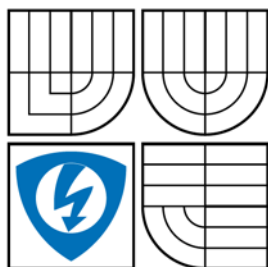


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ
ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY



FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

DECENTRALIZOVANÝ SYSTÉM ŘÍZENÍ VYTÁPĚNÍ BUDOV

DCS FOR BUILDING CONTROL

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

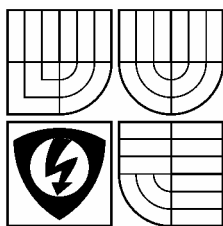
AUTOR PRÁCE
AUTHOR

Bc. PETR KOPEČEK

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. ZDENĚK BRADÁČ, Ph.D.

BRNO 2008



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav automatizace a měřicí techniky

Diplomová práce

magisterský navazující studijní obor

Kybernetika, automatizace a měření

Student: Kopeček Petr, Bc.

ID: 89281

Ročník: 2

Akademický rok: 2007/08

NÁZEV TÉMATU:

Decentralizovaný systém řízení vytápění budov

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s problematikou řízení otopných systémů v budovách. Navrhněte a realizujte decentralizovaný systém řízení tepelné pohody v budovách. Jako centrální jednotku zvolte systém PC.

Navrhněte a realizujte akční členy a senzory. Systém realizujte jako flexibilní s možností upgrade SW v jednotkách. Systém propojte proprietárním komunikačním systémem, navrhněte komunikační protokol.

Jednotky vyřešte po hardwarové stránce a realizujte. Sestavte programové vybavení.

- Realizujte HW moduly, osadte elektronickými součástkami a oživte.
- Vytvořte programové vybavení modulů a ověřte jejich funkčnost.
- Celý systém okomentujte a charakterizujte vlastnosti.
- Sepište diplomovou práci.

DOPORUČENÁ LITERATURA:

Dle doporučení vedoucího práce.

Termín zadání: 3.12.2007

Termín odevzdání: 26.5.2008

Vedoucí projektu: Ing. Zdeněk Bradáč, Ph.D.

prof. Ing. Pavel Jura, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

LICENČNÍ SMLOUVA

POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami:

1. Pan/paní

Jméno a příjmení: Bc. Petr Kopeček

Bytem: Zašovice 15

Narozen/a (datum a místo): 4.10.1983, Třebíč

(dále jen „autor“)

a

2. Vysoké učení technické v Brně

Fakulta elektrotechniky a komunikačních technologií

se sídlem Údolní 244/53, 602 00, Brno

jejímž jménem jedná na základě písemného pověření děkanem fakulty:

doc. Ing. Václav Jirsík, CSc.

(dále jen „nabyvatel“)

Čl. 1

Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):

- ☐ disertační práce
- ☒ diplomová práce
- ☐ bakalářská práce
- ☐ jiná práce, jejíž druh je specifikován jako

.....

(dále jen VŠKP nebo dílo)

Název VŠKP: Decentralizovaný systém řízení vytápění budov

Vedoucí/ školitel VŠKP: Ing. Zdeněk Bradáč, Ph.D.

Ústav: Ústav automatizace a měřicí techniky

Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v*:

- | | | |
|--|---|-------------------|
| ☒ tištěné formě | – | počet exemplářů 1 |
| ☒ elektronické formě | – | počet exemplářů 1 |

* hodící se zaškrtněte

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2

Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti
 - ☒ ihned po uzavření této smlouvy
 - ☐ 1 rok po uzavření této smlouvy
 - ☐ 3 roky po uzavření této smlouvy
 - ☐ 5 let po uzavření této smlouvy
 - ☐ 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3

Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne:

.....
Nabyvatel

.....
Autor

Decentralizovaný systém řízení vytápění budov

Diplomová práce

Studijní obor: Kybernetika, automatizace a měření
Student: Bc. Petr Kopeček
Vedoucí projektu: Ing. Zdeněk Bradáč, Ph.D.

