátory využívají k uchování elektrické energie vratné chemické reakce. Jejich vlastnosti závisí na vnitřním chemickém složení. Nejvyšší energetické hustoty dosahují lithiové akumulátory (Li-Ion a Li-Pol). Lithiové akumulátory nemají tzv. paměťový efekt, disponují vysokou účinností a nízkým samovybíjením. Dalším rozšířeným typem jsou Ni-

Cd (nikl-kadmium) akumulátory. Ni-Cd akumulátory nabízí nižší energetickou hustotu, ale díky nízkému vnitřnímu odporu jsou schopné dodávat vysoký výkon. Podobného typu jsou akumulátory Ni-MH (nikl-metal hydrid), hojně používané v oblasti spotřební elektroniky. Poskytují větší energetickou hustotu než Ni-Cd akumulátory [17]. Detailní srovnání parametrů těchto akumulátorů je uvedeno v tabulce 4.1.

Technologie	Li-Ion	Li-Pol	Ni-MH	Ni-Cd
Energetická hustota [ $Wh \cdot L^{-1}$ ]	435	300-450	140-300	110-150
Samovybíjení (měsíc) [%]	5-10	1-2	1-3	20-30
Životnost (počet cyklů)	2000	1200	180-2000	500-2000
Pracovní napětí [V]	3-3,7	3,7	1,2	1,25

Tabulka 4.1: Srovnání akumulátorů [1] [17]

#### 4.1.2 Superkondenzátory

Kondenzátory využívají k uložení elektrické energie (náboje) elektrické pole – na rozdíl od akumulátorů, které využívají chemických reakcí. Hlavní rozdíly vůči akumulátorům jsou uvedeny níže:

- vyšší výkonová hustota, nižší energetická hustota,
- lineární průběh nabíjení a vybíjení,
- kratší doba nabíjení,
- téměř neomezená životnost a počet nabíjecích cyklů,
- nižší hmotnost.

Pro úplnost bude zjednodušeně vysvětlen princip fungování klasického kondenzátoru. Kondenzátor je tvořen dvěma elektrodami, které jsou odděleny dielektrickým materiálem. Pokud je na elektrody přivedeno napětí, na elektrodách se akumulují náboje vzájemně opačné polarity. Dielektrikum udržuje náboje oddělené, čímž vzniká elektrické pole umožňující uložení elektrické energie. Hlavním parametrem charakterizujícím kondenzátor je kapacita, ta je přímo úměrná ploše elektrod kondenzátoru. Dalším důležitým parametrem je ESR, tedy ekvivalentní sériový odpor. ESR limituje maximální výkon, který je kondenzátor schopen vydat.

Superkondenzátory se od klasických kondenzátorů liší vyšší kapacitou, respektive energetickou hustotou a také nižším limitem pracovního napětí. Superkondenzátory jsou založeny na stejném principu jako klasické kondenzátory, využívají ovšem elektrody s mnohem větší plochou a mnohem menší vzdáleností elektrod. Zvětšením plochy elektrod bylo dosaženo výrazného zvýšení energetické hustoty. Nicméně se snižující se vzdáleností elektrod klesá maximální hodnota pracovního napětí kondenzátoru.

Superkondenzátory mohou být z hlediska typu rozděleny do tří skupin: tzv. Electrochemical Double Layer Capacitors (EDLC), pseudokondenzátory a hybridní kondenzátory. EDLC kondenzátory tvoří uhlíkové elektrody, elektrolyt obsahující kladné i záporné ionty

a iontově propustná membrána. Přivedením napětí na elektrody dojde k pohybu iontů směrem k opačně nabitým elektrodám. Vlastnosti elektrod ovšem zabraňují rekombinaci iontů a na obou elektrodách vznikne dvojitá vrstva náboje. Tímto je zvětšena plocha a zároveň snížena vzdálenost elektrod [8].

Pseudokondenzátory využívají vratných oxidačně-redukčních reakcí na povrchu jedné z elektrod. Principem uchování energie se tedy blíží spíše akumulátorům. Oproti EDLC kondenzátorům se vyznačují nižší životností a nižší účinností nabíjení. Výhodou v porovnání s EDLC kondenzátory je typicky vyšší kapacita. [6].

Hybridní kondenzátory kombinují technologické postupy obou předcházejících variant, například kombinací elektrody s převážně elektrostatickou kapacitancí a elektrody s převážně elektrochemickou kapacitancí [8].

## 4.2 Energy harvesting

Energy harvesting je technika získávání elektrické energie z alternativních zdrojů v prostředí, kde je zařízení situováno. Tato technika našla uplatnění v průmyslu, medicínských aplikacích a také v komerčním sektoru. Rovněž je využívána v oblasti bezdrátových senzorů. Energie získaná technikou energy harvesting se vyznačuje časovou nestálostí a velmi malým dodávaným výkonem. Pro zvýšení efektivity je typicky využíváno více druhů zdrojů energie současně. Napájené zařízení musí mít nízkou spotřebu elektrické energie a režim jeho činnosti by měl umožňovat dlouhodobou akumulaci energie a její následné krátkodobé využití. Využívání alternativních zdrojů energie přináší následující výhody:

- prodloužení životnosti primárního zdroje napájení,
- zmenšení rozměrů primárního zdroje napájení,
- prodloužení doby mezi úkony údržby (např. výměna baterie).

Za určitých podmínek může být konvenční zdroj elektrické energie zcela nahrazen, a bezdrátová zařízení mohou být z hlediska napájení úplně nezávislá.

Využita může být kinetická energie, světelné záření, elektromagnetické pole, proudění vzduchu či teplotní gradient [23]. Srovnání jednotlivých technologií na základě produkovaného výkonu je uvedeno v tabulce 4.2.

Zdroj energie	Podmínky	Výkon [ $\mu W \cdot cm^{-2}$ ]
Kinetická energie	Vibrace $1 m \cdot s^{-2}$	100
Solární energie	Volné prostranství	7500
RF energie	GSM signál	0,1
Teplotní gradient	$\Delta T = 5 \text{ }^\circ\text{C}$	60
Proudění vzduchu	Turbíny	100

Tabulka 4.2: Srovnání energy harvesting technologií [1]



## Kapitola 5

# Návrh architektury systému

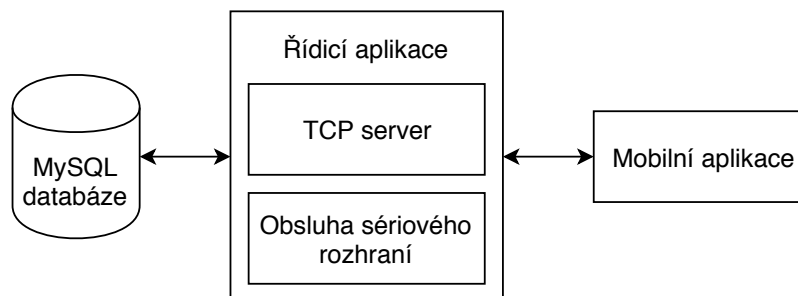
Systém pro řízení regulace vytápění zahrnuje softwarový řídicí systém, centrální jednotku, moduly hlavice pro ovládání ventilů otopných těles a modul pro ovládání zdroje tepla. V této kapitole bude blíže popsána architektura celého systému, funkce jednotlivých prvků a požadavky na ně kladené. Popsán bude také proces výběru komponent systémových modulů a následný návrh jejich zapojení.

### 5.1 Řídicí systém

Řídicí systém bude na nejvyšší úrovni složen z řídicí aplikace, perzistentního úložiště dat v podobě MySQL databáze a mobilní aplikace umožňující interakci s uživatelem. Schéma architektury systému je zobrazeno na obrázku 5.1. Řídicí aplikace bude zajišťovat zejména komunikaci s instancemi mobilní aplikace, obsluhu požadavků systémových modulů a také správu perzistentního úložiště dat. Součástí řídicí aplikace bude rovněž TCP server využívající unixových socketů, pomocí kterého bude realizována komunikace s instancemi mobilní aplikace. Aplikace bude také schopna obsluhy určitého typu sériového rozhraní za účelem komunikace s modulem pro bezdrátovou komunikaci, který bude vstupním bodem do bezdrátové sítě systémových modulů.

Jakožto perzistentní úložiště dat byla zvolena relační databáze MySQL. Databáze bude použita pro uložení naměřených dat, konfigurace systému a také pro uložení informací o entitách vystupujících v systému.

Monitorování a řízení systému bude realizováno prostřednictvím mobilní aplikace pro systém Android. Komunikace s řídicí aplikací bude probíhat prostřednictvím vlastního komunikačního protokolu na aplikační úrovni.



Obrázek 5.1: Architektura řídicího systému

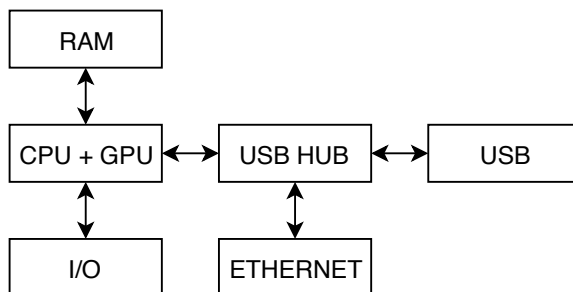
## 5.2 Centrální jednotka

Centrální jednotka bude založena na miniaturním počítači Raspberry Pi, na kterém bude spuštěna řídicí aplikace spolu s MySQL databází. Jednotka musí být schopna komunikovat s moduly hlavic instalovaných na otopných tělesech a s modulem pro ovládání zdroje tepla. Z tohoto důvodu bude počítač Raspberry Pi doplněn bezdrátovým komunikačním modulem. Proces výběru modulu je popsán v sekci 5.3.

### 5.2.1 Raspberry Pi

Raspberry Pi je miniaturní jednodeskový počítač vyvinutý britskou nadací Raspberry Pi Foundation. Výkon počítače je srovnatelný s výkonem dnešních chytrých telefonů střední třídy. Počítač Raspberry Pi je dostupný ve více verzích lišících se hlavně výkonem, rozměry a počtem komunikačních rozhraní. Největší předností této platformy je nízká cena. Dále také velký počet programovatelných vstupů/výstupů tzv. GPIO pro připojení nejrůznějších periférií a malé fyzické rozměry. Původním záměrem, stojícím za vývojem této platformy, byla podpora výuky informatiky v oblasti vestavěných systémů.

Oproti klasickým stolním počítačům je architektura počítače značně zjednodušena (obrázek 5.2) a obsahuje pouze nezbytné prvky. Všechny modely používají techniku SoC (System on Chip), a tedy komponenty jako procesor, grafické jádro či paměť jsou integrovány do jednoho fyzického čipu. U výkonnějších modelů je grafické jádro umístěno na samostatném čipu. Počítač nedisponuje řadičem SATA pro připojení disku, jako hlavní úložiště dat slouží microSD karta. Disk je možné připojit přes USB, ale přenosová rychlost je limitována verzí USB 2.0. Podobně je tomu také u rozhraní Ethernet, které je rovněž připojeno přes kořenový USB rozbočovač [22].



Obrázek 5.2: Architektura Raspberry Pi

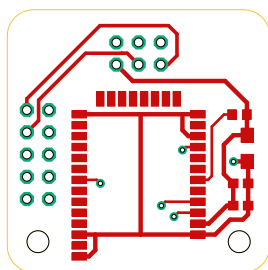
Pro Raspberry Pi je dostupných mnoho distribucí operačních systémů na bázi systému Linux. Jelikož je počítač založen na architektuře ARM, musí být operační systémy a také aplikace portovány na tuto platformu. Nejpoužívanějším operačním systémem je Raspbian, což je přímo pro Raspberry Pi upravený systém Debian.

Pro účely této práce byl vybrán model B verze 3, disponující procesorem rodiny ARM Cortex-A53 se čtyřmi jádry taktovanými na frekvenci 1,2 GHz. Operační paměť má kapacitu 1 GB, je ovšem sdílena s grafickým jádrem. Dále je dostupná čtveřice USB portů, Ethernet, Bluetooth a Wi-Fi. Mezi další dostupná komunikační rozhraní patří UART, I2C, SPI. Rovněž je možné využít až 17 GPIO vstupů/výstupů [15].

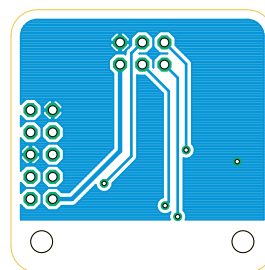
### 5.2.2 Schéma zapojení a deska plošných spojů

Účelem modulu centrální jednotky je propojení ZigBee modulu s deskou počítače Raspberry Pi, zapojení je tedy velmi jednoduché a obsahuje pouze několik podpůrných součástek. ZigBee modul je opatřen jedním oddělovacím a dvěma filtračními kondenzátory. Dále v zapojení figuruje konektorová lišta sloužící ke konfiguraci ZigBee modulu a dutinková lišta pro propojení s konektorovou lištou počítače Raspberry Pi. Toto spojení zahrnuje komunikační rozhraní UART a napájení ZigBee modulu. Schéma zapojení lze nalézt v příloze B.

Deska plošných spojů o rozměrech 30 mm × 30 mm obsahuje spoje z obou stran. Uprostřed desky je umístěn ZigBee modul, kolem kterého jsou rozmístěny podpůrné součástky a pájecí plošky pro osazení konfigurační konektorové lišty. Dutinková lišta pro spojení počítačem Raspberry Pi je určena k osazení ze spodní strany desky. Na desce jsou umístěny dva montážní otvory, které umožňují spojení s deskou počítače Raspberry Pi pomocí distančních sloupků. Předloha desky plošných spojů je zachycena na obrázku 5.3.



(a) Vrchní strana



(b) Spodní strana

Obrázek 5.3: Předloha desky plošných spojů modulu centrální jednotky

## 5.3 Hlavice otopného tělesa

Hlavice otopného tělesa musí být na základě příkazů přicházejících od centrální jednotky schopna ovládat ventil otopného tělesa, a tím regulovat průtok média tímto tělesem. Hlavice by dále měla disponovat senzorem pro měření okolní teploty. Senzor by měl sloužit jako zdroj referenční teploty pro řízení vytápění. Důležité jsou také ovládací a zobrazovací prvky sloužící k lokálnímu nastavení a zobrazení parametrů regulace vytápění – např. okolní teplota či cílová teplota vytápění.

Jelikož bude hlavice zcela bezdrátová, a tedy napájená určitým typem akumulátoru, musí být při výběru komponent kladen důraz na jejich nízkou spotřebu elektrické energie. Dalšími aspekty ovlivňující výběr komponent je cena výsledného řešení a dostupnost na trhu.

### 5.3.1 Bezdrátové rozhraní

Pro bezdrátovou komunikaci s centrální jednotkou se jeví jako nejvhodnější použití hotového modulu, který je možné s mikrokontrolérem (dále jen MCU) propojit pomocí sériového rozhraní. Nebude tak nutné řešit problematiku vlastní radiové komunikace, která leží mimo zaměření této práce. Požadavky na modul bezdrátového rozhraní jsou následující: nízká

spotřeba elektrické energie, komunikace přes sériové rozhraní, nízká cena a dostatečný dosah radiového signálu. Na základě studie zpracované v kapitole 3 byly vybrány moduly zastupující jednotlivé bezdrátové technologie, které vyhovují stanoveným požadavkům.

- Digi XBee3 Zigbee 3.0<sup>1</sup>
  - podpora protokolů ZigBee, BLE a DigiMesh
  - pásmo 2,4 GHz
  - dosah v zastavěném prostředí cca 60 m
  - komunikace skrze UART, SPI, I2C
  - pracovní napětí 2,1 V až 3,6 V
  - spotřeba 40 mA (vysílání 8 dBm); 15 mA (příjem); 1,7  $\mu$ A (*power-down*)
  - cena 14,80 € (28. prosince 2018)
- Silicon Labs AMW136<sup>2</sup>
  - protokol Wi-Fi 802.11b/g/n
  - pásmo 2,4 GHz
  - rozhraní UART, SPI
  - pracovní napětí 3,3 V
  - spotřeba 11,4 mA (vysílání 1 Mb/s); 5,7 mA (příjem); 2,8  $\mu$ A (*standby*)
  - cena 16,39 € (28. prosince 2018)
- STMicroelectronics SPBTLE-1S<sup>3</sup>
  - protokol Bluetooth Low Energy
  - pásmo 2,4 GHz
  - rozhraní UART, I2C, SPI
  - pracovní napětí 1,7 V až 3,6 V
  - spotřeba 15 mA (vysílání 5 dBm); 7,7 mA (příjem); 500 nA (*standby*)
  - cena 9,72 € (28. prosince 2018)

Z výše uvedených modulů byl vybrán modul Digi XBee3 využívající protokol ZigBee. Tento modul nabízí dostatečný dosah radiového signálu. Protokol ZigBee umožňuje také komunikaci mezi zařízeními, která nejsou ve fyzickém dosahu radiového signálu – využitím ostatních uzlů sítě jakožto opakovačů, které přeposílají danou komunikaci. Díky této funkcionalitě lze pomocí ZigBee vytvářet rozsáhlé sítě. Oproti modulům využívajícím technologii Wi-Fi nebude velikost sítě omezena použitým přístupovým bodem, a také nebude zatěžována stávající WLAN síť. V porovnání s modulem využívajícím Bluetooth má zvolený modul nepatrně větší spotřebu elektrické energie a vyšší pořizovací cenu.

---

<sup>1</sup><https://www.digi.com/products/embedded-systems/rf-modules/2-4-ghz-modules/xbee3-zigbee-3>

<sup>2</sup><https://www.silabs.com/documents/login/data-sheets/ADS-MWx36-ZentriOS-101R.pdf>

<sup>3</sup><https://www.st.com/resource/en/datasheet/spbtle-1s.pdf>

### 5.3.2 Ovládání ventilu otopného tělesa

Pro ovládání ventilu otopného tělesa se jeví jako nejvhodnější určitý typ lineárního servomotoru. Běžně dostupné lineární servomotory ovšem mají příliš velké rozměry, vysokou spotřebu elektrické energie a také nepříznivou pořizovací cenu. Možnou alternativou jsou modelářské servomotory, které jsou cenově dostupné, mají nižší spotřebu elektrické energie a menší rozměry. Jediným problémem je skutečnost, že tyto servomotory umožňují otáčivý pohyb pouze v rozsahu  $360^\circ$ , a pro přímé ovládání ventilu otopného tělesa tedy nejsou vhodné. Tento problém lze ovšem snadno vyřešit odstraněním jisté vnitřní části servomotoru, která omezuje otáčení hřídele servomotoru. Takto upravený servomotor již lze používat jako plnohodnotný motor s převodovkou pro ovládání ventilu otopného tělesa.

Pro účely této práce byl zvolen servomotor Tower Pro SG92R, který pracuje již od 3 V a disponuje kroutícím momentem až 2,5 Kg/cm. Aby bylo možné servomotor ovládat pomocí výstupů MCU, je třeba tyto výstupy posílit tzv. H-můstkem. H-můstek umožňuje servomotor nejen aktivovat či deaktivovat, ale také řídit směr jeho otáčení. Činnost H-můstků závisí na kombinaci logických úrovní obou jeho vstupů.

Pro bezpečné ovládání otopného tělesa je také vhodné určitým způsobem monitorovat polohu jeho ventilu. Existuje mnoho přístupů, jakými lze toto monitorování provádět. V tomto případě byla zvolena metoda využívající tahový potenciometr. Hlavní výhodou tohoto řešení je jednoduchost, rychlost a absolutní měření polohy ventilu.

#### Ovládací mechanismus

Základem mechanismu pro ovládání ventilu otopného tělesa je modelářský servomotor Tower Pro SG92R. Servomotor v tomto případě zastupuje elektromotor s převodovkou. Ačkoli je již motor převodovkou vybaven, bude doplněn dodatečným převodovým stupněm, za účelem zvýšení kroutícího momentu. Dodatečný převodový stupeň bude zajišťovat výkonovou rezervu ovládacího mechanismu. Ventil otopného tělesa bude ovládán závitovou tyčí, která bude uváděna do pohybu výstupem převodovky motoru. Mechanismus je zobrazen na obrázku 5.4. Snímky kompletního modelu hlavice otopného tělesa jsou součástí přílohy D.