Abstrakt:

Předmětem této diplomové práce je návrh systému regulujícího vytápění budov. Cílem bylo vytvořit univerzální systém, který může být přizpůsoben konkrétním požadavkům při současné minimalizaci nákladů. Proto byla zvolena modulární koncepce umožňující sestavení systému v potřebné konfiguraci. Centrální jednotkou je standardní PC či průmyslový počítač z důvodu vysokého výpočetního výkonu, snadné vizualizace a připojení k síti Internet. Řídicí počítač a jednotlivé moduly jsou propojeny komunikační sběrnici RS485. Ta byla zvolena z důvodu levné kabeláže a odolnosti vůči rušení.

Jako regulační nástroj byl navržen PSD regulátor, který byl doplněn o další funkce umožňující úpravu signálu. Pro složitější případy mohou být tyto regulátory řazeny paralelně i sériově.

Dalším požadavkem bylo ukládání naměřených dat, což poskytuje možnost zpětné kontroly funkce systému a optimalizace jeho parametrů.

Systém byl navržen a testován v průmyslovém provozu a byla ověřena jeho plná funkčnost.

Klíčová slova:

RS485, regulační systém vytápění, decentralizovaný systém, modulární systém, měření teploty, regulace teploty, systém se vzdálenou správou, ukládání naměřených dat

DCS for building control

Thesis

Specialisation of study: Cybernetics, Control and Measurement
Student: Bc. Petr Kopeček
Supervisor: Ing. Zdeněk Bradáč, Ph.D.

Abstract:

This thesis deals with the building heating control system. The main aim was to develop a general system, which can be adapted according to the individual requirements and which should minimize costs too. This is why the modular conception has been chosen. As a central unit there is used a standard PC or an industrial computer because of the high computing power, easy visualization and connection to the Internet network. Control computer and modules are connected via communication bus RS485. This bus has been chosen because of the cheap wiring harness and high noise immunity.

As regulation tool has been designed PSD regulator, which has been extended by another signal conditioning functions. In the case of more complex applications, these regulators can be placed parallel or serial.

The other requirement was storage of measured data that offer possibility to the system function back control and optimalization of its parameters.

The system has been developed, tested in the industrial application and its full functionality has been verified.

Key words:

RS485, heating control system, decentralized control system, modular system, temperature measurement, temperature regulation, remote control system, measured data logging

Bibliografická citace:

KOPEČEK, P. *Decentralizovaný systém řízení vytápění budov*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 84 s. Vedoucí diplomové práce Ing. Zdeněk Bradáč, Ph.D.

P r o h l á š e n í

„Prohlašuji, že svou diplomovou práci na téma "Decentralizovaný systém řízení vytápění budov" jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.“

V Brně dne :

Podpis:

Poděkování

Děkuji vedoucímu diplomové práce Ing. Zdeňku Bradáčovi, Ph.D. za metodickou pomoc a odborné vedení a majiteli firmy CODEP s.r.o. Ing. Janu Hrejsovi za cenné rady při zpracování diplomové práce a poskytnutí výrobní technologie.

V Brně dne 19. 5. 2008

Obsah

1. ÚVOD	5
2. KONCEPCE SYSTÉMU	6
2.1 Vlastnosti navrženého systému	6
2.1.1 Silová jednotka	6
2.1.2 Zobrazovací jednotka	7
2.2 Vlastnosti konkurenčních systémů.....	7
2.2.1 Regulus IR09 [12]	7
2.2.2 Elesta [13].....	8
2.2.3 XL5000 [14]	9
2.2.4 Xcomfort [15],[16]	9
3. HW REALIZACE	10
3.1 Silová jednotka.....	10
3.1.1 Silové výstupy	10
3.1.2 Digitální vstupy	11
3.1.3 Teplotní vstupy	11
3.1.4 Analogové vstupy	12
3.1.5 Analogové výstupy	12
3.1.6 Univerzální vstupy a výstupy	13
3.1.7 Napájecí zdroj.....	14
3.1.8 Komunikační a programovací rozhraní	15
3.2 Zobrazovací jednotka.....	15
3.2.1 Displej.....	16
3.2.2 Tlačítkové vstupy	16
3.2.3 Teplotní vstup	16
3.2.4 Univerzální vstup.....	17
3.2.5 Analogový vstup.....	17
3.2.6 Napájecí zdroj.....	17
3.2.7 Komunikační a programovací rozhraní	17
3.3 Řídící PC	18
3.4 Akční členy	18