Obrázek 5.4: Mechanismus ovládání ventilu otopného tělesa

### 5.3.3 Senzor teploty

Senzor okolní teploty bude sloužit k měření referenční teploty uvnitř místnosti, která bude využívána při regulaci vytápění. Při výběru senzoru jsou nejdůležitějšími parametry rozsah měřitelných teplot, přesnost senzoru a pořizovací cena. Senzory teploty lze obecně rozdělit

na analogové a digitální. Sensory s analogovým výstupem v závislosti na okolní teplotě mění napětí na svém výstupu. Takovéto senzory vyžadují přítomnost AD převodníku s dostatečným rozlišením, který spojitý analogový signál převede na diskrétní digitální signál pro další zpracování na MCU. Digitální senzory teploty disponují vlastním AD převodníkem a komunikačním rozhraním (např. 1-Wire, SPI či I2C) pro komunikaci s MCU. Více digitálních sensorů může být připojeno ke sdílené sběrnici – navýšením počtu sensorů se nemění nároky na počet vstupů/výstupů MCU. Naopak každý analogový sensor vyžaduje vlastní kanál AD převodníku MCU.

Na základě uvedených požadavků byly vybrány tři senzory, analogové i digitální, které jsou vhodné pro měření okolní teploty hlavice.

- Heraeus PT1000<sup>4</sup>
  - rozsah  $-50\text{ }^{\circ}\text{C}$  až  $150\text{ }^{\circ}\text{C}$
  - přesnost  $\pm 0,3\text{ }^{\circ}\text{C}$  ( $0\text{ }^{\circ}\text{C}$ ) až  $\pm 0,8\text{ }^{\circ}\text{C}$  ( $100\text{ }^{\circ}\text{C}$ )
  - lineární závislost vnitřního odporu na teplotě
  - odebíraný proud cca  $0,2\text{ mA}$
  - cena  $3,12\text{ €}$  (31. 12. 2018)
- Maxim Integrated DS18S20<sup>5</sup>
  - rozsah  $-55\text{ }^{\circ}\text{C}$  až  $125\text{ }^{\circ}\text{C}$
  - přesnost  $\pm 0,5\text{ }^{\circ}\text{C}$
  - rozhraní 1-Wire, 9b AD převodník
  - odebíraný proud cca  $1,5\text{ mA}$
  - cena  $2,74\text{ €}$  (31. 12. 2018)
- Texas Instruments TMP235<sup>6</sup>
  - rozsah  $-40\text{ }^{\circ}\text{C}$  až  $150\text{ }^{\circ}\text{C}$
  - přesnost  $\pm 0,5\text{ }^{\circ}\text{C}$
  - lineární závislost výstupního napětí na teplotě
  - odebíraný proud cca  $9\text{ }\mu\text{A}$
  - cena  $0,78\text{ €}$  (31. 12. 2018)

Všechny výše uvedené senzory nabízí dostatečnou přesnost měření a rozsah měřitelných hodnot pro použití v oblasti regulace vytápění, tedy přibližně  $0,5\text{ }^{\circ}\text{C}$ . Teoretické rozlišení měření je dáno použitým AD převodníkem. Senzor DS18S20 využívá 9b převodník, rozlišení měření je tedy přibližně  $0,352\text{ }^{\circ}\text{C}$ . U zbylých sensorů se rozlišení odvíjí od použitého MCU. Z hlediska parametrů měření jsou tedy senzory srovnatelné.

Pokud se zaměříme na pořizovací cenu a na spotřebu elektrické energie, pak z výběru vyčnívá senzor TMP235. Jelikož jsou tyto parametry kritické, byl vybrán právě senzor TMP235.

<sup>4</sup>[https://cz.mouser.com/datasheet/2/619/SMD1206\\_HSTUSA-1131806.pdf](https://cz.mouser.com/datasheet/2/619/SMD1206_HSTUSA-1131806.pdf)

<sup>5</sup><https://datasheets.maximintegrated.com/en/ds/DS18S20.pdf>

<sup>6</sup><http://www.ti.com/lit/ds/symlink/tmp235.pdf>

### 5.3.4 Ovládací prvky

Lokální nastavení cílové teploty bude možné, mimo mobilní aplikaci, provádět také prostřednictvím ovládacích prvků umístěných na regulační hlavici. Ovládání by mělo být co nejintuitivnější a také by nemělo okupovat příliš mnoho vstupů/výstupů MCU. Nejjednodušší možností je ovládání pomocí spínačů/tlačítek. Toto řešení ovšem vyžaduje vlastní vstup MCU pro každý ze spínačů a například nastavení teploty pouze prostřednictvím tlačítek může být v některých případech příliš zdlouhavé. Z tohoto důvodu bylo zvoleno řešení využívající rotační enkodér ve spojení s jedním tlačítkem. Pomocí enkodéru může být realizováno nastavení cílové teploty či průchod jednoduchým uživatelským rozhraním. Tlačítko může být využíváno k potvrzení volby. Toto řešení vyžaduje celkem tři vstupy/výstupy MCU.

### Zobrazovací prvky

Lokální nastavení, aktuální okolní teplota a další údaje by měly být zobrazovány na určitém typu displeje. Při výběru displeje byla nejdůležitějším parametrem spotřeba elektrické energie. Tento požadavek omezuje možnosti na monochromatické LCD displeje bez podsvícení a tzv. *e-paper* displeje, případně kombinace těchto dvou technologií. Při výběru byla zohledněna také dostupnost a cena displeje. Níže je uveden výběr displejů vhodných pro účely této práce.

- Nokia 5110/3310 LCD<sup>7</sup>
  - úhlopříčka 1,5" (84 px × 48 px), monochromatický
  - řadič PCD8544, komunikace pomocí SPI
  - odebíraný proud 150 μA při 3,3 V
  - cena 8,73 € (30. 12. 2018)
- Adafruit 1.54" Tri-Color eInk Display<sup>8</sup>
  - úhlopříčka 1,54" (152 px × 152 px), tři možné barvy pixelů
  - řízení skrze SPI, vlastní SRAM buffer
  - spotřeba energie pouze při překreslování
  - cena 19,65 € (30. 12. 2018)
- Adafruit SHARP Memory Display<sup>9</sup>
  - úhlopříčka 1,3" (168 px × 144 px), monochromatický
  - řízení prostřednictvím SPI
  - vyžaduje periodické překreslování
  - typicky odebíraný proud 4 μA při 3,3 V
  - cena 21,79 € (30. 12. 2018)

---

<sup>7</sup><https://www.adafruit.com/product/338>

<sup>8</sup><https://www.adafruit.com/product/3625>

<sup>9</sup><https://www.adafruit.com/product/3502>

Z uvedených displejů se jeví jako nejvhodnější Adafruit SHARP Memory Display, který kombinuje nízkou spotřebu elektrické energie *e-paper* displejů a rychlost překreslování klasických LCD displejů. Ovšem zásadní nevýhodou je absence vlastního SRAM bufferu pro dočasné ukládání stavu pixelů k vykreslení. Z tohoto důvodu musí být cca 3 kB dynamické paměti MCU vyhrazeno pouze pro řízení displeje. Další nevýhodou je také vyšší cena. Z důvodu vyšší ceny a nižší rychlosti překreslování byl před tímto displejem upřednostněn klasický LCD displej.

Zvolený LCD displej má oproti ostatním displejům vyšší spotřebu elektrické energie, ale jeho pořizovací cena je řádově nižší. LCD displej bude aktivní pouze během úpravy lokálního nastavení uživatelem, proto by zvolený displej neměl mít příliš veliký dopad na životnost akumulátoru.

### 5.3.5 MCU

Na základě zvolených komponent lze odvodit požadavky na počet a typ vstupů/výstupů a také na komunikační rozhraní, kterými by měl MCU disponovat. Tyto požadavky jsou shrnuty níže:

- 15 digitálních vstupů/výstupů;
- 3 analogové vstupy do AD převodníku;
- rozhraní UART, SPI či I2C pro komunikaci se ZigBee modulem;
- rozhraní SPI pro komunikaci s LCD displejem.

Jelikož bude hlavice napájena akumulátorem, je důležitým kritériem spotřeba elektrické energie. MCU by tedy měl disponovat režimy úspory elektrické energie. Výhodou by byl také integrovaný čítač reálného času (dále pouze RTC), aby byla hlavice schopna v přesných časových intervalech odesílat data na centrální jednotku. Za účelem úspory elektrické energie bude MCU a také další komponenty většinu času v režimu úspory elektrické energie či úplně bez napájení. MCU tedy musí být schopen zpracovat externí přerušování generované na základě ovládacích prvků, a přejít tak do standardního režimu běhu.

Důležitým aspektem je také dostupnost programovacích a ladících nástrojů, míra zdokumentování a typ pouzdra MCU, jež určuje obtížnost ručního osazení na desku plošných spojů. Na základě uvedených požadavků bylo vybráno několik MCU, které jsou vhodné pro účely této práce.

- STMicroelectronics STM32L031K6<sup>10</sup>
  - architektura ARM Cortex-M0+, frekvence až 32 MHz
  - 32 kB FLASH, 8 kB RAM
  - komunikační rozhraní USART, LPUART, SPI, I2C
  - AD převodník s rozlišením 12 b a 10 kanály
  - pracovní napětí 1,65 V až 3,6 V
  - odebíraný proud při napájecím napětí 3 V od 76  $\mu$ A/MHz v režimu *run*, 570 nA v režimu *standby* s aktivním RTC
  - pouzdro LQFP32 či UFQFPN32

<sup>10</sup><https://www.st.com/resource/en/datasheet/stm32l031g4.pdf>



- cena 2,20 € (29. 12. 2018)
- Texas Instruments MSP430FR2355<sup>11</sup>
  - 16b architektura RISC, frekvence až 24 MHz
  - 32 kB FRAM, 4 kB RAM
  - komunikační rozhraní UART, SPI, I2C
  - AD převodník s rozlišením 12 b a 12 kanály
  - pracovní napětí 1,8 V až 3,6 V
  - odebíraný proud při napájecím napětí 3 V od 142 μA/MHz v režimu *active*, 620 nA v režimu *standby*
  - pouzdro LQFP48, TSSOP38 či VQFN40
  - cena 5,29 € (29. 12. 2018)
- NXP Semiconductors LPC1224FBD48<sup>12</sup>
  - architektura ARM Cortex-M0, frekvence až 45 MHz
  - 32 kB FLASH, 4 kB RAM
  - komunikační rozhraní UART, SPI, I2C
  - 10b AD převodník
  - pracovní napětí 3,2 V až 3,6 V
  - odebíraný proud při napájecím napětí 3 V od 420 μA/MHz v režimu *active*, 720 nA v režimu *deep power-down*
  - pouzdro LQFP48
  - cena 3,43 € (29. 12. 2018)

Výběr obsahuje zástupce několika společností, kdy všechny vybrané MCU spadají do stejné cenové kategorie a rovněž disponují podobnou sadou periférií. Je patrné, že zástupce společnosti STMicroelectronics má oproti ostatním MCU dvojnásobnou kapacitu volatilní paměti RAM, řádově nižší spotřebu elektrické energie a také nejpříznivější cenu. Disponuje 12b AD převodníkem, čímž je zajištěno dostatečné rozlišení při měření okolní teploty, napětí akumulátoru a polohy ventilu otopného tělesa. MCU se vyrábí v pouzdře se 32 vývody, poskytuje tak optimální počet vstupů/výstupů pro připojení zvolených komponent.

### 5.3.6 Napájení

Komponenty regulační hlavice budou napájeny akumulátorem. Před výběrem konkrétních prvků systému napájení je nutné stanovit pracovní napětí celého obvodu a maximální odebíraný proud všemi komponentami. Při výběru vlastního akumulátoru je nutné kromě jmenovitého napětí a proudu zohlednit také jeho kapacitu a rozměry (poměr kapacity ku fyzickému objemu).

Průnikem rozsahů pracovního napětí všech vybraných komponent je 3,3 V. Dále je nutné stanovit maximální velikost odebíraného proudu celým obvodem, který je roven součtu odebíraných proudů všemi komponentami, ten má velikost 0,256 A. Bude tedy nutné použít

<sup>11</sup><http://www.ti.com/lit/ds/symlink/msp430fr2355.pdf>

<sup>12</sup><https://www.nxp.com/docs/en/data-sheet/LPC122X.pdf>

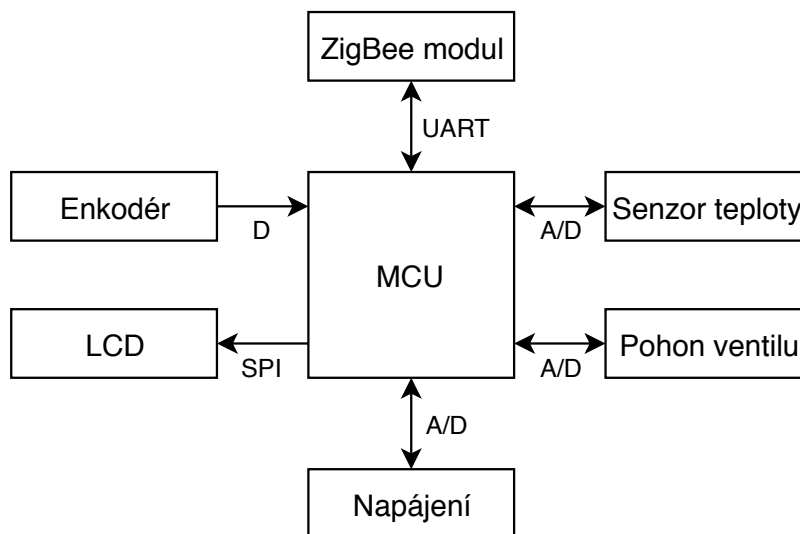
stabilizovaný zdroj napětí o velikosti 3,3 V, jenž je schopen trvale poskytovat proud alespoň 0,256 A.

Jako nejvhodnější se jeví použití Li-Ion nebo Li-Pol článků, jelikož poskytují výborný poměr kapacity ku fyzickým rozměrům článku a jsou snadno dostupné v mnoha provedeních. Tyto články mají nominální napětí 3,7 V, ovšem plně nabitě dosahují napětí 4,2 V a je možné je vybit až na napětí cca 2,6 V bez poškození článku. Pro zajištění stabilizovaného napětí bude nutné použít DC-DC měnič, který proměnlivé vstupní napětí transformuje na konstantní výstupní napětí. Pro tento účel byl vybrán měnič TPS63031, který pracuje se vstupním napětím v rozsahu 1,8 V až 5,5 V. Vstupní napětí převádí s efektivitou až 98 % na výstup s konstantním napětím o velikosti 3,3 V. Jestliže je vstupní napětí vyšší než 2,4 V, měnič je schopen trvale dodávat proud až 500 mA. Měnič je rovněž schopen automaticky přejít do úsporného režimu.

Za účelem úspory energie budou rozvody napájení rozděleny do několika nezávislých domén tak, aby bylo možné aktivovat potřebné komponenty a deaktivovat ty nepotřebné. Touto technikou bude eliminován statický příkon komponent. Připojení domén k napájení bude realizováno programově ze strany MCU.

### 5.3.7 Blokové schéma

Na obrázku 5.5 je zachyceno blokové schéma hlavičky otopného tělesa. U hran spojujících jednotlivé komponenty je uveden typ rozhraní či signálu (D značí digitální signál, A/D značí kombinaci analogových a digitálních signálů).



Obrázek 5.5: Blokové schéma hlavičky

### 5.3.8 Schéma zapojení

Schéma zapojení lze rozdělit do několika logických bloků, které jsou popsány níže. Kompletní schéma zapojení je zobrazeno v příloze B.

## MCU

Zapojení MCU STM32L031K6 vychází z vývojové desky NUCLEO-L031K6<sup>13</sup>. MCU je vybaven externím oscilátorem pro čítač reálného času, jež je doplněn dvojicí 4,3 pF kondenzátorů pro jeho snazší rozběh. Referenční napětí AD převodníku je odděleno od napájecího feritovou perlou za účelem eliminace případného vysokofrekvenčního rušení. Zapojení obsahuje také mikropsínač pro reset MCU a propojku pro volbu zdroje programu MCU. Napájení MCU je odděleno keramickými 100 nF kondenzátory.

## Bezdrátový modul

Bezdrátový modul XBee3 ZigBee je doplněn filtračními 1  $\mu$ F a 8,2 pF kondenzátory pro eliminaci případného vysokofrekvenčního rušení ze zdroje napájení. V zapojení se vyskytuje také oddělovací 10  $\mu$ F kondenzátor. Propojení modulu s MCU je realizováno pomocí propojek, aby bylo možné modul izolovat za účelem jeho konfigurace.

## Nabíjení a ochrana akumulátoru

Nabíjení akumulátoru zajišťuje obvod MCP73833, který je vhodný jak pro Li-Ion, tak i pro Li-Pol akumulátory. Obvod je schopen detekovat konec nabíjení a rovněž nabíjení opětovně spustit při vybití akumulátoru. Obvod rovněž disponuje možností nastavení nabíjecího proudu a tepelnou ochranou akumulátoru.

Obvod je zapojen dle katalogového listu a je doplněn NTC termistorem o hodnotě 10 k $\Omega$  pro kontrolu teploty akumulátoru.

Obvod BQ29700D zajišťuje kompletní ochranu akumulátoru, která zahrnuje ochranu proti přebití, hlubokému vybití, zkratu na výstupu a také proudovou ochranu. Obvod musí být doplněn dvojicí tranzistorů, které ovládají výkonovou část zapojení. Obvod pro své fungování vyžaduje, kromě dvojice tranzistorů, minimum externích součástek v podobě dvou rezistorů a oddělovacího kondenzátoru.

## DC-DC měnič

Zapojení měniče vychází z katalogového zapojení<sup>14</sup>. V tomto zapojení byly upraveny hodnoty a typ oddělovacích kondenzátorů na vstupu a výstupu měniče. Použity jsou tantalové 10  $\mu$ F kondenzátory, které jsou pro tuto aplikaci vhodné zejména z důvodu malých rozměrů.

## H-můstek

Pro ovládání servomotoru je použit H-můstek v podobě integrovaného obvodu BD6211F. Předností tohoto obvodu je téměř nulová spotřeba elektrické energie v klidovém stavu. Obvod je k MCU připojen prostřednictvím dvou digitálních výstupů. Proud na těchto výstupech je omezen rezistory o hodnotě 1 k $\Omega$ . Snadný rozběh servomotoru je zajištěn k němu paralelně připojeným 100 nF kondenzátorem.

## Napájecí domény

Selektivní připojování určitých částí obvodu k napájení je realizováno pomocí p-kanál mosfetů. Mosfety jsou doplněny o pull-up rezistor a také o rezistor omezující proud, kterým

<sup>13</sup><https://www.st.com/en/evaluation-tools/nucleo-l031k6.html>

<sup>14</sup><http://www.ti.com/lit/ds/symlink/tps63030.pdf>

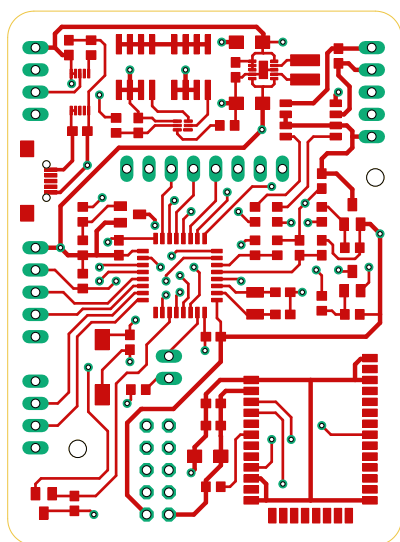
jsou řízeny. Tímto způsobem může být připojen či odpojen od napájení obvod ovládní servomotoru, LCD displej, teplotní čidlo a napěťový dělič pro měření napětí akumulátoru.

### 5.3.9 Deska plošných spojů

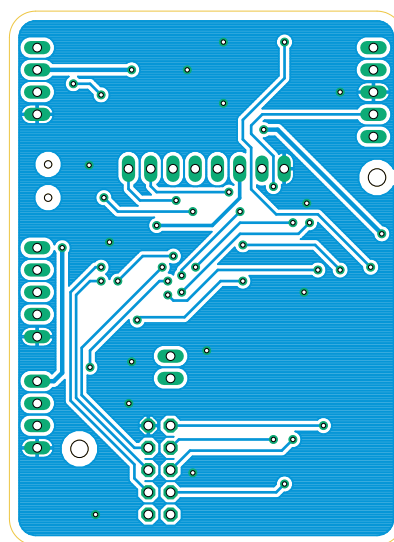
Deska plošných spojů má rozměry 61 mm × 45 mm a je oboustranná. Spodní strana desky je vyplněna zemnicím polygonem. V horní části desky jsou umístěny komponenty spojené s napájením modulu a řízení servomotoru, tedy obvody pro nabíjení a ochranu akumulátoru, DC-DC měnič, H-můstek a podpůrné součástky. Vlevo od tohoto bloku jsou umístěny pájecí plošky pro připojení akumulátoru a ochranného termistoru. Vpravo se pak nachází plošky pro připojení servomotoru a jeho zpětné vazby v podobě potenciometru. Pod napájecím blokem je umístěna vývodová lišta pro připojení LCD displeje. Vlevo od lišty je prostor pro micro-USB konektor sloužící k nabíjení akumulátoru.

Uprostřed desky je umístěn MCU, kolem kterého jsou rozmístěny podpůrné součástky. Ve střední části desky jsou rovněž umístěny mosfety napájecích domén, mikropínač pro reset MCU a také propojka pro volbu zdroje programu MCU. Na levém okraji desky je umístěna sada pájecích plošek pro připojení rotačního enkodéru.

Ve spodní části desky se nachází teplotní čidlo a ZigBee modul s podpůrnými součástkami. Nalevo se nachází pájecí plošky pro nahrávání a ladění programu MCU. Předlohy obou stran desky plošných spojů jsou na obrázku 5.6.



(a) Vrchní strana



(b) Spodní strana

Obrázek 5.6: Předloha desky plošných spojů hlavice otopného tělesa

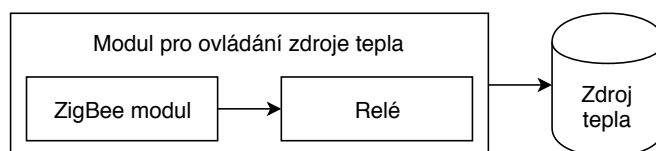
## 5.4 Modul pro ovládní zdroje tepla

Modul pro ovládní zdroje tepla musí být schopen přijímat příkazy od centrální jednotky a na základě nich zdroj tepla zapínat či vypínat. Předpokládá se, že zdroj tepla standardně

využívá napájení ze sítě, a tedy modul pro jeho ovládání bude rovněž napájen ze sítě. Nebude tedy nutné provádět jakoukoli údržbu tohoto modulu. Kotle centrálního vytápění typicky disponují speciálními vývody určenými k externímu ovládání jejich činnosti. Tyto vývody slouží pouze ke spínání zařízení a nejsou jimi přenášeny vysoké proudy.

Hlavní komponentou modulu pro ovládání zdroje tepla bude totožný ZigBee modul, jaký byl vybrán pro hlavice otopných těles i centrální jednotku. ZigBee modul disponuje GPIO vývody, které mohou být libovolně konfigurovány. Není tedy nutné, aby modul obsahoval separátní MCU, jak je tomu u hlavice otopných těles. Pomocí vstupů/výstupů ZigBee modulu bude řízeno relé, které bude zajišťovat samotné ovládání zdroje tepla. Relé bylo vybráno z důvodu univerzality – umožňuje spínat jak střídavý, tak i stejnosměrný proud a rovněž má široký rozsah spínaného napětí.

Modul bude napájen ze sítě, což umožňuje použití síťového adaptéru se stabilizovaným výstupním napětím. Výběr adaptéru závisí na pracovním napětí a potřebném proudu použitých komponent. Pracovní napětí ZigBee modulu je 2,1 V až 3,6 V [5] a běžně dostupná relé mají pracovní napětí od 3 V. Použit tedy bude adaptér s výstupním napětím 3,3 V bez dodatečné napájecí elektroniky. Blokové schéma modulu pro ovládání zdroje tepla je zachyceno na obrázku 5.7.



Obrázek 5.7: Blokové schéma modulu pro ovládání zdroje tepla

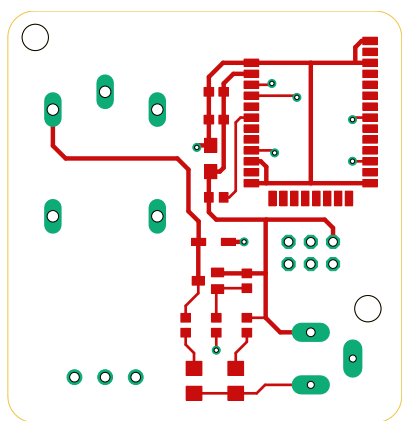
#### 5.4.1 Schéma zapojení

Hlavní komponentou modulu pro ovládání zdroje tepla je ZigBee modul. Modul je doplněn oddělovacím 10  $\mu$ F kondenzátorem a dvojicí filtračních kondenzátorů o kapacitě 1  $\mu$ F a 8,2 pF. Výstupem, určeným ke spínání zdroje tepla, je přes 1 k $\Omega$  rezistor buzen p-mosfet, který ve vybuzeném stavu přivádí napájení na relé. Paralelně k relé je zapojena dioda, která zajišťuje vyfiltrování případných napěťových špiček vznikajících na cívce relé, a chrání tak budící mosfet. Zapojení obsahuje také dvojici LED diod signalizujících přítomnost napájení a sepnutí relé. Kompletní schéma zapojení je umístěno v příloze B.

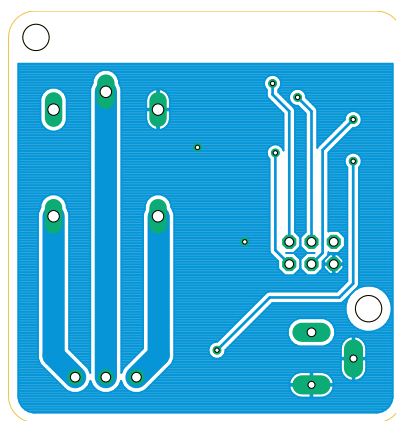
#### 5.4.2 Deska plošných spojů

Deska plošných spojů je oboustranná, o rozměrech 45 mm  $\times$  47 mm. V levé části desky je umístěno relé společně se šroubovací svorkovnicí, na kterou jsou vyvedeny spínané kontakty relé. Tato svorkovnice slouží k propojení modulu se zdrojem tepla. V pravé části desky je umístěn ZigBee modul a konektor pro připojení externího napájecího adaptéru. Rovněž se zde nachází pájecí otvory pro instalaci konektorové lišty použitelné ke konfiguraci ZigBee modulu. Uprostřed desky jsou umístěny podpůrné součástky a signalizační LED diody.

Spodní strana desky je vyplněna zemnicím polygonem. Část polygonu v horní části desky je, dle doporučení výrobce, odstraněna za účelem eliminace negativních vlivů na anténu ZigBee modulu. Předloha desky plošných spojů je zachycena na obrázku 5.8.



(a) Vrchní strana



(b) Spodní strana

Obrázek 5.8: Předloha desky plošných spojů modulu pro ovládání zdroje tepla

## Kapitola 6

# Návrh a implementace řídicí aplikace

Tato kapitola se zabývá návrhem architektury a implementací řídicí aplikace. Bude popsána struktura implementovaného programu a princip fungování jeho částí. Stěžejní funkce aplikace jsou následující:

- komunikace s hlavicemi otopných těles a modulem pro ovládání zdroje tepla,
- komunikace s mobilní aplikací,
- vyhodnocování získaných dat a úprava parametrů systému,
- správa perzistentního úložiště dat.

Výběr implementačního jazyka byl značně ovlivněn výše uvedenými požadavky. Pro daný jazyk by mělo být dostupné aplikační rozhraní zajišťující obsluhu lokální MySQL databáze. Rovněž by mělo být dostupné aplikační rozhraní pro obsluhu UART rozhraní či přímo pro obsluhu vybraného ZigBee modulu. Důležitým faktorem při výběru implementačního jazyka byla také cílová platforma.

Pro implementaci řídicí aplikace byl použit jazyk Python. Výhodou tohoto jazyka je dostupnost velkého množství rozšiřujících modulů a podpora více programovacích paradigmat. Kladem je také skutečnost, že je Python součástí základní instalace většiny distribucí operačního systému GNU/Linux, včetně distribuce Raspbian. Jedná se o interpretovaný, dynamicky typovaný jazyk, z čehož mohou plynout i potenciální nevýhody v podobě nižšího výkonu či nestandardního chování programu. Jazyk Python byl zvolen také z důvodu volně dostupných modulů pro obsluhu MySQL databáze a použitého ZigBee modulu.

Řídicí aplikaci lze rozdělit do tří logických bloků: hlavní proces, vlákno pro komunikaci se ZigBee moduly a vlákno pro komunikaci s instancemi mobilní aplikace.

### 6.1 Hlavní proces

Účelem hlavního procesu je inicializace systému a následné vytvoření a spuštění komunikačních vláken. V rámci hlavního procesu jsou rovněž vykonávány rutinní a periodické činnosti. Mezi tyto činnosti patří kontrola aktivity systémových modulů za účelem zjištění, zda korektně pracují. V negativním případě jsou nastaveny odpovídající hodnoty v databázi a tato informace je sdělena uživateli prostřednictvím mobilní aplikace.

### 6.1.1 Úložiště dat

Součástí řídicí aplikace je také perzistentní úložiště dat v podobě relační MySQL databáze sloužící k uchování stavu systému. ER diagram popisující schéma vytvořené databáze je součástí přílohy E. Základ diagramu je tvořen entitními množinami reprezentujícími abstrakce používané v rámci systému, tedy místnosti, zařízení a uživatelé. Dále byla definována entitní množina `access_rights`, která reprezentuje množinu přístupových práv uživatelů k jednotlivým místnostem. Díky této množině je možné definovat přístupová práva pro každého uživatele a místnost separátně, čímž je zajištěna flexibilita systému přístupových práv.

Systém umožňuje nastavení dvou typů programů vytápění. Prvním typem jsou klasické denní programy. Ty jsou v diagramu reprezentovány entitní množinou `daily_programs`. Z povahy systému je nutné, aby pro každou místnost existoval alespoň jeden denní program vytápění. Druhým typem programů jsou programy jednorázové. Tyto programy jsou reprezentovány entitní množinou `one_time_programs`. Pro každou místnost může existovat nejvýše jeden aktivní jednorázový program.

Systémové moduly jsou v diagramu reprezentovány entitní množinou `devices`, jejíž parametry reflektují stav těchto modulů. Entita modulu může být svázána nejvýše s jednou entitou reprezentující místnost. Volná vazba mezi těmito entitami je vyžadována z důvodu existence modulů, které ještě nebyly přiřazeny do žádné z místností. Se systémovými moduly souvisí také entitní množina `devices_values`, jež reprezentuje množinu záznamů obsahujících naměřenou teplotu v odpovídající místnosti a příznak reflektující polohu ventilu otopného tělesa.

Relační databáze byla využita i pro uložení konfigurace systému regulace vytápění. Z tohoto důvodu lze v ER diagramu nalézt také entitní množinu `settings`, která reprezentuje aktuální nastavení systému.

### 6.1.2 MySQL API

Pro obsluhu MySQL databáze bylo použito rozhraní `MySQLdb`, které je založeno na modulu `_mysql` a zaručuje přenositelnost napříč databázemi. Nad tímto rozhraním byla implementována třída `MySQLAPI`, která přizpůsobuje přístup k databázi specifickým potřebám aplikace. Třída rovněž zajišťuje synchronizaci přístupu na úrovni vláken pomocí binárních zámků.

V rámci konstruktoru třídy je vytvořeno spojení s lokálním MySQL serverem a také získán objekt `Cursor`, prostřednictvím kterého je možné vykonávat dotazy. Třída dále disponuje dvěma veřejnými metodami `query` a `escape`. Metoda `query` zajišťuje vykonávání dotazů. Jestliže se jedná o dotaz modifikující obsah databáze, vrací tato metoda status operace. Rovněž je zajištěno obnovení stavu databáze v případě vzniku chyby během provádění dotazu. Jedná-li se o dotaz typu `SELECT`, vrací tato metoda kromě statusu také pole požadovaných záznamů a jejich počet.

## 6.2 Vlákno pro komunikaci se ZigBee moduly

Činnost tohoto vlákna spočívá ve zpracování dotazů od systémových modulů. Zpracování dotazu zahrnuje aktualizaci obsahu databáze a odeslání odpovědi s požadovanými daty. Jako vstupní bod do ZigBee sítě slouží modul instalovaný na desku počítače Raspberry Pi. Obsluha tohoto modulu je realizována prostřednictvím knihovny `XBee`<sup>1</sup> pro jazyk Py-

<sup>1</sup><https://github.com/digidotcom/python-xbee>



thon. Tato knihovna poskytuje rozhraní pro příjem a odesílání zpráv, provádění lokálních i vzdálených AT příkazů a také pro správu ZigBee sítě.

Vlákno pro komunikaci se ZigBee moduly je reprezentováno třídou `XBeeThread`, která rozšiřuje třídu `Thread` dostupnou z modulu `threading`. Třída `XBeeThread` přetěžuje konstruktor a metodu `run` rodičovské třídy. Metoda `run` reprezentuje úsek programu, který má být vykonáván v rámci samostatného vlákna. V rámci této metody je inicializováno sériové rozhraní za účelem komunikace s lokálním ZigBee modulem. Následně je registrována metoda `__recvCallback` jakožto tzv. *callback* funkce, jejímž účelem je zpracování příchozích zpráv ze ZigBee sítě.

### 6.2.1 Komunikační protokol

Pro komunikaci mezi systémovými moduly je používán binární komunikační protokol fungující způsobem dotaz-odpověď. Protokol podporuje celkem čtyři typy dotazů a odpovědí: párování, aktualizace dat, nastavení jednorázového programu vytápění a ovládání zdroje tepla.

#### Párování

Systémové moduly musí být před použitím nejdříve spárovány s centrální jednotkou. Proces párování zahrnuje výměnu konfiguračních dat a v případě hlavice otopného tělesa také jeho přiřazení do některé z místností. Formát párovacích zpráv je zobrazen na obrázku 6.1.

Po přijetí žádosti o párování je nejdříve zjištěno, zda jsou jeho parametry již uloženy v databázi. V negativním případě jsou parametry modulu do databáze vloženy. Jedná-li se o modul pro ovládání zdroje tepla, proces párování v tomto okamžiku končí. V případě hlavice otopného tělesa následuje přiřazení do některé z místností. Přiřazení je prováděno uživatelem pomocí mobilní aplikace (viz sekce 8.3). Po úspěšném přiřazení do místnosti je modul spárován a připraven k použití.

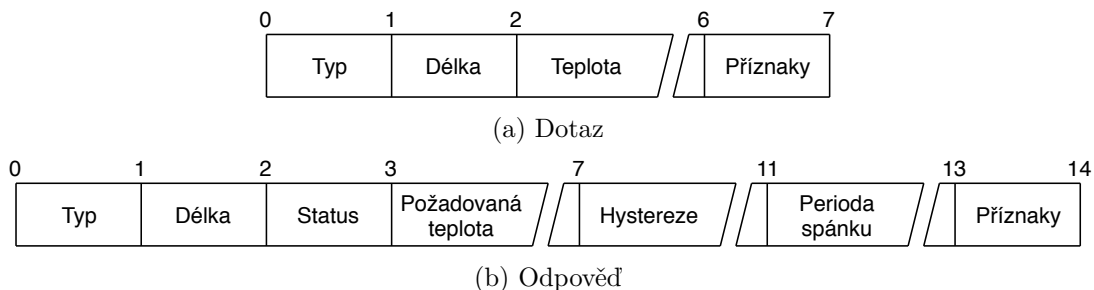


Obrázek 6.1: Formát párovacích zpráv

#### Aktualizace dat

Mezi hlavicemi otopných těles a centrální jednotkou dochází k periodickým výměnám dat. Hlavice poskytuje centrální jednotce informace o teplotě v místnosti, poloze ventilu či stavu akumulátoru. Naopak centrální jednotka sděluje hlavici aktuální nastavení systému v podobě požadované teploty, hystereze či periody spánku hlavice. Formát zpráv periodické výměny dat je zachycen na obrázku 6.2.

Zpracování žádosti o aktuální data začíná kontrolou, zda je daný modul úspěšně spárován s centrální jednotkou. Následně dojde k získání požadovaných dat z databáze a jejich odeslání nazpět. Po zpracování žádosti je na základě přijatých dat provedena aktualizace obsahu databáze. Tato činnost je provedena metodou `__updateHeatingInfo`.

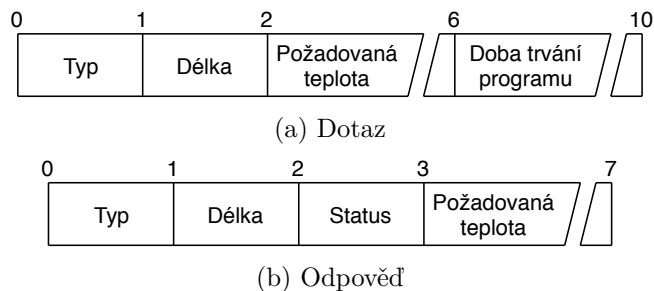


Obrázek 6.2: Formát zpráv periodické výměny dat

### Nastavení jednorázového programu vytápění

Prostřednictvím hlavičky otopného tělesa lze aktivovat či deaktivovat jednorázový program vytápění – je možné nastavit požadovanou teplotu a dobu platnosti tohoto nastavení. Takto nastavený program má poté přednost před denním programem místnosti.

Žádost o nastavení jednorázového programu je zpracována pouze tehdy, je-li daný modul spárován s centrální jednotkou. V pozitivním případě dojde k nastavení jednorázového programu prostřednictvím metody `__updateOneTimeProg`. Formát zpráv pro nastavení jednorázového programu je zobrazen na obrázku 6.3.



Obrázek 6.3: Formát zpráv pro nastavení jednorázového programu

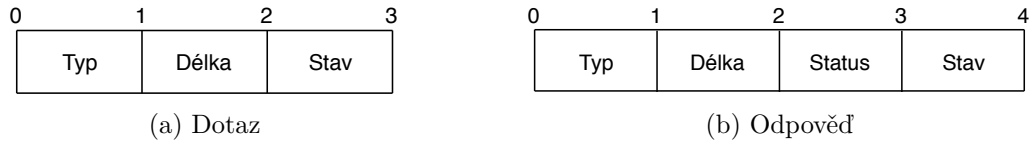
### Ovládání zdroje tepla

Pro správné fungování systému je nutné ovládat také zdroj tepla – např. plynový či elektrický kotel. Zdroj tepla by měl být aktivní ve chvíli, kdy je otevřen ventil alespoň jednoho otopného tělesa. Informace o tom, zda je ventil otopného tělesa otevřen či nikoli, je součástí zpráv vyměněných při aktualizaci dat.

Žádost spojená s ovládáním zdroje tepla může být odeslána jak modulem směrem na centrální jednotku, tak i centrální jednotkou na modul. Tento přístup byl zvolen za účelem rychlé adaptace systému na změny požadované teploty. Po přijetí žádosti je metodou `__sendResBoiler` zjištěn stav ventilů otopných těles na základě obsahu databáze. Následně je sestavena a odeslána odpověď (obrázek 6.4).

#### 6.2.2 Stanovení požadované teploty v místnosti

Pro správné fungování regulace vytápění musí být řídicí aplikace schopna stanovit aktuální požadovanou teplotu pro danou místnost. Požadovanou teplotou je řízena činnost modulů



Obrázek 6.4: Formát zpráv pro ovládání zdroje tepla

hlavic otopných těles a také je zobrazována uživateli prostřednictvím mobilní aplikace. Stanovení požadované teploty pro danou místnost závisí na více faktorech.