4. SW REALIZACE.....	20
4.1 Komunikační protokol	20
4.1.1 Formát paketu	20
4.1.2 Adresový prostor systému	22
4.1.3 Komunikační událost.....	25
4.2 Softwarová podoba systému	26
4.2.1 Řídicí počítač	26
4.3 Režimy chodu	29
4.3.1 Servisní režim	29
4.3.2 Aplikační režim	33
4.4 Regulační software.....	34
4.4.1 Regulační smyčka.....	34
4.4.2 Žádané hodnoty a režimy úspory energie	40
4.4.3 Tabulka žádaných hodnot v bytových jednotkách a řízení režimů.....	41
4.4.4 Tabulka konstant a propojovacích proměnných	44
4.4.5 Vizualizační okno, hlavní okno aplikace.....	47
4.4.6 Ukládání naměřených a vypočtených hodnot.....	49
4.4.7 Bezpečnostní systém.....	52
4.4.8 Uživatelský a administrátorský mód	57
4.4.9 Inicializace a spouštění systému	58
4.4.10 Okno nastavení systému	59
5. NASTAVENÍ REGULÁTORŮ	61
5.1 Diskrétní akční zásahy	61
5.2 Poruchy a omezení	62
5.3 Adaptivita a identifikace	63
6. ZÁVĚR.....	64
7. SEZNAM LITERATURY	66
8. SEZNAM ZKRATEK.....	68
9. PŘÍLOHY	69
9.1 Schémata	70
9.1.1 Zobrazovací jednotka	70

9.1.2 Silová jednotka	71
9.2 Desky plošných spojů	72
9.2.1 Zobrazovací jednotka	72
9.2.2 Silová jednotka	74
9.3 Funkce třídy cSystem.....	76

Seznam obrázků

Obrázek 4.1: Formát paketu.....	21
Obrázek 4.2: Formát virtuální adresy kanálu.....	23
Obrázek 4.3: Servisní okno systému.....	30
Obrázek 4.4: Nastavení regulační smyčky.....	36
Obrázek 4.5: Tabulka žádaných hodnot v bytových jednotkách	42
Obrázek 4.6: Tabulka konstant a propojovacích proměnných.....	45
Obrázek 4.7: Okno vizualizace procesu.....	48
Obrázek 4.8: Okno interpretace dat	51
Obrázek 4.9: Nastavení informačního emailu.....	54
Obrázek 4.10: Okno spuštění módu administrátora.....	58
Obrázek 4.11: Okno nastavení systému	59
Obrázek 9.1: Schéma zobrazovací jednotky	70
Obrázek 9.2: Schéma silové jednotky	71
Obrázek 9.3: Zobrazovací jednotka - spoje horní vrstva	72
Obrázek 9.4: Zobrazovací jednotka - spoje spodní vrstva	72
Obrázek 9.5: Zobrazovací jednotka - servisní potisk horní vrstva.....	73
Obrázek 9.6: Zobrazovací jednotka - servisní potisk spodní vrstva	73
Obrázek 9.7: Senzor	73
Obrázek 9.8: Silová jednotka - spoje horní vrstva	74
Obrázek 9.9: Silová jednotka - spoje spodní vrstva.....	74
Obrázek 9.10: Silová jednotka - servisní potisk horní vrstva	75
Obrázek 9.11: Silová jednotka - servisní potisk spodní vrstva	75

1. ÚVOD

Tepelná pohoda je jedním z hlavních předpokladů pro to, aby byl člověk schopen úspěšně pracovat či plnohodnotně relaxovat. Zajištění tepelné pohody proto patří k základním úlohám regulace. Z tohoto důvodu již bylo vyvinuto velké množství systémů regulace teploty. Pravděpodobně tedy nelze očekávat nějaký převratný průlom v této oblasti. Důvodem pro vývoj nového systému regulace vytápění může být integrace nových vlastností, jako například kombinace se systémem řízení přístupu nebo docházky, popřípadě se zabezpečovacím systémem. Budovám vybaveným těmito systémy se říká inteligentní budovy. Dalším důvodem je minimalizace ceny nabízeného zařízení.