Nejdříve je nutné ověřit, zda je systém regulace vytápění zapnut. Je-li systém vypnut, požadovaná teplota pro všechny místnosti je 0 °C. Pokud je systém zapnut, musí být ověřeno, zda není aktivní některý ze speciálních režimů systému (viz sekce 8.3). Jestliže je aktivní speciální režim činnosti systému, požadovaná teplota se odvíjí od uživatelského nastavení režimu. Není-li aktivní žádný ze speciálních režimů, musí být provedena kontrola existence jednorázového programu vytápění. Jednorázový program může být nastaven po určitý časový interval či permanentně. Pokud není aktivní ani jednorázový program, teplota je určena dle nastavení denního programu dané místnosti. V tomto případě je vyhledán poslední platný úsek programu pro aktuální denní dobu.

### Aktivace programu s časovým předstihem

Systém regulace vytápění umožňuje také aktivaci denního programu s předstihem. Je-li tato funkce povolena, dojde k aktivaci odpovídajícího úseku programu s dostatečným časovým předstihem tak, aby v časový okamžik daný programem již teplota v místnosti odpovídala teplotě požadované.

Délka časového předstihu se odvíjí od tepelné setrvačnosti dané místnosti a příslušného otopného tělesa. Rovněž závisí na rozdílu aktuální teploty v místnosti a požadované teploty. Délka předstihu tedy musí být počítána průběžně. Před každým dotazem na aktuální požadovanou teplotu jsou nejdříve získány časové intervaly, ve kterých byl v minulosti aktivní zdroj tepla. Následně je stanovena délka těchto intervalů a rozdíl teplot v místnosti v okamžiku aktivace a deaktivace zdroje tepla. Podíl rozdílu teplot a délky intervalu udává velikost změny teploty za jednotku času pro daný interval. Podíly z jednotlivých intervalů jsou zprůměrovány. Výsledná hodnota udává průměrnou velikost změny teploty v dané místnosti za jednotku času, je-li aktivní zdroj tepla. Délka předstihu je následně stanovena jako podíl rozdílu požadované teploty a aktuální teploty v místnosti ku vypočtené průměrné velikosti změny teploty. Je-li aktuální teplota vyšší než teplota požadovaná, délka předstihu je automaticky nulová. Tímto způsobem je ošetřena předčasná deaktivace zdroje tepla. Vztah pro výpočet délky předstihu udává rovnice 6.1.

$$p = \begin{cases} \frac{T_p - T_a}{\frac{1}{N} \cdot \sum_{n=1}^N \frac{\Delta T_n}{\Delta t_n}} & \text{pro } (T_p - T_a) \geq 0 \\ 0 & \text{jinak} \end{cases} \quad (6.1)$$

kde:

$p$  – délka časového předstihu [s];

$T_p$  – požadovaná teplota [°C];

$T_a$  – aktuální teplota v místnosti [°C];

$N$  – počet časových intervalů, ve kterých probíhalo vytápění [–];

$T_n$  – velikost změny teploty v průběhu  $n$ -tého časového intervalu [°C];

$t_n$  – délka  $n$ -tého časového intervalu [ $s$ ].

## 6.3 Vlákno pro komunikaci s mobilní aplikací

Účelem tohoto vlákna je komunikace s instancemi mobilní aplikace pro monitorování a ovládání systému.

Vlákno je reprezentováno třídou `TCPThread`, jež rozšiřuje třídu `Thread` dostupnou z modulu `threading`. Třída `TCPThread` implementuje konkurentní TCP server, který obsluhuje požadavky od klientů v podobě instancí mobilní aplikace. Třída `TCPThread` přetěžuje konstruktor a metodu `run` rodičovské třídy. Konstruktor je v tomto případě použit pro předání hodnot a inicializaci instančních proměnných. Metoda `run` slouží pro vytvoření socketu, na kterém server naslouchá, a následně předání příchozích požadavků samostatným vláknům.

### 6.3.1 Komunikační protokol

Komunikace probíhá prostřednictvím textového protokolu ve formátu JSON. Formát JSON byl zvolen zejména pro nezávislost na cílové platformě, jednoduchost a čitelnost. Nevýhodou je obecně větší délka přenášených zpráv v porovnání s binárními protokoly. Nicméně objem přenášených dat je relativně malý, a rychlost komunikace tedy by neměla být negativně ovlivněna. Protokol funguje způsobem dotaz-odpověď.

Každá zpráva je tvořena hlavičkou a tělem, vnitřní struktura těchto částí se odvíjí od typu zprávy. Jedná-li se o dotaz, hlavička musí obsahovat číselný identifikátor typu zprávy, přihlašovací jméno uživatele a hash hodnotu hesla uživatele (viz sekce 8.2). Výjimkou je požadavek o registraci uživatele, který přihlašovací údaje uživatele neobsahuje. Hlavička zprávy reprezentující odpověď obsahuje pouze identifikátor typu zprávy a příznak indikující úspěšnost zpracování souvisejícího dotazu. Je-li tento příznak roven nule, došlo k chybě zpracování dotazu a případný obsah těla zprávy je neplatný. Tělo rámce nemá předepsaný pevný formát a zcela závisí na konkrétním typu zprávy. Závaznou strukturu zpráv protokolu ilustrují výpisy 6.1 a 6.2. Pro korektní fungování komunikace mezi centrální jednotkou a instancemi aplikace je podporováno celkem 46 typů zpráv odpovídajících 23 dotazům a stejnému počtu odpovědí.

```
{
  "header": {
    "type": "typ zpravy",
    "login": "login",
    "hash": "hash hodnota hesla"
  },
  "data": {...}
}
```

Výpis 6.1: Formát dotazu

```
{
  "header": {
    "type": "typ zpravy",
    "status": "status operace"
  },
  "data": {...}
}
```

Výpis 6.2: Formát odpovědi

Zpracování příchozích zpráv je prováděno metodou `__recvCallback` v samostatných vláknech a obecně probíhá následujícím způsobem:

1. identifikace typu zprávy,
2. ověření přihlašovacího jména a hash hodnoty hesla (nejedná-li se o žádost o registraci),
3. načtení požadovaných dat či aktualizace obsahu databáze na základě dat přijatých,
4. vygenerování a odeslání odpovědi.

Po zpracování zprávy je spojení s klientem ukončeno a pro každý další dotaz musí být vytvořeno nové spojení.

## Kapitola 7

# Návrh a implementace firmware systémových modulů

V této kapitole bude popsán návrh a implementace obslužného firmware hlavice otopného tělesa a modulu pro ovládání zdroje tepla. Bude vysvětlen princip činnosti firmware a také obsluha stěžejních periférií obou modulů.

### 7.1 Hlavice otopného tělesa

Hlavice otopného tělesa je řízena MCU STM32L031K6 architektury ARM Cortex-M0+. Programování a ladění MCU řady STM32 může být realizováno prostřednictvím rozhraní SWD či JTAG (viz obrázek 7.1) za pomoci příslušného převodníku. Pro vývoj firmware je dostupných hned několik integrovaných vývojových prostředí. Výhodou je také existence nástroje STM32CubeMX<sup>1</sup>, který díky intuitivnímu uživatelskému rozhraní značně zjednodušuje konfiguraci a inicializaci periférií MCU.



Obrázek 7.1: Schéma komunikace s MCU řady STM32

Obslužný firmware byl implementován v jazyce C s využitím vysokoúrovňové knihovny HAL (High Abstraction Layer). Knihovna HAL byla použita zejména pro její dobrou přenositelnost a možnost rychlé a přehledné konfigurace periférií. Případnou nevýhodnou, plynoucí z vyšší míry abstrakce této knihovny, může být nižší efektivita či omezená možnost konfigurace periférií.

Obslužný firmware byl vyvíjen za pomoci vývojového prostředí Atollic TrueSTUDIO. Jedná se o volně dostupné vývojové prostředí na bázi platformy Eclipse, které disponuje vestavěnými ladícími nástroji pro MCU rodiny STM32. Počáteční konfigurace periférií MCU byla provedena prostřednictvím aplikace STM32CubeMX.

<sup>1</sup><https://www.st.com/en/development-tools/stm32cubemx.html>

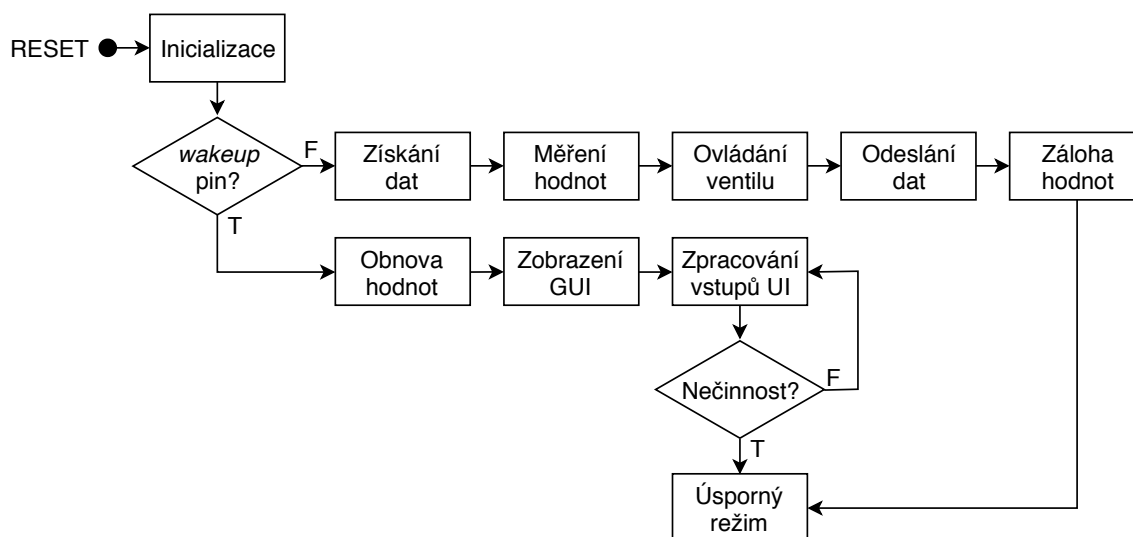
### 7.1.1 Princip činnosti

Obslužný firmware je založen na jádru, které poskytuje rozhraní pro obsluhu potřebných periférií, a tedy řízení modulu. Funkce z rozhraní jádra jsou následně volány z obslužných rutin přerušeni a odpovídajících pozic v hlavní funkci programu.

Prvním krokem po spuštění programu je zjištění, zda byl MCU probuzen z režimu úspory elektrické energie. Až poté je provedena inicializace systémových hodin a potřebných periférií. Probuzení z úsporného režimu může být iniciováno čítačem reálného času nebo uživatelem prostřednictvím tlačítkového spínače enkodéru. Poté následuje inicializace jádra a připojení komunikačního modulu k ZigBee síti.

Jestliže byl MCU probuzen čítačem reálného času nebo pouze resetován, dojde k získání aktuální požadované teploty od centrální jednotky. Následně jsou aktualizovány hodnoty měřených veličin čtením ze senzorů. Regulace vytápění je přizpůsobena aktuálním hodnotám a informace o jejím výsledném stavu jsou odeslány centrální jednotce. Poté přechází ZigBee modul i MCU zpět do úsporného režimu.

Bylo-li probuzení MCU iniciováno uživatelem, nedochází k automatické komunikaci s centrální jednotkou – hodnoty sledovaných veličin jsou obnoveny ze záložních registrů. Uživateli je zpřístupněno rozhraní sestávající z LCD displeje a rotačního enkodéru s vestavěným tlačítkovým spínačem. Možnosti uživatelského rozhraní jsou popsány v sekci 7.1.6. MCU přechází nazpět do úsporného režimu, je-li uživatel po určitou dobu neaktivní.



Obrázek 7.2: Základní princip činnosti firmware

### 7.1.2 Komunikace se ZigBee modulem

Komunikace s lokálním ZigBee modulem je realizována prostřednictvím sériového rozhraní UART. ZigBee modul pro komunikaci využívá tzv. *API* rámce<sup>2</sup>. Podporována je celá sada rámců, ovšem pro správné fungování firmware jsou stěžejní pouze následující z nich:

- *AT Command Frame* – rámec sloužící ke čtení či úpravě konfigurace modulu;
- *AT Command Response Frame* – rámec nesoucí odpověď na *AT Command Frame*;

<sup>2</sup><https://www.digi.com/resources/documentation/digidocs/90001539/>

- *Transmit Request Frame* – slouží k odeslání zprávy některému z uzlů ZigBee sítě;
- *Receive Packet Frame* – rámeček nesoucí příchozí zprávu ze ZigBee sítě.

Obsluha rozhraní UART pro komunikaci se ZigBee modulem je implementována v souboru `uart.c`. Velikost rámečků je relativně malá, což dovoluje jejich synchronní odesílání bez negativního vlivu na běh firmware. Odesílání rámečků zajišťuje funkce `uart_transmit_data`.

### Příjem rámeček

Příjem rámečků ze ZigBee modulu je plně asynchronní – implementace využívá přerušení na přijímací lince UART rozhraní a DMA přenos přijatých dat. Pomocí standardních technik čtení dat z UART rozhraní je možné číst pouze bloky dat o předem známé délce. Délka rámečků je ovšem variabilní, a tudíž jsou tyto techniky nepoužitelné. Z tohoto důvodu byla implementována obsluha UART rozhraní využívající techniku *idle-line detection*.

Technika *idle-line detection* spočívá v detekci okamžiku, kdy sběrnice přechází do stavu nečinnosti (*idle*). Tato událost generuje přerušení, čímž je uvědoměn procesor. V praxi to znamená, že za každým souvislým blokem dat následovaným pomlčkou o šířce alespoň 1 bit bude generováno přerušení indikující příjem dalšího rámečku či jeho fragmentu. Jelikož je používán DMA přenos, přijatý blok dat je při vstupu do obslužné rutiny přerušení již zapsán do odpovídajícího bufferu. Z obslužné rutiny přerušení je volána funkce `uart_check_rx_buffer`, která zkontroluje délku přijatých dat a předá je ke zpracování.

### Zpracování rámeček

Přijatý rámeček je ve funkci `uart_process_rx_data` zpracován pomocí konečného automatu po jednotlivých bytech. Tímto způsobem je ověřena jeho platnost a rovněž je zajištěna regenerace rámečku přijatého po částech (z důvodu chyby komunikace či naplnění kapacity DMA bufferu). Data, která rámeček obsahuje, jsou na základě jejich typu uložena do odpovídajícího bufferu. Využívána je sada bufferů pro uložení výsledků AT příkazů nebo pro uložení zpráv komunikačního protokolu popsaného v sekci 6.2. Data v těchto bufferech jsou zpracována v rámci standardního běhu programu. Tímto způsobem je možné postupně zpracovat více bezprostředně následujících rámečků bez jejich zahazování.

#### 7.1.3 Komunikace s centrální jednotkou

Komunikace s centrální jednotkou probíhá pomocí binárního protokolu popsaného v sekci 6.2. Jelikož modul hlavice periodicky přechází do režimu úspory energie včetně ZigBee modulu, je komunikace s centrální jednotkou iniciována pouze ze strany modulu. Generování a odesílání zpráv zajišťuje funkce `heat_send_request`. Této funkci je nutné předat typ zprávy, jež má být odeslána. Funkce na základě předaného typu sestaví tělo zprávy a prostřednictvím funkce `xbee_generate_tx_request_frame` jej zapouzdří do API rámečku k odeslání ZigBee modulu. Následuje odeslání rámečku výše uvedeným způsobem.

Zpracování přijaté zprávy je prováděno pomocí funkce `heat_process_response`. Tato funkce vyžaduje dva parametry – typ očekávané zprávy a časový interval, po který budou realizovány pokusy o příjem zprávy daného typu. Funkce cyklicky prochází sadu bufferů pro ukládání příchozích zpráv a obsah jednotlivých bufferů testuje na požadovaný typ zprávy. Tato činnost je opakována dokud není nalezena požadovaná zpráva nebo do uplynutí daného časového intervalu. Je-li zpráva nalezena, dojde k jejímu zpracování – typicky aktualizace odpovídajících hodnot.

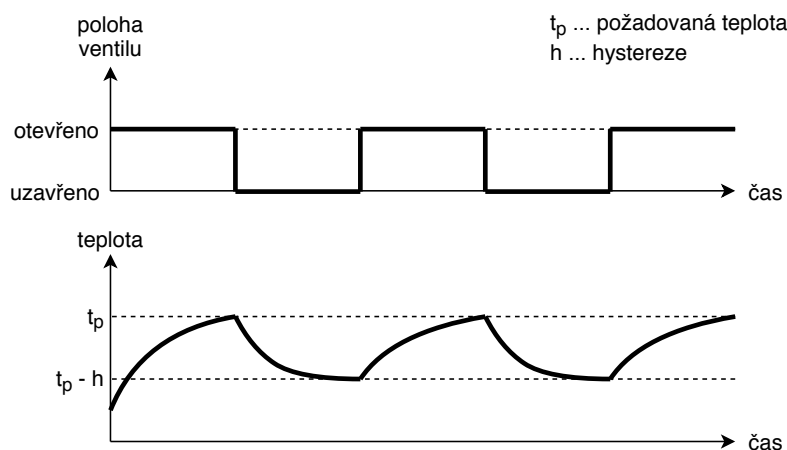


### 7.1.4 Regulace vytápění

Regulace vytápění je prováděna otevíráním a uzavíráním ventilu otopného tělesa. Ovládání ventilu je řízeno tzv. dvupolohovou regulací, která je implementována v rámci funkce `heat_regulation`. Tato regulace operuje se dvěma základními hodnotami – požadovanou teplotou a hysterezí. Princip dvupolohové regulace je následující:

1. je-li ventil otopného tělesa uzavřen a okolní teplota je nižší než rozdíl požadované teploty a hystereze, dojde k otevření ventilu;
2. pokud je ventil otopného tělesa otevřen a okolní teplota je vyšší nebo rovna požadované teplotě, ventil je uzavřen.

Průběh regulované veličiny, okolní teploty, charakteristický pro tento typ regulace ilustruje obrázek 7.3. Ze zobrazeného průběhu je patrné, že teplota je udržována v intervalu daném hysterezí. Dvupolohová regulace tedy dosahuje nižší přesnosti než například tzv. *PID* regulace. Pro své fungování ovšem vyžaduje mnohem méně ovládacích cyklů ventilu otopného tělesa, což je pro akumulátorem napájený systém zásadním parametrem.



Obrázek 7.3: Průběh dvupolohové regulace teploty

### 7.1.5 Režim úspory elektrické energie

Modul hlavičky otopného tělesa periodicky přechází do režimu *standby* za účelem úspory elektrické energie. V tomto režimu zůstává aktivní pouze čítač reálného času a registry záložní domény. Zbylé periferie jsou odpojeny od napájení, rovněž obsah paměti RAM je ztracen. Po opuštění režimu *standby* prochází MCU resetem, a musí tedy dojít k opětovné inicializaci všech periférií. Přechod zpět do aktivního režimu může být iniciován hardwarovým resetem, čítačem RTC nebo pomocí tzv. *wakeup* vstupu.

Periodického probouzení MCU z režimu *standby* je docíleno aktivací tzv. *wakeup* čítače, který je součástí periferie RTC. Tento čítač má šířku 16 b a je řízen hodinovým signálem RTC čítače. Pro úpravu rozlišení a maximální periody *wakeup* čítače lze vstupní hodinový signál upravit pomocí děliče, případně zafixovat na 1 MHz. Jelikož je žádoucí, aby modul hlavičky odesílal informace centrální jednotce v intervalech o délce v řádu jednotek až desítek minut, je *wakeup* čítač buzen hodinovým signálem o frekvenci 1 MHz. Perioda probouzení MCU tedy může být nastavena v rozmezí 1 s až 65 535 s.