Prvotní ideou této práce bylo vytvořit systém vytápění budov, který bude splňovat pokud možno oba tyto požadavky, bude co nejvíce univerzální, levný a dostupný. Zachování těchto vlastností pro mě bylo po celou dobu vývoje prioritou. Univerzálnosti jsem se snažil dosáhnout dvěma způsoby. Za prvé to byla naprostá volnost při nastavování regulačních smyček, kdy vstupními i výstupními veličinami regulátorů mohou být kterékoliv veličiny v systému. Za druhé je to otevřenost vůči implementaci jakéhokoliv dalšího systému. Za účelem minimalizace ceny byl systém navržen jako modulární. Konfigurace systému tedy může být přizpůsobena konkrétním potřebám uživatele.

Systém byl vyvíjen především pro průmyslové budovy střední velikosti a na budovách tohoto typu byl také testován.

2. KONCEPCE SYSTÉMU

2.1 VLASTNOSTI NAVRŽENÉHO SYSTÉMU

Jak již bylo zmíněno v úvodu, byla zvolena modulární koncepce decentralizovaného systému. V základní verzi systému jsou k dispozici dva moduly: silová jednotka a zobrazovací jednotka. Konfigurace systému je plně v rukou konstruktéra, který může volit počty, druh i zapojení jednotlivých modulů. Jednotlivé vstupní a výstupní kanály na modulech jsou na sobě navzájem nezávislé a může jich být použit libovolný počet až do omezení maximálním počtem modulů.

Komunikační sběrnici byl ze dvou důvodů zvolen standard RS485. Za prvé to byly vhodné vlastnosti této sběrnice pro průmyslové prostředí a za druhé zpětná kompatibilita s předchozími systémy. Je použita dvou vodičová verze této sběrnice. Systém je napájen střídavým napětím o velikosti 24 V, které je rozváděno společně s datovou sběrnicí jedním čtyřvodičovým kabelem. Předpokládá se, že systém bude v budoucnu doplněn bránou RS485/Ethernet, aby do něj mohly být zakomponovány moduly komunikující po této sběrnicí.

2.1.1 Silová jednotka

Silová jednotka je určena k ovládání primárních akčních členů:

- Spínání topných spirál
- Spínání řídicích signálů plynových kotlů
- Spínání ventilátorů
- Spínání čerpadel
- Nastavování polohy škrticích klapek
- Nastavování polohy směšovacích ventilů

Dále ke sběru dat z regulovaného procesu slouží:

- Teplotní vstupy
- Digitální vstupy
- Analogové vstupy
- Univerzální datové vstupy

Maximální počet silových jednotek je omezen hardwarově celkovým maximálním počtem všech zařízení v síti 256 bez použití opakovače nebo počtem 256 zařízení v úsecích mezi opakovači. Softwarově je potom omezen velikostí adresovacího prostoru. V základní verzi je počet adres pro silové jednotky 504. Nepředpokládá se ale plné využití adresového prostoru.

2.1.2 Zobrazovací jednotka

Zobrazovací jednotka tvoří hlavní rozhraní mezi uživatelem a systémem. K tomuto účelu slouží:

- LCD displej
- Tlačítka

Pro sběr dat z regulovaného procesu je implementován:

- Teplotní vstup
- Univerzální vstup

Maximální počet zobrazovacích jednotek je omezen hardwarově, stejně jako v případě silové jednotky, celkovým maximálním počtem všech zařízení v síti na 256 bez použití opakovače nebo počtem 256 zařízení v úsecích mezi opakovači. Softwarově je potom omezen velikostí adresovacího prostoru, který čítá v základní verzi 512 adres. Ani zde se ale nepředpokládá plné využití adresového prostoru.

2.2 VLASTNOSTI KONKURENČNÍCH SYSTÉMŮ

2.2.1 Regulus IR09 [12]

Regulus IR09 je regulátor pro regulaci velkých topných systémů s více zdroji tepla a s více topnými okruhy. Umožňuje připojení všech čidel pomocí jedné dvouvodičové sběrnice. Akční členy se musí připojovat přímo k řídící jednotce, přičemž se jedná o centralizovaný systém. Tento regulátor umožňuje řízení elektrokotlů, kotlů na plyn i na tuhá paliva. Vlastnosti systému citované z [www stránek výrobce](#) shrnuje následující seznam:

- postupné spínání až 4 zdrojů dle zadaných podmínek

- 4 směřované topné okruhy s volbou regulace podle venkovní nebo vnitřní teploty (nastavitelný vliv)
- ohřev teplé užitkové vody
- okruh slunečních kolektorů s přepínáním do několika zón (například TUV a bazén nebo TUV a přitápění)
- komunikace s PC nebo dalšími systémy
- datová teplotní čidla připojená společnou sběrnici (paralelní připojení všech teplotních čidel)
- snadná konfigurace parametrů z klávesnice regulátoru
- veškeré regulační závislosti jsou naprogramovány a uloženy v paměti, servisní technik zadává pouze požadované teploty, časy, topné křivky
- model pro kotle na tuhá paliva řídí minimální teplotu vratné vody (dva samostatné okruhy).

2.2.2 Elesta [13]

Regulátory firmy Elesta lze použít v široké škále aplikací. Regulátor průběžně vyhodnocuje naměřené hodnoty, určuje z nich parametry budovy a podle nich vypočítává optimální topný režim. Tyto regulátory se vyrábí v několika řadách s různými možnostmi nastavení. Základní typy uvádí následující seznam (citace ze stránek distributora):

- DOMOTESTA RDO 100 - jednoduché nastavování pomocí potenciometrů. Ovládání kotle, směšovacího ventilu a přípravy TUV (dle typu). Volba typu hodin - analogové denní, týdenní nebo digitální. Protimrazová ochrana celého topného systému. Možnost použití pokojového ovládání.
- DOMOTESTA RDO 200 - komfortní obsluha pomocí klávesnice a přehledného displeje. Ovládání kotle, směšovacího ventilu a přípravy TUV i s cirkulací dle časového programu. Ochrana TUV proti bakteriím Legionella. Protimrazová ochrana celého topného systému. Možnost použití pokojového ovládání „prázdninové“ funkce umožňuje snadný přechod do útlumového režimu v době, kdy jste např. na dovolené.

- DOMOTESTA RDO 300 - možnost propojování po sběrnici do složitých regulačních systémů, zadávání programů digitálně i z počítače. Řízení až čtyř tepelných zdrojů, sedm směřovaných okruhů, ohřev TUV v časovém režimu. Každý kotel v kaskádě může mít vlastní (samostatnou) regulaci teploty vratné vody.

2.2.3 XL5000 [14]

Excel 5000 je modulární regulační systém firmy Honeywell. Jedná se o rozsáhlý systém s možností propojení na ostatní systémy inteligentní budovy. Systém využívá pro vzájemné propojení modulů i pro připojení k nadřazenému systému komunikační sběrnice C-BUS, LON, M-BUS a popř. modemové, ISDN nebo GSM připojení. Vizualizace regulovaného procesu je realizována nástrojem Excel CARE pracujícím v prostředí MS Windows. V tomto systému mohou být použity typy regulátorů uvedené v následující tabulce (citace z www stránek výrobce):

- Excel 10 – předkonfigurované regulátory vhodné pro regulaci prostředí v místnostech.
- Excel 50 – volně programovatelné nebo konfigurovatelné regulátory pro menší aplikace s širokými komunikačními možnostmi.
- Excel 100 – volně programovatelné regulátory pro menší a střední aplikace.
- Excel 500 – volně programovatelné modulární regulátory pro velké a náročné aplikace s širokými komunikačními možnostmi.

2.2.4 Xcomfort [15],[16]

Xcomfort je systém pro inteligentní budovy vyráběný firmou Moeller. Nabízí regulaci vytápění i veškeré ostatní funkce inteligentních budov jako například ovládání osvětlení, ovládání žaluzií, příjezdové brány atd. Tento systém se vyrábí ve dvou variantách: Sběrníkový systém Nikobus a bezdrátový systém RF Xcomfort. Systém nabízí možnost použití vizualizační jednotky a propojení s dalšími systémy, detailní možnosti ale výrobce neuvádí.

3. HW REALIZACE

Schémata zapojení i desky plošných spojů byly navrženy v programu OrCAD 10. Rady a návody k používání programu byly čerpány z [1].