Využito bylo také tzv. *wakeup* vstupu, ke kterému je připojen spínač rotačního enkodéru. MCU tedy může být probuzen i asynchronně uživatelem. Zdroj probuzení je nutné rozlišovat, aby byla dodržena perioda zasílání dat centrální jednotce. Dojde-li k probuzení čítačem, MCU je po provedení výměny dat s centrální jednotkou opětovně uspán na dobu odpovídající periodě probouzení. Perioda probouzení je součástí dat, které modul hlavice obdrží od centrální jednotky v rámci periodické komunikace. Pokud je MCU probuzen uživatelem, komunikace neprobíhá a délka následného setrvání v režimu *standby* je snížena o dobu již v tomto režimu strávenou. Tohoto chování je docíleno uložením obsahu čítače RTC před každým uspáním MCU. Při následném probuzení tedy může být vypočtena dosavadně strávená doba v úsporném režimu.

Jak již bylo řečeno, přechodem do režimu *standby* je ztracen obsah paměti RAM, potřebná data tedy musí být zálohována. K zálohování dat je možné použít registry záložní domény periferie RTC. Použitý MCU disponuje pěticí registrů o velikosti 32 bitů. Zálohována jsou pouze konfigurační data a aktuální stav modulu.

### 7.1.6 Uživatelské rozhraní

Uživatelské rozhraní hlavice otopného tělesa sestává z grafického LCD displeje a rotačního enkodéru vybaveného tlačítkovým spínačem. Součástí obslužného firmware musí být obsluha použitého LCD displeje, detekce sepnutí spínače a detekce otáčení enkodéru. LCD displej je ovládán za pomoci knihovny `Nokia-LCD5110-HAL`<sup>3</sup>, která disponuje funkcemi pro vykreslení znaků, řetězců a základních grafických primitiv (úsečka, obdélník). Knihovna rovněž zajišťuje inicializaci a konfiguraci displeje. Knihovna ke svému fungování vyžaduje soubor `font.h`, ve kterém je definována binární reprezentace vykreslovaných znaků. Aby nebylo uživatelské rozhraní omezeno pouze na základní sadu znaků, byly v rámci souboru `graphics.h` definovány další sady znaků a symbolů.

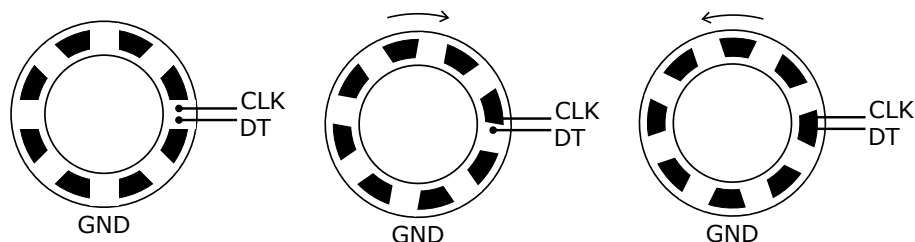
Použitý rotační enkodér je založen na mechanickém principu a disponuje dvěma výstupy označovanými jako CLK a DT, případně A a B. Obsluha rotačního enkodéru vychází z jeho vnitřní struktury, která je znázorněna na obrázku 7.4. Detekce změny polohy rotoru enkodéru využívá generování externího přerušení na pinu MCU, ke kterému je připojen CLK výstup enkodéru. Přerušení je generováno se sestupnou i vzestupnou hranou signálu. Výstup enkodéru DT je připojen ke vstupnímu pinu MCU. Z obslužné rutiny přerušení je volána funkce `heat_process_encoder`. V této funkci je přečtena logická hodnota na výstupu enkodéru. Následuje porovnání získaných hodnot, na základě kterého je určen směr otáčení rotoru enkodéru.

Součástí obslužného firmware je také obsluha tlačítkového spínače vestavěného do rotačního enkodéru. Pro správné fungování uživatelského rozhraní je třeba detekovat sepnutí tohoto spínače a rovněž jeho délku. Této funkce je docíleno generováním externího přerušení se sestupnou i vzestupnou hranou signálu na výstupu spínače. Obslužnou rutinou přerušení je volána funkce `heat_process_button`, ve které jsou nastaveny příznaky a zaznamenány časy obou událostí.

### Grafické uživatelské rozhraní

Grafické uživatelské rozhraní je složeno ze čtyř obrazovek odpovídajících režimu, ve kterém se modul nachází. Poté, co je modul hlavice probuzen prostřednictvím spínače enkodéru, je zobrazena hlavní obrazovka (obrázek 7.5a). V horní části hlavní obrazovky je umístěna

<sup>3</sup><https://github.com/Zeldax64/Nokia-LCD5110-HAL>



Obrázek 7.4: Princip činnosti mechanického rotačního enkodéru

stavová lišta. Lišta obsahuje ikony reflektující stav akumulátoru, polohy ventilu otopného tělesa a stav spárování s centrální jednotkou. Na této liště je rovněž zobrazena požadovaná teplota v dané místnosti. Zbývající prostor obrazovky je využit k zobrazení aktuální okolní teploty.

Stiskem spínače enkodéru po dobu 2 s je možné z hlavní obrazovky přejít do režimu párování. Na displeji je zobrazena obrazovka se symbolem signalizujícím probíhající párování (obrázek 7.5b). Jakmile dojde k úspěšnému spárování s centrální jednotkou, je opět zobrazena hlavní obrazovka.

Krátkým stiskem spínače enkodéru lze přejít k nastavení jednorázového programu. Prvním parametrem jednorázového programu je požadovaná teplota, kterou lze upravit pomocí enkodéru. Mimo konkrétní teplotu lze nastavit také možnost *auto*. Potvrzením této možnosti dojde k deaktivaci existujícího jednorázového programu a regulace vytápění bude pokračovat dle aktuálního denního programu. Obrazovka zobrazená na LCD displeji během režimu nastavení teploty je zachycena na obrázku 7.5c.

Potvrzením konkrétní teploty přejde modul do režimu nastavení délky platnosti jednorázového programu. Opět lze nastavit konkrétní časový interval nebo aktivovat program trvale. Trvalé aktivování programu je provedeno potvrzením volby *perm*. Obrazovka odpovídající tomuto režimu je zobrazena na obrázku 7.5d.

Je-li uživatel neaktivní po dobu 5 s, přejde modul zpět na hlavní obrazovku, případně dojde k přechodu do úsporného režimu.

## 7.2 Modul pro ovládání zdroje tepla

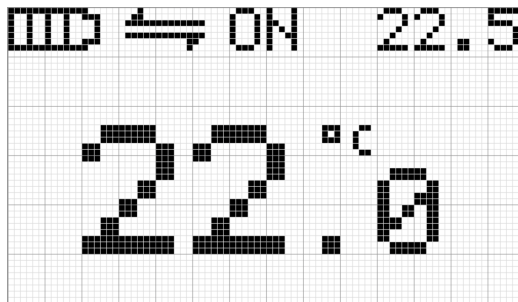
Hlavní funkcí tohoto modulu je ovládání zdroje tepla na základě příkazů od centrální jednotky. Aby mohl být modul plnohodnotnou součástí systému regulace vytápění, musí být také schopen projít procedurou párování s centrální jednotkou.

Použitý ZigBee modul disponuje podporou programovacího jazyka MicroPython<sup>4</sup>. Modul pro ovládání zdroje tepla tedy může být řízen přímo ZigBee modulem bez přítomnosti separátního MCU. MicroPython je minimalistická implementace programovacího jazyka Python 3, jež je optimalizovaná pro běh na MCU a v obecně omezeném prostředí. MicroPython obsahuje pouze malou podmnožinu standardní knihovny jazyka Python, disponuje ovšem knihovnou pro přístup k hardwaru na nízké úrovni.

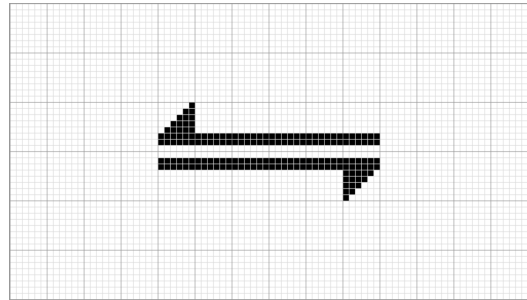
Pro konfiguraci a nahrání firmware ZigBee modulu byl použit nástroj XCTU<sup>5</sup>. Tento nástroj, mimo jiné, nabízí terminál, prostřednictvím kterého lze interagovat s implementací jazyka MicroPython na ZigBee modulu. Terminál umožňuje kompilovat zdrojový kód, na-

<sup>4</sup><https://micropython.org/>

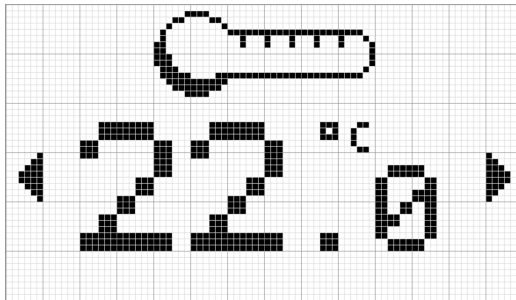
<sup>5</sup><https://www.digi.com/products/iot-platform/xctu>



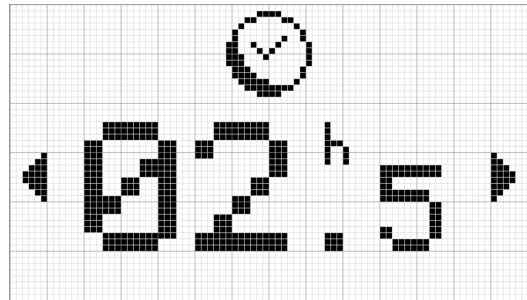
(a) Hlavní obrazovka



(b) Obrazovka párování



(c) Obrazovka nastavení teploty programu



(d) Obrazovka nastavení trvání programu

Obrázek 7.5: Obrazovky grafického uživatelského rozhraní

hrávat a spouštět výsledný program. Rovněž také umožňuje uložení kompilovaného kódu do FLASH paměti modulu a jeho automatické spuštění se spuštěním ZigBee modulu.

Implementace obslužného firmwaru využívá knihovny `xbee` pro obsluhu rádiové části modulu a také pro ovládání GPIO pinů. Ovládání GPIO pinů je řešeno prostřednictvím lokálních AT příkazů, jelikož knihovna `machine` umožňující přímý přístup k perifériím nebyla pro použitý ZigBee modul dostupná.

### 7.2.1 Princip činnosti

Spuštěním modulu dojde k inicializaci programu uloženého ve FLASH paměti ZigBee modulu a započne jeho vykonávání. Nejdříve je proveden pokus o spárování s centrální jednotkou (funkce `pair`). Na adresu centrální jednotky je pomocí funkce `sendPairingRequest` odeslána žádost o spárování. Pokud není v předem stanoveném časovém intervalu přijata pozitivní odpověď, proces se opakuje. Přijetím pozitivní odpovědi proces párování končí.

Následuje vstup do hlavní smyčky programu, ve které jsou zpracovávány příchozí zprávy. Součástí hlavní smyčky je také periodická výměna zpráv s centrální jednotkou za účelem získání aktuálních informací potřebných k řízení zdroje tepla. Jak již bylo předesláno v sekci 6.2, modul je schopen žádosti spojené s ovládáním zdroje tepla nejen přijímat, ale také odesílat. Komunikace tedy může být iniciována jak ze strany centrální jednotky, tak i ze strany modulu.

Zasílání dotazů na centrální jednotku probíhá v časových intervalech předem stanovené délky. Časový okamžik odeslání žádosti je zaznamenán, a jestliže není odpověď od centrální jednotky včas přijata, zdroj tepla je z bezpečnostních důvodů deaktivován. Opětná aktivace je provedena pouze na základě budoucí korektně přijaté odpovědi či přímého příkazu ze strany centrální jednotky.

Ovládání zdroje tepla je prováděno pomocí elektromagnetického relé. Relé je ovládáno výstupním pinem ZigBee modulu, jenž je dodatečně posílen tranzistorem. Řízení relé zajišťuje funkce `powerBoilerOn`.

## Kapitola 8

# Návrh a implementace ovládací aplikace

Tato kapitola je věnována návrhu a implementaci mobilní aplikace sloužící pro monitorování a řízení systému regulace vytápění. Bude popsána struktura implementované aplikace a princip fungování jejích stěžejních částí.

Mobilní aplikace je jediným prostředkem pro interakci uživatele se systémem regulace vytápění. Aplikace by svým grafickým rozhraním měla uživateli vhodným způsobem interpretovat získaná data, na základě kterých bude uživatel rozhodovat o dalším nastavení systému. Grafické rozhraní by rovněž mělo umožňovat intuitivním způsobem řídit činnost celého systému. Základní požadavky na funkcionalitu aplikace jsou následující:

- prezentování aktuální teploty a stavu modulů v jednotlivých místnostech (zónách),
- zobrazení a správa programů vytápění pro jednotlivé místnosti a dny v týdnu,
- nastavení jednorázových programů vytápění,
- tvorba a správa místností,
- registrace a správa systémových modulů,
- tvorba a správa uživatelských účtů.

Mezi další funkce, kterými by aplikace měla disponovat, patří například zobrazení průběhu regulace vytápění v čase a odpovídajících statistik či možnost rychlého nastavení nepřítomnosti osob v objektu.

### 8.1 Technologie

Mobilní aplikace byla vyvíjena pro platformu Google Android. Tato platforma byla zvolena zejména kvůli jejímu vysokému zastoupení v oblasti chytrých telefonů a tabletů.

Pro vývoj aplikace bylo využito integrované vývojové prostředí Android Studio. Android Studio nad rámec editoru zdrojového kódu a základních vývojářských nástrojů nabízí také integrovaný emulátor systému Android, nástroje pro profilování a testování kompatibility aplikací či využití šablon usnadňujících tvorbu aplikací. Android Studio využívá systém Gradle, který zajišťuje řízení a automatizaci sestavování aplikace. Tímto je zajištěna možnost přizpůsobení procesu sestavování aplikace a obecně jeho flexibilita.

Implementace programové části aplikace byla realizována v programovacím jazyce Java. Jazyk Java je jedním ze dvou oficiálně podporovaných jazyků pro vývoj aplikací na platformě Android. Od této skutečnosti se odvíjí míra dokumentovanosti implementačních prostředků a také dostupnost knihoven.

### 8.1.1 Struktura aplikace systému Android

Základním stavebním prvkem aplikace pro platformu Android je třída `Activity`. Způsob, jakým jsou aktivity propojeny a spouštěny, určuje činnost aplikace. Aplikace pro systém Android mají na rozdíl od klasických počítačových aplikací více možných vstupních bodů. Spuštění aplikace je realizováno tak, že systém Android iniciuje zdrojový kód instance třídy `Activity` voláním metod odpovídajících specifické fázi jejího životního cyklu. Aplikace se většinou skládají z více aktivit, kdy jedna z nich je typicky označována jako *main activity*. Tato aktivita je zahájena jako první po spuštění aplikace uživatelem. Všechny aktivity, ze kterých se aplikace skládá a jejich atributy musí být zapsány do tzv. *manifestu*.

Z hlediska uživatelského rozhraní každá aktivita poskytuje vlastní okno, do kterého je vykresleno uživatelské rozhraní aplikace.

Uživatelské rozhraní v rámci aktivity lze rozšiřovat pomocí třídy `Fragment`, jež reprezentuje určitý úsek chování a uživatelského rozhraní aktivity. Životní cyklus instancí třídy `Fragment` podléhá životnímu cyklu odpovídající aktivity.

## 8.2 Backend

Implementovaná aplikace je založena na dvou aktivitách. Aktivita reprezentovaná třídou `LoginActivity` je aktivována při spuštění aplikace uživatelem. Tato aktivita zajišťuje přihlášení či registraci uživatele. Po úspěšném přihlášení, případně registraci, je tato aktivita ukončena a proběhne spuštění aktivity reprezentované třídou `MainActivity`. Třída `MainActivity` zajišťuje veškerou funkcionalitu spojenou s řízením a monitorováním systému regulace vytápění.

### 8.2.1 Komunikace s centrální jednotkou

Komunikace s centrální jednotkou probíhá pomocí textového komunikačního protokolu ve formátu JSON. Pro přenos zpráv byl na straně aplikace implementován TCP klient s využitím socketů. TCP klient je reprezentován třídou `TcpClient`, která disponuje metodami pro navázání a uzavření spojení, odeslání dat a příjem dat. Jelikož se vlákno aktivity stará o chod uživatelského rozhraní, musí být síťová komunikace a obecně všechny časově náročné operace vykovávány v samostatných vláknech. Za tímto účelem byla implementována třída `RequestTask`, která s využitím třídy `TcpClient` zajišťuje asynchronní odeslání a příjem dat v samostatném vlákně.

Třída `RequestTask` rozšiřuje třídu `AsyncTask`. Konstruktor je využit pro předání parametrů spojení, typu zprávy a jejího obsahu. Předávána je rovněž instance rozhraní, pomocí jehož metod je možné interagovat s rodičovským vlákem. Mimo konstruktor jsou přetěžovány metody `onPreExecute` a `onPostExecute`, které jsou volány před a po vykonání úlohy v samostatném vlákně. V těchto metodách jsou volány metody z předaného rozhraní. Další přetěžovanou metodou je `doInBackground`, která obsahuje úsek zdrojového kódu, který má být vykonán v samostatném vlákně. V rámci této metody je na základě předaného typu a dat sestaveno tělo zprávy k odeslání. Následně je vytvořeno spojení s centrální jednotkou

a zpráva je odeslána. Poté je proveden pokus o příjem odpovědi. Proces příjmu dat je realizován v separátním vlákně, jelikož je blokující a musí být možné ho zastavit v případě, že data nejsou přijata v rámci určitého časového intervalu. Výsledek úlohy a případně i přijatá data jsou předány rodičovskému vlákně prostřednictvím metody z předané instance rozhraní.

Instance třídy `RequestTask` jsou využívány výhradně v rámci metody `sendRequest`, která je součástí tříd `MainActivity` a `LoginActivity`. Tyto třídy obsahují také implementaci rozhraní, prostřednictvím kterého mohou být získána přijatá data. Přijatá data jsou zpracována metodou `processResponse`.

Data přijatá od centrální jednotky jsou uchovávána ve strukturách reprezentovaných třídou `HeatData`.

## 8.2.2 Správa přihlašovacích údajů

Jelikož aplikace disponuje systémem uživatelských účtů, je třeba určitým způsobem zajistit bezpečný přenos a uložení přihlašovacích údajů. Přihlašovací jména mohou být přenášena a ukládána v textové podobě bez vniku bezpečnostního rizika. S hesly by ovšem tímto způsobem nemělo být nakládáno. Vhodnou metodou je využití hash funkce, která na základě zadaného hesla vygeneruje unikátní hash hodnotu. Tato hodnota může být odeslána a uložena do databáze. Při ověřování přihlašovacích údajů již nejsou porovnávána hesla v textové podobě, ale jim odpovídající hash hodnoty.

Pro generování hash hodnot hesel byl použit algoritmus PBKDF2 (Password Based Key Derivation Function). Tento algoritmus generuje klíč o požadované délce na základě uživatelem zadaného hesla a tzv. soli. Generování probíhá opakovaným voláním určité pseudonáhodné funkce, v tomto případě HMAC-SHA1, jejímž výstupem je hash určité délky. Sůl je veřejná hodnota jisté délky zvyšující odolnost vygenerované hodnoty vůči slovníkovým útokům. Hash funkce je aplikována na konkatenaci soli a hesla.

Generování hash hodnoty na straně aplikace je zajištěno třídou `PasswordHashTask`. Tato třída rozšiřuje třídu `AsyncTask`, výpočet je tedy asynchronní a vykonáván v samostatném vlákně. Instance této třídy může operovat ve dvou režimech. První režim je používán při registraci uživatele, kdy je vygenerována náhodná sůl pomocí objektu `SecureRandom`. Náhodná sůl je následně použita pro vygenerování hash hodnoty. Druhý režim slouží pro vygenerování hash hodnoty na základě soli získané z databáze, což je využito při ověření přihlašovacích údajů.

## Registrace a přihlášení

Zadané přihlašovací či registrační údaje nejdříve prochází základní kontrolou na straně aplikace. Po jejich kontrole následuje proces registrace či přihlášení.