Při návrhu desek plošných spojů jsem postupoval tak, aby byla výroba a především osazování co nejlevnější. Desky jsou navrženy ve 4. třídě přesnosti s maximálním využitím SMD součástek. Z důvodu jednoduchosti a ekonomičnosti strojového osazování je použito co nejméně různých hodnot a typů součástek. Kvůli úspoře místa byla zvolena velikost SMD součástek 0603.

3.1 SILOVÁ JEDNOTKA

Výkres viz příloha 9.1.2.

Silová jednotka slouží ke spínání akčních členů v systému a ke sběru dat ze systému. Jednotlivá rozhraní silové jednotky jsou podrobně popsána v následujících kapitolách.

Jádrem silové jednotky je osmibitový mikroprocesor ATmega8 z řady Atmel AVR. Tento procesor řídí veškerou činnost jednotky. Je taktován externím krystalovým oscilátorem na frekvenci 7,3728 MHz a je doplněn blokovacími kondenzátory a LC filtrem referenčního napětí podle doporučení v katalogovém listě [2].

Deska plošného spoje silové jednotky je navržena pro umístění do krabice Hensel DP9220. Tato krabice poskytuje krytí IP54, takže se hodí pro použití v průmyslovém prostředí.

Z hlediska bezpečnosti jsou všechny výstupy modulu konstruovány tak, že v případě výpadku napájení jsou rozepnuté či neaktivní.

3.1.1 Silové výstupy

Viz příloha 9.1.2 A4, A5, B4, B5, C4, C5

Silová jednotka obsahuje tři silové spínací výstupy, přičemž každý je realizován triakem BT138 [6] a může spínat napětí do 250 V a proud do 4 A. Buzení výkonového triaku a galvanické oddělení od vysokonapěťové části zajišťuje

optotriak MOC3043 [5]. Tyto výstupy mohou pracovat jako nezávislé spínače i jako jeden kompaktní spínač s hardwarovým blokováním nežádoucího sepnutí. Volba režimu se provádí přepnutím spínačů 1 a 2 přepínače SW1. Hardwarové blokování funguje tak, že pokud dojde k sepnutí více výstupů současně a hardwarové blokování je aktivní, bude sepnut pouze spínač s vyšší prioritou. Nejvyšší prioritu má spínač S1 a nejnižší S3, přičemž hardwarové blokování lze jednotlivě nastavit u spínačů S2 a S3. Každý ze spínačů je chráněn proti nadproudu tavnou SMD pojistkou o jmenovitém proudu 4 A. Tyto pojistky jsou zpožděné (typ „T“), takže umožňují spínání zátěží induktivního charakteru, u kterých je proud v okamžiku bezprostředně po sepnutí několikanásobně vyšší než proud jmenovitý. Sepnutí spínače je indikováno LED diodou s odpovídajícím označením S1-S3.

3.1.2 Digitální vstupy

Viz příloha 9.1.2 B1

Mezi vstupní rozhraní silové jednotky patří čtyři digitální vstupy. Každý z těchto vstupů je galvanicky oddělen od dalších obvodů modulu pomocí optočlenu PC357 [7]. Ten může být spínán napětím 5-24 V a má na vstupu ochrannou Schottkyho diodu proti přepólování. Na vstupní svorkovnici jsou vyvedeny anoda i katoda optočlenu, takže ten může být spínán externím napětím. Na svorkovnici je dále vyvedeno napájecí napětí 5 V pro případ, že externí napětí není k dispozici a na vstupu je zapojen pouze spínací prvek. Způsob spínání lze nastavit jednotlivě pro každý vstup pomocí přepínače SW2. Sepnutím odpovídajícího spínače se spojí katoda optočlenu se zemí. Po jeho rozepnutí lze vstup používat jako bezpotenciálový opticky oddělený. Logická hodnota každého digitálního vstupu je indikována patřičnou LED diodou (DI1-DI4), přičemž svítící dioda indikuje sepnutý spínač.

3.1.3 Teplotní vstupy

Viz příloha 9.1.2 A3

Teplotní vstupy tvoří spolu se silovými výstupy hlavní rozhraní silové jednotky, protože umožňují základní úlohu navrženého systému – regulaci teploty. Na každé desce jsou situovány tři konektory pro připojení teplotního senzoru,

příčemž je plně v kompetenci uživatele, kolik jich využije. Jako teplotní senzory jsou použity obvody Dallas DS18S20 nebo DS18B20 [4]. Rozlišitelnost použitého teploměru je devět bitů s absolutní přesností $\pm 1^\circ\text{C}$ v případě DS18S20 nebo dvanáct bitů s absolutní přesností $\pm 1^\circ\text{C}$ v případě DS18B20. Obvody DS18x20 komunikují po sběrnici 1-wire, přičemž každý teploměr využívá svou sběrnici, aby v případě poruchy jednoho z nich nedošlo k zablokování sběrnice.

Senzor DS18x20 je vyráběn v pouzdře TO92, které je vhodné k umístění do jímky nebo trubky, která může být zasunuta do měřeného prostředí.

3.1.4 Analogové vstupy

Viz příloha 9.1.2 C3

Silová deska je vybavena dvěma analogovými vstupy. První je jednoúčelový s pevně nastaveným rozsahem 0-10 V, je vyveden do svorkovnice pro připojení servomotoru a zajišťuje jeho zpětnou vazbu. Druhý analogový vstup je univerzální s rozsahem 0-10 V. Za třetí analogový vstup by bylo možné považovat vnitřní měření napájecího napětí jednotky. Dalším analogovým vstupem může být univerzální IO-pin, záleží však na jeho softwarové konfiguraci (viz. kapitola 3.1.6). Všechny analogové vstupy jsou vybaveny analogovým filtrem s časovou konstantou 1 ms pro odfiltrování rušivých signálů a dvojicí ochranných diod proti přepětí a přepólování.

3.1.5 Analogové výstupy

Viz příloha 9.1.2 B3,C3

Silová jednotka je opatřena pouze datovými analogovými výstupy, které nemohou být použity k silovému spínání. Jeden analogový výstup je jednoúčelový s rozsahem 0-10 V, je vyveden do svorkovnice pro připojení servomotoru a zajišťuje nastavení žádaného natočení servomotoru. Další čtyři analogové výstupy jsou univerzální s rozsahem 0-5 V. Tyto výstupy musí být správně nakonfigurovány (viz. kapitola 3.1.6). Napětí na všech těchto výstupech je generováno pomocí pulzně šířkové modulace a RC filtru typu dolní propust'. Právě odpor ve filtru omezuje maximální proud, který je z výstupu možné odebírat. Vstupní odpor připojeného

obvodu by měl být nejméně v řádu 100 k Ω . V případě výstupu pro servomotor je napětí zesilováno dvakrát operačním zesilovačem TLV2371 typu rail-to-rail. Z důvodu špatné dostupnosti zesilovačů typu rail-to-rail jsou na desce plošného spoje tři místa pro osazení různých typů operačních zesilovačů rail-to-rail, aby bylo možné osadit typ, který je právě k dispozici.

3.1.6 Univerzální vstupy a výstupy

Viz příloha 9.1.2 A4, C3

Všechny uvedené vstupy a výstupy silové jednotky patřily mezi výchozí požadavky na tuto jednotku. Po jejich realizaci zůstaly na mikroprocesoru nepoužité IO piny. Tyto piny byly využity jako univerzální vstupy a výstupy. Jedná se o vývod označený ve schématu IO a tři vývody označené A-DO.

Vývod IO je zcela univerzální. Může být použit jako digitální i analogový vstup i výstup. Je vybaven vstupním i výstupním RC filtrem typu dolní propust' s časovou konstantou 1 ms, ochrannými diodami a přepínatelným vstupním děličem 1:1. Funkční mód se nastavuje softwarově z uživatelského rozhraní na řídicím počítači. Pouze dělicí poměr vstupního děliče napětí je nutné nastavit mechanicky. Toto má vliv pouze v případě, že je zvolen mód analogového vstupu (AI). Jeho rozsah může být 0-5V nebo 0-10V a volí se pomocí spínače č.4 na přepínači SW1. (Sepnutý spínač znamená rozsah 0-10V.)

Vývody A-DO mohou sloužit jako analogové nebo digitální výstupy. Primárně jsou však navrženy jako digitální a pro analogové použití by byla nutná úprava firmwaru v silové jednotce. Tyto vývody jsou opatřeny výstupním RC filtrem typu dolní propust' s časovou konstantou 1 ms. Jejich omezením je skutečnost, že tyto piny mikroprocesoru jsou společné s programovacím rozhraním, takže v okamžiku programování mohou na výstupech A-DO vzniknout nežádoucí stavy. To se však týká pouze přímého programování programátorem, které se u nainstalovaného systému nepředpokládá, nikoliv upgradu firmwaru pomocí bootloaderu.