V případě registrace je nejprve vygenerována hash hodnota hesla a spolu se solí a přihlašovacím jménem je odeslána centrální jednotce. Na centrální jednotce jsou data uložena do databáze a uživateli je umožněn vstup do hlavní aktivity aplikace.

Proces přihlášení je o jeden komunikační cyklus delší, jelikož použitý algoritmus vyžaduje sůl, která je uložena v databázi na straně centrální jednotky. Prvním krokem je odeslání požadavku obsahujícího pouze přihlašovací jméno uživatele. Na základě přihlašovacího jména je z databáze získána sůl, jež je následně odeslána zpět mobilní aplikaci. Na straně mobilní aplikace proběhne vygenerování hash hodnoty na základě zadaného hesla a získané soli. Výsledná hodnota je opět spolu s přihlašovacím jménem odeslána na cent-



rální jednotku, kde je vygenerovaná hash hodnota porovnána s hodnotou uloženou v databázi. Jsou-li hodnoty shodné, je uživateli umožněn vstup do hlavní aktivity aplikace.

## 8.3 Frontend

Jak již bylo v předchozí sekci uvedeno, aplikace využívá dvou aktivit – aktivity pro přihlášení či registraci uživatele a hlavní aktivity zajišťující funkcionalitu spojenou s monitorováním a řízením systému regulace vytápění. Základní struktura uživatelského rozhraní se odvíjí od aktuálně běžící aktivity. Obrázky ilustrující vybrané části uživatelského rozhraní jsou součástí přílohy F.

Společným prvkem uživatelského rozhraní obou aktivit je horizontální nástrojová lišta, která uživateli usnadňuje orientaci v rámci aplikace. V případě aktivity pro přihlášení tato lišta nabízí možnost nastavení síťové adresy centrální jednotky. Je-li aktivní hlavní aktivita, je na této liště umístěno tlačítko pro zobrazení navigační lišty.

Při prvním spuštění aplikace je nejdříve zobrazen přihlašovací formulář. Volbou ve spodní části obrazovky lze přejít na formulář registrace uživatele. Poté, co jsou vyplněny přihlašovací údaje, lze provést přihlášení.

Součástí přihlašovacího formuláře je také možnost zapamatování přihlašovacích údajů. Je-li tato možnost vybrána, dojde k uložení přihlašovacího jména a vygenerované *hash* hodnoty hesla do perzistentního úložiště. Uložení je prováděno prostřednictvím objektu `SharedPreferences`. Tento objekt umožňuje zápis párů klíč a hodnota do souboru přístupného pouze z dané aplikace. Při každém dalším spuštění aplikace jsou uložené hodnoty ze souboru načteny a přihlášení je provedeno automaticky.

### 8.3.1 Přehled

Po vstupu do hlavní aktivity je zobrazen přehled místností ve formě dlaždic. Každá dlaždice obsahuje název místnosti, informace o aktuální a požadované teplotě a také sadu ikon reflektujících stav modulů v dané místnosti. Zobrazení dlaždic přímo závisí na datech přijatých od centrální jednotky, jsou tedy dynamické. Zobrazování dynamických seznamů a rozložení je možné provádět pomocí tzv. adaptérů, které zajišťují výběr dat potřebných pro každou dynamickou entitu a také její vytvoření. Zobrazení dlaždic zajišťuje třída `HomeGridAdapter` rozšiřující třídu `BaseAdapter`.

Ve spodní části obrazovky je umístěna trojice tlačítek. Dvě z nich slouží pro aktivaci speciálních režimů, které mohou být nastaveny pouze uživateli v roli administrátora systému. Platnost režimů lze nastavit trvale nebo po určitou dobu. Režimy jsou následující:

- *mimo domov* – teplota ve všech místnostech je nastavena na nižší teplotu (výchozí nastavení je 15 °C),
- *dovolená* – regulace vytápění je řízena denním programem volného dne (výchozí nastavení je sobota).

Kromě tlačítek pro nastavení režimů se ve spodní části obrazovky nachází také tlačítko pro vypnutí či zapnutí celého systému. Výběrem dlaždice z přehledu je možné přejít na detail místnosti. Funkcionalita detailu místnosti je zajištěna třídou `RoomFragment`.

## Detail místnosti

Detail místnosti uživateli sděluje požadovanou a aktuální teplotu v dané místnosti. Zobrazeny mohou být také ikony indikující stav systémových modulů v místnosti. V pravém horním rohu je umístěna dvojice plovoucích tlačítek, pomocí kterých je možné přejít k nastavení denního programu dané místnosti či zobrazit sekci statistik. Kliknutím na oblast požadované teploty lze přejít na nastavení jednorázového programu.

Většina prostoru obrazovky nastavení jednorázového programu je využita k zobrazení voliče požadované teploty. Mimo volič teploty se zde nachází také tlačítka pro aktivaci programu. Program může být aktivní dočasně nebo trvale. V případě dočasněho programu je uživatel před aktivací vyzván k nastavení časového okamžiku, po jehož dosažení bude program automaticky deaktivován. Potvrzením nastavení je požadavek na aktivaci programu odeslán centrální jednotce a následně je opětovně zobrazen detail místnosti.

## Denní program

Pro nastavení denního programu je nutné nejdříve vybrat den, pro který bude program místnosti upravován. Po provedení výběru je zobrazen seznam úseků programu. Každý úsek je reprezentován časovým okamžikem a teplotou, jaká má být od daného okamžiku v místnosti udržována. Položky seznamu jsou zobrazovány za pomoci třídy `ProgramListAdapter`. V dolní části obrazovky se nachází trojice tlačítek pro zkopírování všech úseků programu do jiných dnů, přidání úseku a odstranění všech úseků daného dne.

Kliknutím na položku ze seznamů úseků lze přejít na detail daného úseku. Obrazovka detailu nabízí volič požadované teploty a také tlačítko pro vyvolání dialogu nastavení časového okamžiku udávajícího rozsah platnosti úseku. Nachází se zde také tlačítko pro odstranění daného úseku denního programu. Po modifikaci některého z parametrů úseku je zobrazeno tlačítko pro uložení změn.

## Statistiky

Aplikace umožňuje také zobrazení denních statistik průběhu regulace vytápění jednotlivých místností. Obrazovka statistik, spravovaná fragmentem `StatisticsFragment`, je z velké části tvořena spojnicovým grafem, který ilustruje průběh teploty uvnitř místnosti v rámci daného dne. V oblasti grafu jsou rovněž vyznačeny časové intervaly, po které byl aktivní zdroj tepla, díky čemuž může mít uživatel detailní přehled o činnosti systému. Oblast nad grafem zaujímá volič data, pro které má být statistika zobrazena. Volič umožňuje sekvenci procházení po jednotlivých dnech nebo také umožňuje vyvolání dialogu pro výběr konkrétního dne z kalendáře. V dolní části obrazovky jsou umístěny údaje shrnující průběh regulace vytápění v rámci daného dne. Je zde zobrazena například celková doba, po kterou byl aktivní zdroj tepla.

Standardní knihovny pro platformu Android nedisponují funkcionalitou pro vykreslování grafů. Graf průběhu regulace vytápění je vykreslován s použitím knihovny `androidplot`<sup>1</sup>, která podléhá Apache licenci, je tedy volně dostupná a použitelná. Knihovna `androidplot` podporuje mnoho typů grafů, u kterých je možné využít pokročilé funkce jako interpolace a normalizace hodnot či dynamické překreslování v samostatném vlákně. Pro účely statistik byl použit graf reprezentovaný třídou `XYPlot`, jeho inicializaci zajišťuje metoda `updateUI`.

---

<sup>1</sup><http://androidplot.com/>

### 8.3.2 Nastavení

Sekce nastavení umožňuje uživateli provádět základní nastavení aplikace a systému regulace vytápění. Dále umožňuje správu místností, systémových modulů a uživatelů. Nastavení ovlivňující chod systému mohou upravovat pouze uživatelé v roli administrátora, ostatním uživatelům jsou nedostupné volby skryty. V sekci základního nastavení lze upravit parametry režimů *mimo domov* a *dovolená*, hysterezi regulace teploty a také síťovou adresu centrální jednotky.

#### Místnosti

Předpokladem pro fungování systému je existence alespoň jedné místnosti (zóny), se kterou může být následně svázán modul hlavice otopného tělesa. Pro vytvoření místností a jejich správu slouží právě sekce nastavení místností. Po vstupu do této sekce je zobrazen seznam aktuálně existujících místností v rámci systému. Pod tímto seznamem je zobrazeno tlačítko pro přidání nové místnosti.

Výběrem položky ze seznamu místností dojde k přechodu na detail místnosti. Obrazovka detailu nabízí ovládací prvky pro úpravu parametrů či odstranění místnosti. Lze zde také povolit či zakázat úpravu programu regulace vytápění prostřednictvím modulu hlavice umístěného v dané místnosti.

#### Systémové moduly

Hlavním účelem sekce nastavení systémových modulů je možnost jejich přidělení do místností, případně jejich odstranění. Po vstupu do této sekce je zobrazen seznam systémových modulů organizovaných do skupin podle místnosti, do které jsou přiděleny. V samostatné skupině jsou zobrazeny také moduly, které prozatím nejsou přiděleny. Ve spodní části obrazovky je umístěno tlačítko pro aktualizaci seznamu modulů.

Kliknutím na položku již přiděleného modulu je možné zobrazit jeho detail. Detail modulu poskytuje základní informace o modulu – např. jeho adresa v rámci ZigBee sítě nebo název místnosti, do které je přidělen. Rovněž se zde nachází ovládací prvky pro nastavení časového intervalu, ve kterém má modul zasílat informační zprávy centrální jednotce.

Výběrem položky některého z doposud nepřidělených modulů dojde k zobrazení dialogu obsahující seznam aktuálně dostupných místností. Následně je na uživateli, do které místnosti bude modul přidělen.

#### Uživatelské účty

Sekce nastavení uživatelských účtů má dvě základní funkce: nastavení profilu uživatele a nastavení práv přístupu k jednotlivým místnostem. První funkce je přístupná všem uživatelům bez ohledu na jejich zařazení. Druhá funkce je přístupná pouze uživatelům v roli administrátora systému. Uživatelům v roli administrátora je umožněno spravovat také účty ostatních uživatelů. Po vstupu do sekce uživatelských účtů je zobrazen seznam uživatelů systému, pomocí kterého lze přejít na detail uživatele.

Detail uživatele umožňuje nastavení jména uživatele a jeho přiřazení do role administrátora. Nachází se zde také seznam místností opatřených přepínači, které reflektují existenci práva přístupu uživatele k dané místnosti. Uživateli, který není administrátorem, je tento seznam zobrazen, ale není mu povoleno provádět změny.

## Kapitola 9

# Použití systému

V této kapitole bude popsán průběh uživatelského testování navrženého systému. Budou zhodnoceny dosažené výsledky. Na základě dat získaných uživatelským testem bude stanovena spotřeba elektrické energie systémových modulů. Na závěr kapitoly budou navrženy možné optimalizace a rozšíření systému.

### 9.1 Uživatelský test

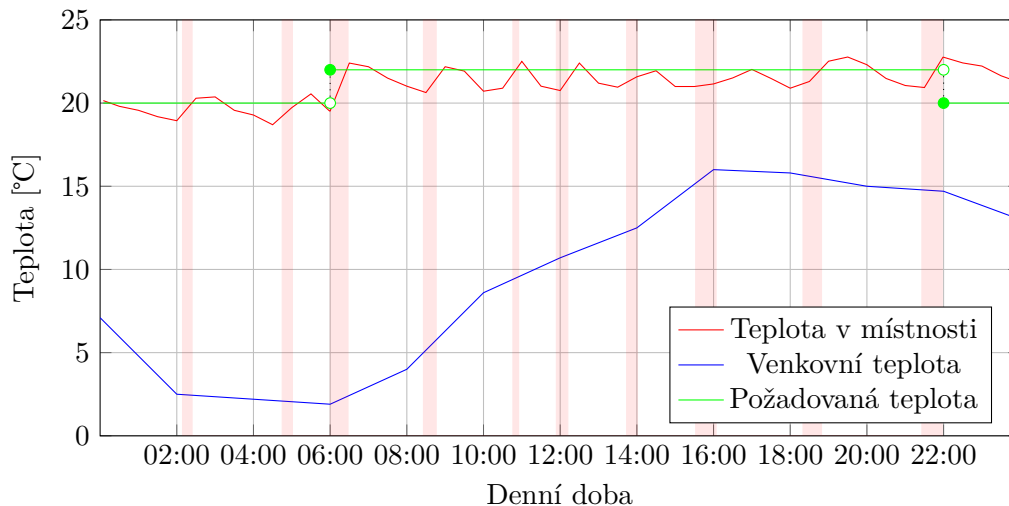
Pro účely uživatelského testu byla využita místnost o přibližné rozloze  $19\text{ m}^2$ , ve které se nacházelo jedno otopné těleso. Zeď, na které bylo instalováno otopné těleso, byla zdí obvodovou. Jako zdroj tepla byl použit kombinovaný plynový kotel NOVATHERM ZW 20 s nominálním výkonem  $20\text{ kW}$  na ohřev vody pro topný okruh. Otopná tělesa nacházející se mimo testovací místnost byla ponechána s ventily v otevřeném stavu. Modul hlavice byl instalován na otopné těleso v testovací místnosti. Centrální jednotka byla umístěna ve vzdálenosti cca  $4\text{ m}$  od otopného tělesa v totožné místnosti. Modul pro ovládání zdroje tepla byl umístěn ve vzdálenosti cca  $3,5\text{ m}$  od centrální jednotky za dvěma zdmi.

Perioda probouzení modulu hlavice otopného tělesa byla nastavena na  $300\text{ s}$ . V těchto intervalech tedy docházelo k výměně dat s centrální jednotkou, vyhodnocení požadované a okolní teploty a případné úpravy polohy ventilu otopného tělesa. Hystereze regulace vytápění byla nastavena na  $1,5\text{ }^\circ\text{C}$ .

#### 9.1.1 Průběh testu

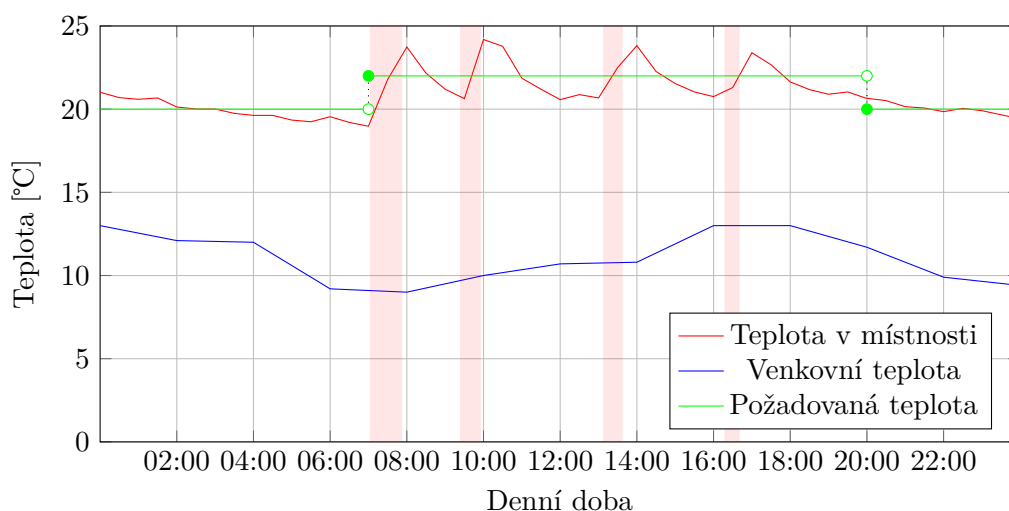
Test probíhal po dobu tří dnů, během kterých se venkovní teplota pohybovala v rozmezí  $0\text{ }^\circ\text{C}$  až  $19\text{ }^\circ\text{C}$ . Program regulace vytápění byl v průběhu testu upravován dle potřeb uživatele.

První den testování byl 8. květen 2019. Program regulace vytápění pro tento den byl složen ze tří úseků. Požadovaná teplota pro úseky  $0:00$  až  $5:59$  a  $22:00$  až  $23:59$  byla  $22\text{ }^\circ\text{C}$ . Úsek mezi  $6:00$  a  $21:59$  měl přidělenou požadovanou teplotu  $20\text{ }^\circ\text{C}$ . Průměrná venkovní teplota daného dne byla  $10,9\text{ }^\circ\text{C}$ . Průběhy sledovaných veličin jsou zaneseny do grafu na obrázku 9.1 ve 30minutových intervalech. V tomto grafu jsou rovněž vyznačeny časové úseky, ve kterých byl aktivní zdroj tepla. Detailní přehled hodnot je součástí přílohy G. Na základě naměřených hodnot byla pro každý úsek programu vytápění nejdříve vypočtena průměrná teplota uvnitř místnosti a následně stanovena odchylka této hodnoty od odpovídající požadované teploty. Odchylky od požadované teploty pro jednotlivé úseky jsou  $0,414\text{ }^\circ\text{C}$ ,  $0,342\text{ }^\circ\text{C}$  a  $2,539\text{ }^\circ\text{C}$ . Vysoká hodnota odchylky pro poslední úsek programu je způsobena tepelnou setrvačností otopného tělesa.



Obrázek 9.1: Graf průběhu regulace vytápění – 8. květen 2019

Dalším dnem, kdy probíhal uživatelský test, byl 9. květen 2019. Program regulace vytápění opět sestával ze tří úseků. Požadované teploty těchto úseků byly totožné, ovšem změnil se jejich hranice. Časové intervaly úseků byly následující: 00:00 až 6:59, 7:00 až 19:59 a 20:00 až 23:59. Průměrná venkovní teplota pro tento den byla 11,76 °C. Průběhy sledovaných hodnot v čase jsou zaneseny do grafu na obrázku 9.2. Na základě informací o průběhu regulace vytápění z předešlého dne byla zavedena hystereze také pro přechod ze stavu aktivního vytápění do stavu neaktivního vytápění. Cílem této úpravy bylo prodloužení intervalů ovládání ventilů otopných těles a spínání zdroje tepla. Prodloužení intervalů má za následek snížení spotřeby elektrické energie modulů hlavice otopných těles.



Obrázek 9.2: Graf průběhu regulace vytápění – 9. květen 2019

Pro úseky denního programu vytápění byly stanoveny odchylky průměrné hodnoty teploty v místnosti od teploty požadované. Způsob výpočtu odchylek byl popsán výše. Díky nově zavedené hysterezi průměrná teplota v místnosti více odpovídá požadované teplotě, což

má za následek značné snížení velikosti odchylek jednotlivých úseků programu. Vypočtené odchylky jsou 0,097 °C, 0,18 °C a 0,028 °C.

Uživatelský test posledního dne již probíhal beze změn v procesu regulace vytápění a sloužil zejména k důkladnému ověření správného fungování systému. Hodnoty naměřené během posledního dne uživatelského testu jsou uvedeny v příloze G.

### 9.1.2 Shrnutí

Uživatelský test systému regulace vytápění probíhal celkem tři dny. Na základě dat získaných během prvního dne testování byla navržena optimalizace v podobě zavedení hystereze pro přechod ze stavu aktivního vytápění do stavu neaktivního vytápění. Během následujících dvou dnů testování již regulace probíhala s nově zavedenou hysterezi. Analýzou naměřených dat bylo zjištěno, že došlo ke značnému snížení odchylky průměrné teploty v místnosti oproti požadované teplotě. Rovněž došlo k prodloužení intervalů ovládní ventilu otopného tělesa, a tím i ke snížení spotřeby elektrické energie modulu hlavice otopného tělesa.

## 9.2 Spotřeba elektrické energie

Spotřeba elektrické energie je kritická zejména pro modul hlavice otopného tělesa, jelikož je tento modul napájen z lithium-iontového akumulátoru. Na spotřebu elektrické energie modulu má zásadní vliv délka intervalů mezi ovládacími cykly ventilu otopného tělesa. S rostoucí délkou intervalů dochází ke zkracování doby, po kterou je aktivní servomotor mechanismu pro ovládní ventilu otopného tělesa, a tedy i snížení spotřeby elektrické energie. Dalším faktorem ovlivňujícím spotřebu elektrické energie je perioda probouzení MCU hlavice z režimu *standby*. Po probuzení MCU je probuzen také ZigBee modul za účelem komunikace s centrální jednotkou. Životnost akumulátoru lze prodloužit nalezením optimálního nastavení programu regulace vytápění a periody probouzení modulu.

Nyní bude stanovena přibližná spotřeba elektrické energie modulu hlavice otopného tělesa na základě dat získaných z uživatelského testu. Pro stanovení celkové spotřeby je nejdříve nutné určit spotřebu jednotlivých komponent modulu (uvedené hodnoty byly naměřeny při napájecím napětí 3,3 V):

- MCU včetně podpůrných komponent – režim *run* 4,2 mA, režim *standby* 850 nA;
- ZigBee modul – příjem 17 mA, vysílání 40 mA, režim *power-down* 28 µA;
- servomotor – bez zátěže 110 mA, v zátěži 200 mA;
- LCD displej – 150 µA.

Následně je nutné určit dobu, po kterou komponenty setrvávají v jednotlivých režimech činnosti. Během režimu úspory je odebíraný proud všemi komponentami 28,85 µA. V aktivním režimu stráví ZigBee modul odesíláním dat 0,224 ms při přenosové rychlosti 250 kb/s (režie přenosu byla zanedbána), po zbytek času je ZigBee modul v režimu příjmu dat. Je-li modul hlavice aktivní, průměrný odebíraný proud je 21,202 mA. Tato hodnota odpovídá jednomu úseku aktivity modulu, během kterého nebyl ovládnut ventil otopného tělesa. Modul setrvává v aktivním stavu po dobu 4,45 s. Během úseku aktivity, kdy byl ovládnut ventil otopného tělesa, je průměrný odebíraný proud 145,51 mA po dobu 22,45 s.

Z dat získaných uživatelským testem lze odvodit přibližnou skladbu životního cyklu modulu. Během 72 h, po které test probíhal, došlo celkem k 17 aktivacím zdroje tepla, a tedy i otevření ventilu otopného tělesa. Po dosažení cílové teploty byl ventil vždy opětovně uzavřen. Servomotor byl tedy aktivován celkem 34krát. Perioda probouzení modulu byla nastavena na 300 s, což znamená, že modul byl probuzen a opětovně uspán 864krát. Skladba režimů je tedy následující:

- 0,294 % – aktivní režim s ovládním ventilu otopného tělesa,
- 1,425 % – aktivní režim bez ovládním ventilu otopného tělesa,
- 98,291 % – režim úspory elektrické energie.

Z uvedených údajů vyplývá, že průměrný odebíraný proud všemi komponentami modulu je 730  $\mu$ A. Je-li uvažováno napájení akumulátorem o kapacitě 3450 mAh, modul hlavice je při dané skladbě režimů schopen provozu po dobu až 197 dnů.

### 9.3 Optimalizace a rozšíření

Hlavním předmětem případné optimalizace by mohla být hlučnost a spotřeba elektrické energie mechanismu pro ovládním ventilu otopného tělesa. Použitý servomotor by mohl být nahrazen modelem, který bude mít při zachování kroutícího momentu a počtu otáček za minutu nižší spotřebu elektrické energie. Nicméně servomotor těchto parametrů nebyl během realizace práce na trhu nalezen. Případně by bylo možné celou koncepci mechanismu upravit po vzoru komerčních řešení. Takovýto přístup je vzhledem k požadavkům na rozměry výsledného zařízení v amatérských podmínkách obtížně realizovatelný.

Případné rozšíření by mohlo spočívat v možnosti řízení a monitorování vytvořeného systému prostřednictvím Internetu. Předpokladem pro realizaci této úpravy je možnost adresace centrální jednotky v rámci Internetu. Dále by bylo také nutné řešit zabezpečení komunikace mezi centrální jednotkou a mobilní aplikací. Současné řešení ke komunikaci využívá unixové sockety, které nevyužívají žádný zabezpečovací mechanismus. Zabezpečení komunikace by bylo možné zajistit využitím kryptografických protokolů TLS/SSL. Tyto protokoly vytváří mezivrstvu mezi transportním protokolem TCP a aplikačním protokolem. Použitím TLS/SSL je zajištěna důvěrnost, integrita a také autentizace jedné či obou komunikujících stran.

Případně by bylo možné využít určitého typu cloudového řešení, kam by byla periodicky odesílána data reflektující aktuální stav systému. Bylo by tak umožněno monitorování systému v rámci Internetu, ovšem úprava parametrů regulace vytápění či nastavení systému by nebyla možná.

# Kapitola 10

## Závěr

V rámci této diplomové práce byla vypracována studie existujících systémů pro automatické řízení centrálního vytápění. Rovněž byly prostudovány bezdrátové komunikační technologie a metody napájení použitelné v oblasti automatizace domácnosti.

Na základě získaných poznatků byla navržena architektura systému pro automatické řízení a monitorování centrálního vytápění. Navržený systém zahrnuje modul centrální jednotky založený na miniaturním počítači Raspberry Pi a modul pro ovládání ventilu otopného tělesa řízený MCU architektury ARM Cortex-M0+. Součástí systému je také modul pro ovládání zdroje tepla, tedy elektrického či plynového kotle. Komunikace systémových modulů využívá technologii ZigBee. Za tímto účelem byly systémové moduly vybaveny bezdrátovými moduly Digi XBee3 ZigBee 3.0.

Následně byl proveden návrh systémových modulů na obvodové úrovni a jejich fyzická realizace. Realizace modulů obnášela zajištění výroby desek plošných spojů, jejich osazení a následné oživení. Tato fáze zahrnovala také návrh a realizaci mechanismu pro ovládání ventilu otopného tělesa, jež je součástí vytvořeného prototypu hlavice otopného tělesa.

Po fyzické realizaci systémových modulů následovala implementace řídicí aplikace a obslužného firmware. Řídicí aplikace zajišťuje správu perzistentního úložiště dat, komunikaci se systémovými moduly a instancemi mobilní aplikace. Společně s řídicí aplikací byl vyvíjen také obslužný firmware modulu hlavice otopného tělesa. Firmware zajišťuje výměnu dat s centrální jednotkou, měření teploty v místnosti a řízení polohy ventilu otopného tělesa na základě získaných dat. Za účelem snížení spotřeby elektrické energie firmware zajišťuje připojování napájecích domén dle potřeby a přechod do úsporného režimu. Součástí implementovaného firmware je také obsluha uživatelského rozhraní modulu.

Modul pro ovládání zdroje tepla je řízen firmware běžícím přímo na ZigBee modulu. Firmware se stará o komunikaci s centrální jednotkou – jak pasivní, tak i aktivní. Na základě dat přijatých od centrální jednotky zajišťuje řízení činnosti zdroje tepla.

Dále byla navržena a implementována mobilní aplikace pro platformu Android. Aplikace uživateli přehledně interpretuje informace o průběhu regulace vytápění. Průběh regulace je možné přizpůsobovat potřebám uživatele úpravou denních či jednorázových programů vytápění. Aplikace rovněž disponuje systémem uživatelských účtů a přístupových práv.

Vytvořený systém byl podroben uživatelskému testu za účelem ověření jeho správné funkčnosti. Uživatelským testem bylo zjištěno, že se průměrná odchylka naměřené teploty v místnosti od požadované teploty pohybuje nejvýše v řádech desetin stupně Celsia. Na základě naměřených dat byla rovněž odvozena přibližná spotřeba elektrické energie modulu hlavice otopného tělesa. Modul je za podmínek, jež panovaly během uživatelského testu, schopen provozu po dobu 197 dnů na jedno nabití akumulátoru.



Výsledkem práce je funkční systém pro automatické řízení a monitorování regulace vytápění. Systém ke svému fungování využívá platformu Raspberry Pi a běžně dostupný hardware, čímž je umožněna jeho integrace s dalšími otevřenými automatizačními systémy. Vytvořený systém by bylo vhodné rozšířit o možnost jeho správy prostřednictvím Internetu. Předpokladem pro implementaci takového rozšíření je možnost adresace centrální jednotky v rámci Internetu. Komunikaci mimo lokální síť by bylo vhodné adekvátně zabezpečit, například použitím kryptografických protokolů SSL/TLS. Případně by mohlo být využito cloudové řešení umožňující akumulaci a následnou interpretaci naměřených dat uživateli. Toto řešení by ovšem neumožňovalo vzdálené ovládání systému.

# Literatura

- [1] Ahmed, F.; Annus, P.; Le Moullec, Y.: *Energy Harvesting Technologies: Potential Application to Wearable Health-Monitoring*. 01 2015.
- [2] Bluetooth Special Interest Group: *Bluetooth Specification Version 5.1* [online]. [cit. 17.11.2018].  
URL <https://www.bluetooth.com/specifications/bluetooth-core-specification>
- [3] Chandra, P.: *Wireless security*. Amsterdam: Newnes, 2009, ISBN 978-1-85617-529-6.
- [4] Cragie, R.: *ZigBee Security*. 2009, zigBee Alliance ZARC Security Task Group. [cit. 15.01.2019].  
URL <http://modsec.zimmerle.org/wireless-sec-papers/zigbee%20-%20sec.pdf>
- [5] Digi International: *Digi XBee3 Zigbee 3*. [cit. 15.01.2019].  
URL <https://www.digi.com/products/embedded-systems/rf-modules/2-4-ghz-modules/xbee3-zigbee-3>
- [6] Dvořák, P.: *Superkondenzátory*. [cit. 18.11.2018].  
URL <https://oze.tzb-info.cz/teorie-obnovitelna-energie/6710-superkondenzatory>
- [7] Gislason, D.: *Zigbee wireless networking*. Amsterdam; Boston: Elsevier/Newnes, 2003, ISBN 978-0-7506-8597-9.
- [8] Halper, M. S.; Ellenbogen, J. C.: *Supercapacitors: A Brief Overview*. McLean, US: MITRE Nanosystems Group, 2006, 41 s. [cit. 14.01.2019].  
URL [https://www.mitre.org/sites/default/files/pdf/06\\_0667.pdf](https://www.mitre.org/sites/default/files/pdf/06_0667.pdf)
- [9] Hersent, O.; Boswarthick, D.; Elloumi, O.: *The internet of things: applications to the smart grid and building automation*. Hoboken: John Wiley & Sons, 2012.
- [10] Honeywell: *HR90EE TheraPro: programovatelná termostatická hlavice*. [cit. 30.04.2019].  
URL <http://www.honeywell.cz/home/Odb.clanky/HR90EE-CZ.pdf>
- [11] Honeywell: *Regulační systém - EvoHOME*. [cit. 30.04.2019].  
URL [http://www.honeywell.cz/home/Odb.clanky/Honeywell\\_EvoHome.pdf](http://www.honeywell.cz/home/Odb.clanky/Honeywell_EvoHome.pdf)
- [12] Koton, J.; Číka, P.; Křivánek, V.: *Standard nízkorychlostní bezdrátové komunikace ZigBee*. [cit. 17.11.2018].  
URL <http://access.feld.cvut.cz/view.php?cisloclanku=2006032001>

- [13] Padgette, J.; Bahr, J.; Batra, M.; aj.: *Guide to Bluetooth Security*. 4th Watch Publishing Co., 2017, ISBN 978-1547074006.
- [14] Poole, I.: *IEEE 802.11 Wi-Fi Standards*. [cit. 18.11.2018].  
URL <https://www.radio-electronics.com/info/wireless/wi-fi/ieee-802-11-standards-tutorial.php>
- [15] Raspberry Pi Foundation: *Raspberry Pi 3 Model B*. [cit. 15.01.2019].  
URL <https://www.raspberrypi.org/products/raspberry-pi-3-model-b>
- [16] Roundy, S.; Steingart, D.; Frechette, L.; aj.: Power Sources for Wireless Sensor Networks. In *Wireless Sensor Networks*, editace H. Karl; A. Wolisz; A. Willig, Springer, Berlin, Heidelberg, 2004.
- [17] Saurav, P.: *Batteries for IoT devices*. [cit. 15.01.2019].  
URL <https://www.baseapp.com/iot/batteries-iot-device-types-character/>
- [18] Sauter, M.: *From GSM to LTE-Advanced: An Introduction to Mobile Networks and Mobile Broadband, Revised Second Edition*. John Wiley & Sons, 2014, ISBN 9781118861950.
- [19] Schiller, J.: *Mobile Communications*. Addison-Wesley, 2003, ISBN 0-321-12381-6.
- [20] Softlink: *FLEXIM - zónová regulace topení*, Softlink s.r.o. 2016 [cit. 30.04.2019].  
URL <https://www.softlink.cz/flexi/>
- [21] Softlink: *WACOtherm - zónová regulace topení s radiovým ovládáním I*. 2011 [cit. 30.04.2019].  
URL <https://vytapeni.tzb-info.cz/mereni-a-regulace/8117-wacotherm-zonova-regulace-topeni-s-radiovym-ovladanim-i-popis-systemu>
- [22] Upton, E.: *Raspberry Pi : uživatelská příručka*. Brno: Computer Press, druhé vydání, 2016, ISBN 978-80-251-4819-8.
- [23] Visconti, P.; Primiceri, P.; Ferri, R.; aj.: An Overview on State-of-art Energy Harvesting Techniques and Choice Criteria: a WSN Node for Goods Transport and Storage Powered by a Smart Solar-based EH System. *International Journal of Renewable Energy Research*, ročník 7, 09 2017: s. 1281–1295.

## Příloha A

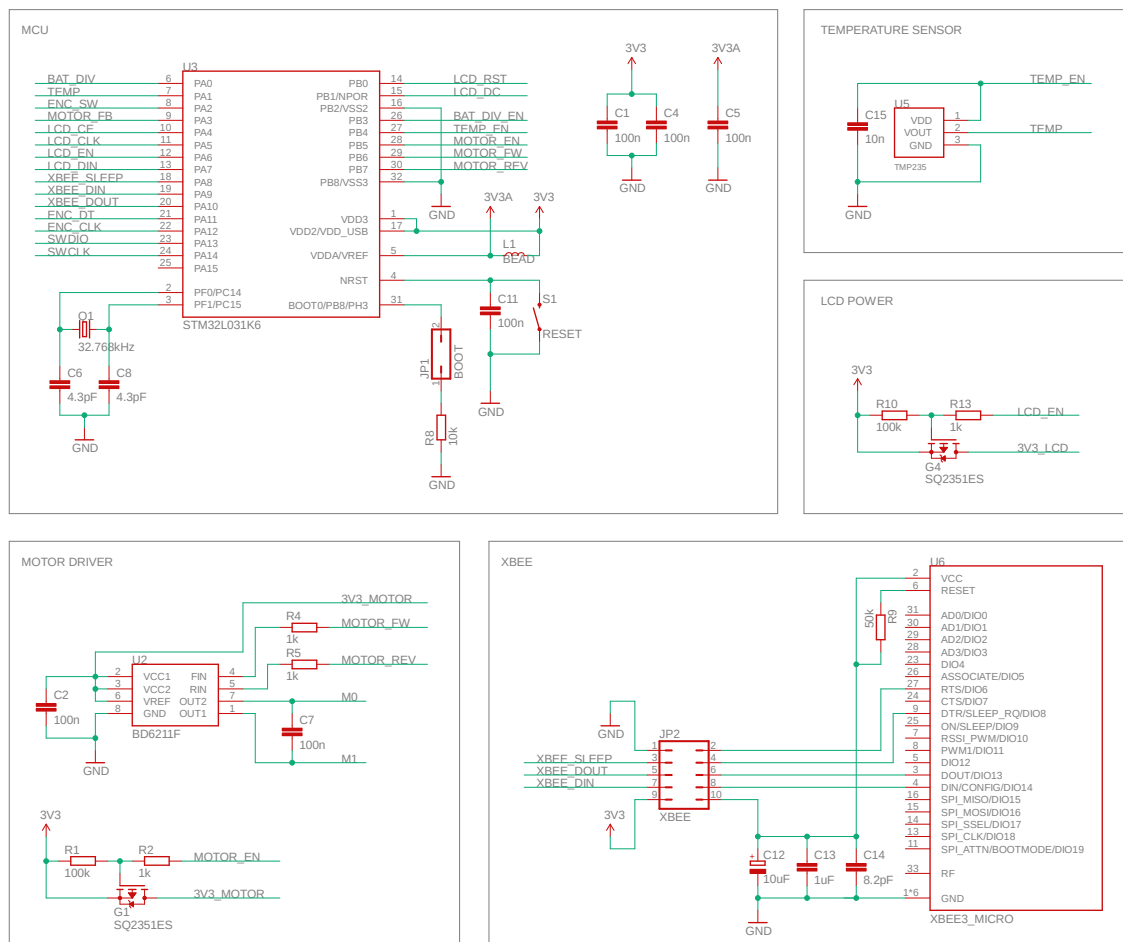
# Obsah přiloženého paměťového média

/	
├── src/	.....zdrojové soubory
│   ├── central_module/	.....řídící aplikace
│   ├── radiator_module/	.....modul otopného tělesa
│   ├── boiler_module/	.....modul pro ovládání zdroje tepla
│   └── app/	.....aplikace pro Android Studio
│       └── heat.apk	.....balík aplikace pro systém Android
├── hw/	.....zdrojové soubory schémat a desek plošných spojů
├── media/	
│   ├── screenshots/	.....snímky obrazovek aplikace
│   └── photos/	.....snímky vytvořených modulů
├── test_data/	.....data získaná uživatelským testem
├── doc/	.....zdrojové soubory technické zprávy
└── xsocma00.pdf	.....technická zpráva

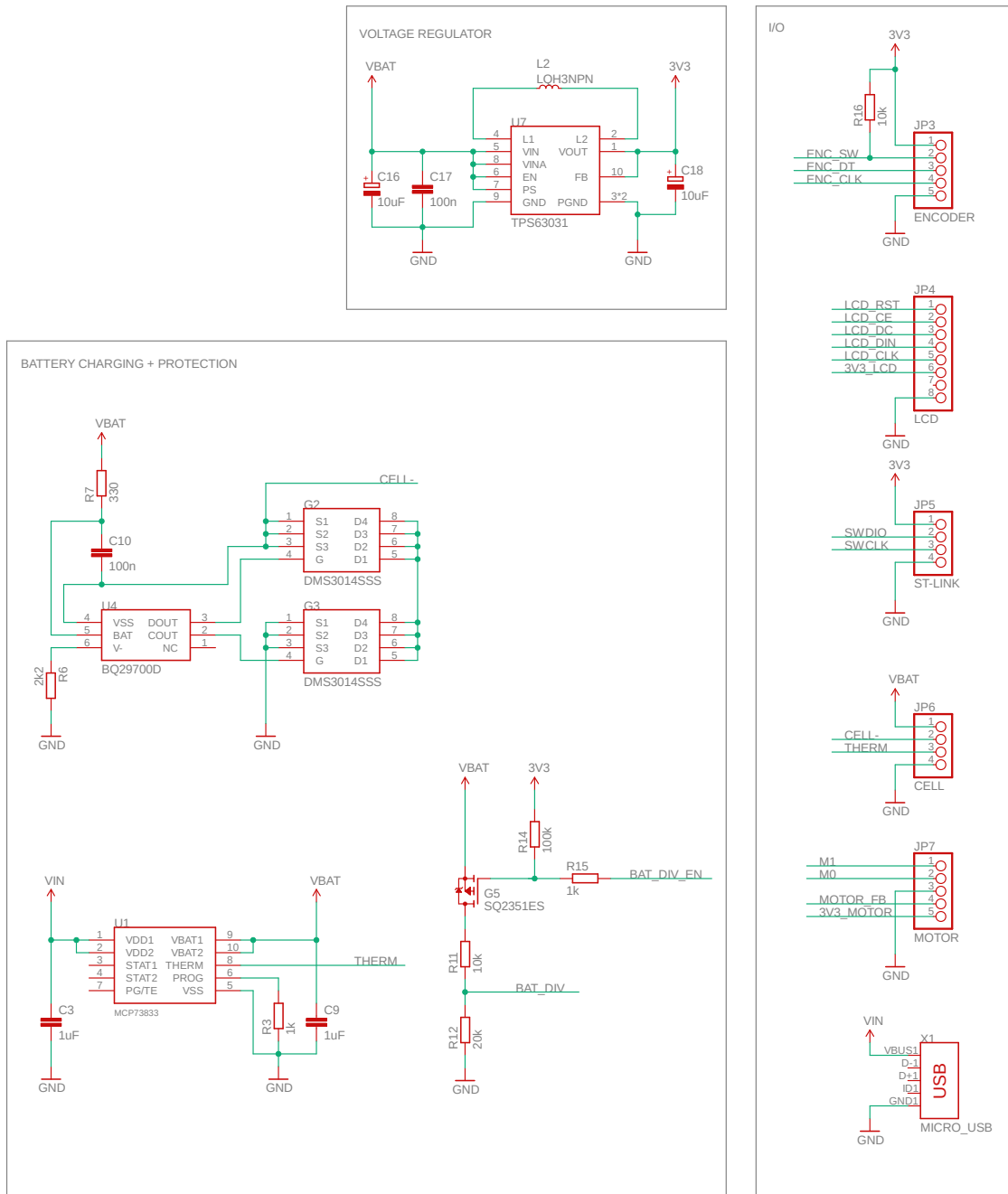
# Příloha B

## Schémat systémových modulů

Tato příloha obsahuje schémata systémových modulů, které byly vytvořeny v rámci této práce. Soubory schémat a odpovídající předloh desek plošných spojů pro software EAGLE se nachází na přiloženém paměťovém médiu.



Obrázek B.1: Schéma modulu hlavice otopného tělesa – první část



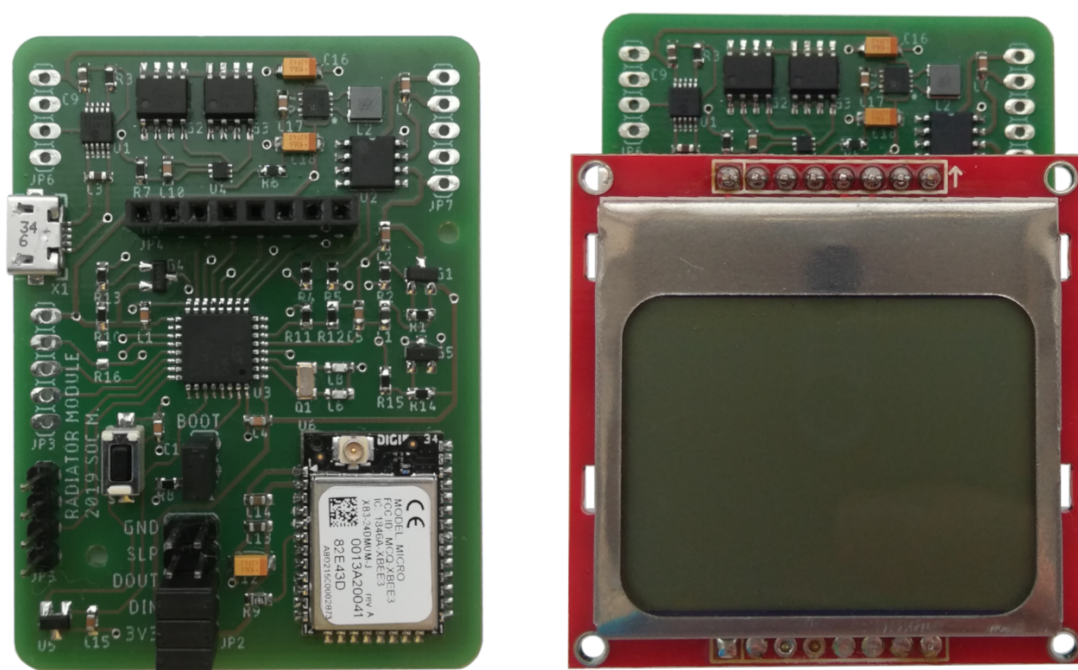
Obrázek B.2: Schéma modulu hlavice otopného tělesa – druhá část



## Příloha C

# Snímky systémových modulů

Součástí této přílohy jsou snímky všech vytvořených systémových modulů. Snímky v plném rozlišení lze nalézt na přiloženém paměťovém médiu.

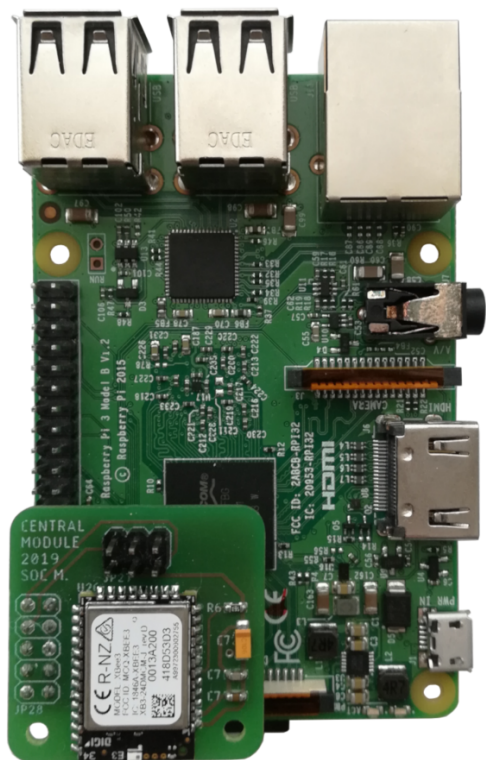


(a) Modul hlavice otopného tělesa

(b) Modul hlavice otopného tělesa včetně displeje

Obrázek C.1: Snímky vytvořených modulů – první část





(a) Modul centrální jednotky

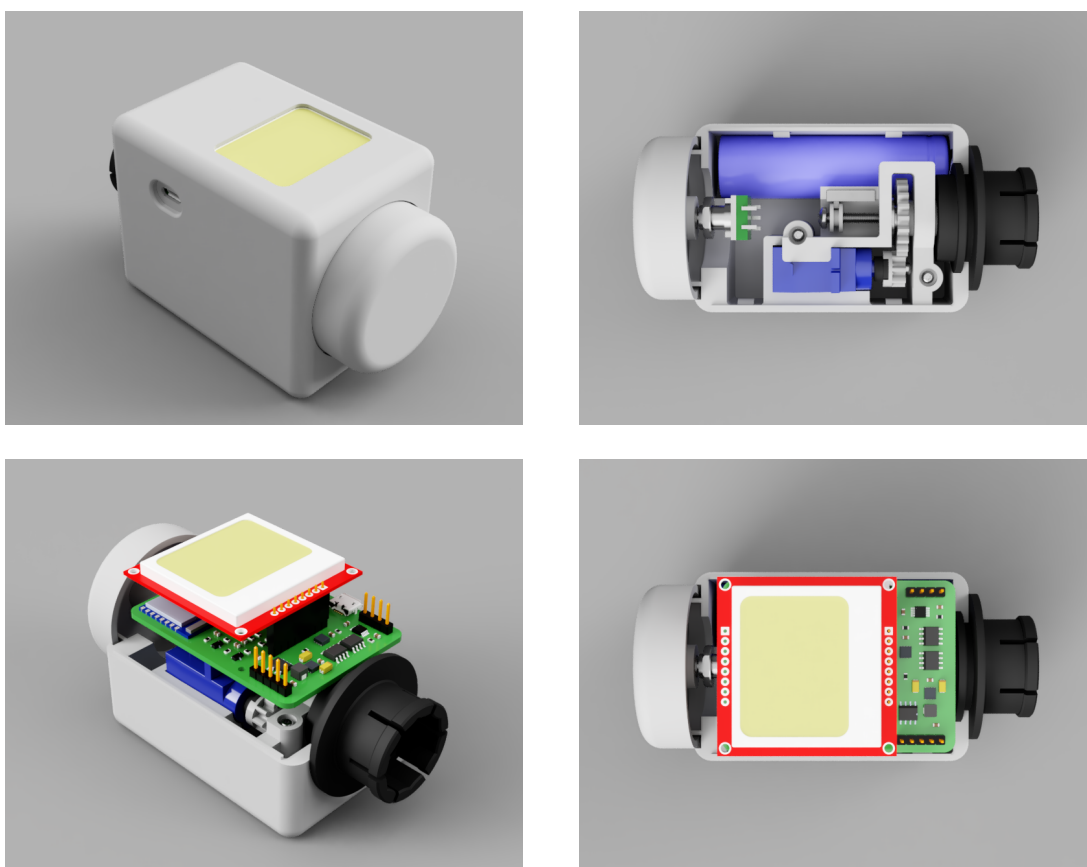


(b) Modul pro ovládání zdroje tepla

Obrázek C.2: Snímky vytvořených modulů – druhá část

## Příloha D

# Snímky prototypu hlavice otopného tělesa



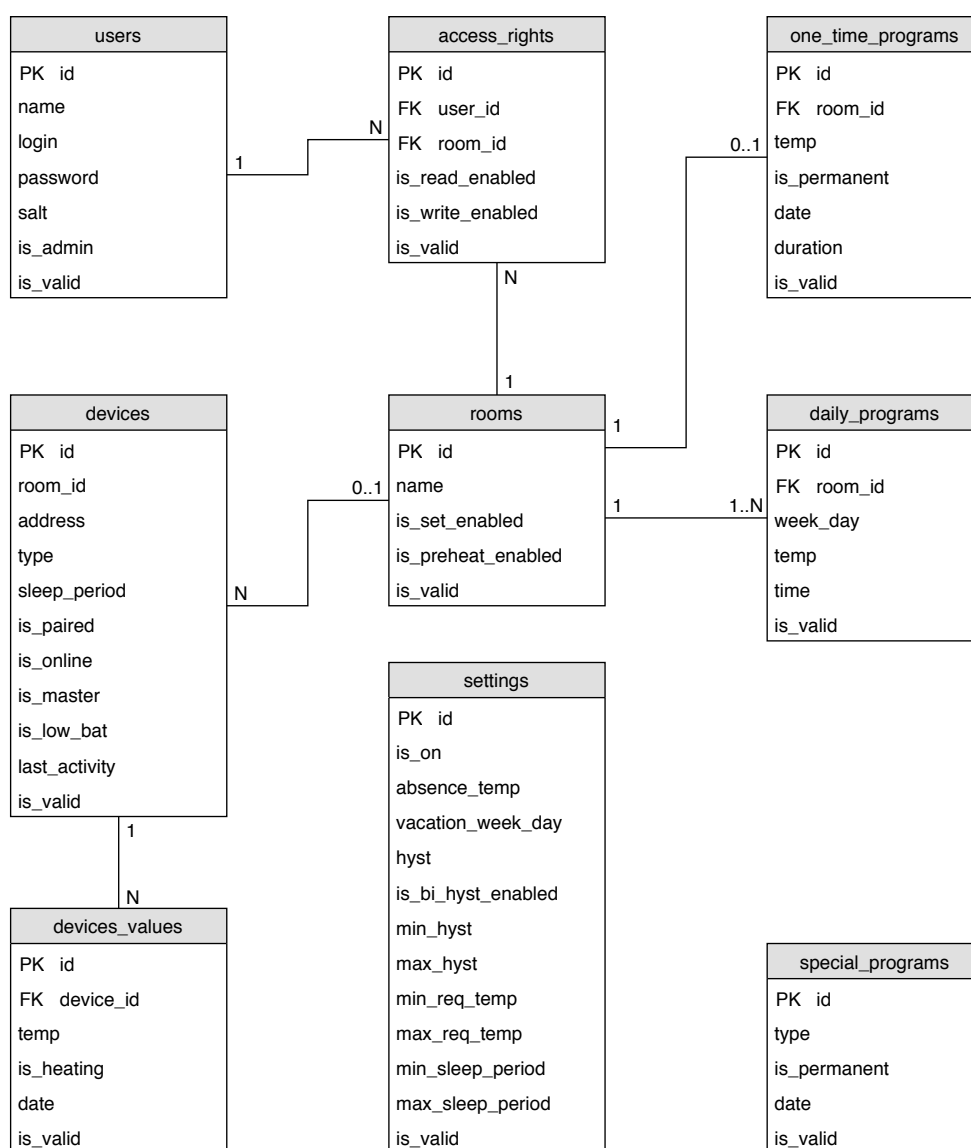
Obrázek D.1: Snímky modelu hlavice otopného tělesa



Obrázek D.2: Snímky prototypu hlavice otopného tělesa

# Příloha E

## ER diagram databáze

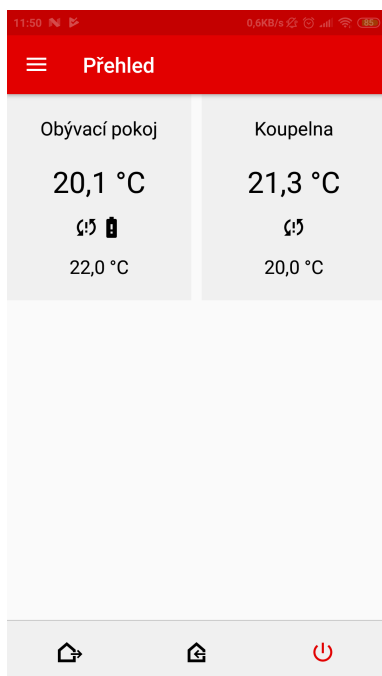


Obrázek E.1: ER diagram databáze

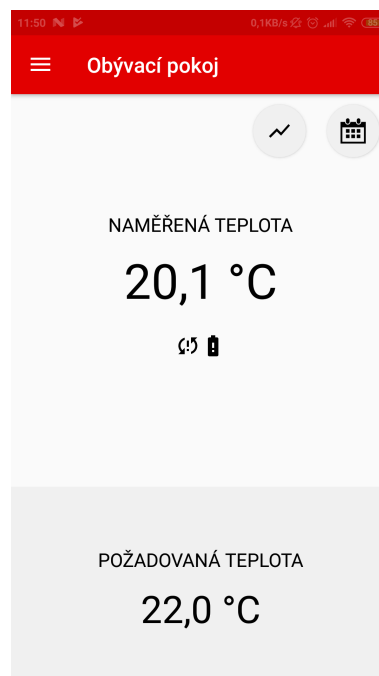
## Příloha F

# Snímky obrazovek mobilní aplikace

Tato příloha obsahuje snímky obrazovek vytvořené mobilní aplikace. Více snímků lze nalézt na přiloženém paměťovém médiu.

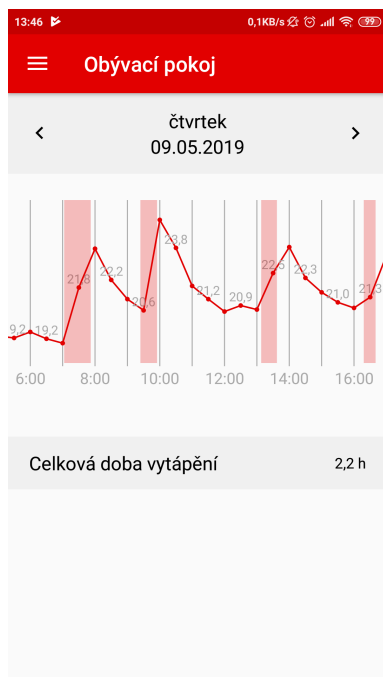


(a) Přehled



(b) Detail místnosti

Obrázek F.1: Snímky obrazovek mobilní aplikace – první část



(a) Denní statistika



(b) Nastavení úseku programu vytápění

Přihlášení

Přihlašovací jméno

Heslo

Zapamatovat

PŘIHLÁSIT

Registrovat

(c) Přihlašovací formulář

Obrázek F.2: Snímky obrazovek mobilní aplikace – druhá část

## Příloha G

# Naměřená data během uživatelského testu

V této příloze jsou uvedena data získaná během třídenního uživatelského testu. Data jsou rozdělena do tří tabulek po jednotlivých dnech. Jelikož je sada získaných dat poměrně rozsáhlá, jsou v tabulkách uvedeny pouze vzorky ve 30minutových intervalech. Kompletní sadu dat lze nalézt na přiloženém paměťovém médiu.

Denní doba	Teplota [°C]	Aktivita	Denní doba	Teplota [°C]	Aktivita
0:05	20,149	0	12:00	20,754	1
0:30	19,807	0	12:30	22,406	0
1:00	19,565	0	13:00	21,197	0
1:30	19,182	0	13:30	20,955	0
2:00	18,941	0	14:00	21,580	1
2:30	20,290	0	14:30	21,942	0
3:00	20,371	0	15:00	20,996	0
3:30	19,565	0	15:30	20,996	0
4:00	19,283	0	16:00	21,151	1
4:30	18,699	0	16:30	21,511	0
5:00	19,726	1	17:00	22,023	0
5:30	20,552	0	17:30	21,479	0
6:00	19,525	0	18:00	20,895	0
6:30	22,406	0	18:30	21,298	1
7:00	22,184	0	19:00	22,515	0
7:30	21,499	0	19:30	22,768	0
8:00	21,016	0	20:00	22,305	0
8:30	20,633	1	20:30	21,479	0
9:00	22,184	0	21:00	21,056	0
9:30	21,922	0	21:30	20,935	1
10:00	20,713	0	22:00	22,756	0
10:30	20,895	0	22:30	22,413	0
11:00	22,507	0	23:00	22,224	0
11:30	21,016	0	23:30	21,640	0

Tabulka G.1: Data získaná během uživatelského testu – 8. květen 2019

Denní doba	Teplota [°C]	Aktivita	Denní doba	Teplota [°C]	Aktivita
0:00	21,016	0	12:00	20,572	0
0:30	20,693	0	12:30	20,875	0
1:00	20,593	0	13:00	20,673	0
1:30	20,673	0	13:30	22,507	1
2:00	20,129	0	14:00	23,816	0
2:30	20,008	0	14:30	22,265	0
3:00	20,008	0	15:00	21,539	0
3:30	19,746	0	15:30	21,036	0
4:00	19,626	0	16:00	20,754	0
4:30	19,626	0	16:30	21,298	1
5:00	19,344	0	17:00	23,393	0
5:30	19,243	0	17:30	22,668	0
6:00	19,545	0	18:00	21,640	0
6:30	19,202	0	18:30	21,177	0
7:00	18,981	0	19:00	20,895	0
7:30	21,781	1	19:30	21,036	0
8:00	23,735	0	20:00	20,653	0
8:30	22,164	0	20:30	20,512	0
9:00	21,197	0	21:00	20,149	0
9:30	20,633	1	21:30	20,069	0
10:00	25,186	0	22:00	19,847	0
10:30	23,776	0	22:30	20,049	0
11:00	21,862	0	23:00	19,908	0
11:30	21,197	0	23:30	19,505	0

Tabulka G.2: Data získaná během uživatelského testu – 9. květen 2019



Denní doba	Teplota [°C]	Aktivita	Denní doba	Teplota [°C]	Aktivita
0:00	19,605	0	12:00	20,129	1
0:30	19,424	0	12:30	22,990	0
1:00	19,767	0	13:00	22,426	0
1:30	19,344	0	13:30	21,761	0
2:00	19,122	0	14:00	21,096	0
2:30	19,102	0	14:30	20,653	0
3:00	19,122	0	15:00	20,532	0
3:30	18,860	0	15:30	20,311	0
4:00	18,679	0	16:00	20,431	0
4:30	18,981	0	16:30	20,049	0
5:00	18,679	0	17:00	20,069	0
5:30	21,116	1	17:30	19,867	0
6:00	23,272	1	18:00	20,028	0
6:30	23,050	0	18:30	20,028	0
7:00	21,600	0	19:00	19,948	0
7:30	21,197	0	19:30	19,767	0
8:00	20,975	0	20:00	19,908	0
8:30	20,633	0	20:30	19,626	0
9:00	20,472	0	21:00	22,124	1
9:30	20,311	0	21:30	22,950	0
10:00	19,988	0	22:00	21,902	0
10:30	20,089	0	22:30	21,137	0
11:00	19,686	0	23:00	20,794	0
11:30	19,726	0	23:30	20,431	0

Tabulka G.3: Data získaná během uživatelského testu – 10. květen 2019