3.1.7 Napájecí zdroj

Viz příloha 9.1.2 C1, C2, C4

Silová jednotka vyžaduje tři stejnosměrná napájecí napětí: 5 V pro veškerou logiku, 12 V pro zesilovač výstupního analogového signálu a 24 V pro napájení servomotoru. Jednotka může být napájena stejnosměrným nebo střídavým napětím, čehož je dosaženo zařazením Graetzova můstku na vstup zdroje. Amplituda střídavého napětí by se měla pohybovat v rozmezí 15-35V a totéž platí pro napájení stejnosměrným napětím. Před Graetzovým můstkem je dále zapojen transil, který zabrání průchodu napěťových špiček dále do zdroje. Dále jsou zde zapojeny tlumivky, které odstraňují vysokofrekvenční rušení.

Pro stabilizaci napájecích napětí 5 a 24 V jsou použity integrované spínané zdroje typu MC34063A [9]. Pro získání napájecího napětí 5V je zmíněný obvod použit v klasickém zapojení snižovacího DC/DC měniče. Situace je ale komplikovanější v případě napájecího napětí 24V pro servomotor. Protože se vstupní napětí měniče může pohybovat v rozmezí 15-35V a výstupní napětí má být 24V, nelze použít ani snižovací ani zvyšovací zapojení měniče. Zde by bylo vhodné použít zapojení nazývané „flyback“, které umožňuje takovou regulaci výstupního napětí. To však není ekonomicky výhodné, protože vyžaduje buď transformátor, nebo jiný typ měniče. V tomto případě bylo zvoleno sériové zařazení dvou obvodů MC34063A. První snižuje napětí na cca 20V a druhý toto napětí zvyšuje na potřebných 24V. Situaci neovlivní ani stav, kdy je vstupní napájecí napětí natolik malé, že na výstupu prvního měniče napětí nedosáhne 20V. V tom případě je první měnič plně otevřen a potřebné napětí i tak zajistí druhý zvyšující měnič. Protože odběr zesilovače výstupního analogového napětí je minimální, je pro stabilizaci 12 V použit stabilizátor 78L12.

Zapojení měničů i hodnoty podpůrných součástek vychází z doporučení výrobce uvedených v katalogovém listě [9]. Dále je napájecí napětí blokováno keramickými kondenzátory o kapacitě 100 nF.

Přítomnost napájecího napětí indikuje LED s popisem PWR.

3.1.8 Komunikační a programovací rozhraní

Viz příloha 9.1.2 A2, C3

Jako převodník mezi RS232 v TTL úrovních potřebných pro mikroprocesor a použitou sběrnici RS485 byl použit obvod sn65hvd3082 [10]. Tento obvod vyrábí firma Texas Instruments a svými vývody i funkcí je kompatibilní se staršími používanými obvody 75176, přičemž vykazuje ve všech kriteriích lepší vlastnosti. Protože byla zvolena pouze dvouvodičová sběrnice, převodník by měl být přepnut do vysílacího stavu pouze po nejkratší možnou dobu. Z důvodu snazšího zjištění stavu na sběrnici je přepnutí převodníku indikováno LED diodou s označením TxD. Svítící LED indikuje, že převodník je ve stavu vysílání a tedy blokuje sběrnici.

Programování mikroprocesoru se provádí pomocí technologie ISP skrze rozhraní SPI. Signálové vodiče MOSI, MISO a SCK tohoto rozhraní jsou spolu s napájením a signálem RST vyvedeny na programovací konektor. Programování tímto způsobem sice umožňuje naprogramovat mikroprocesor přímo na desce plošného spoje, ale vyžaduje připojení programátoru, což je velmi nevýhodné pro případný upgrade firmwaru. Proto se takto programuje pouze část firmwaru – bootloader – a konfigurační paměť mikroprocesoru a aplikační část firmwaru se programuje přes komunikační sběrnici z řídicího počítače.

3.2 ZOBRAZOVACÍ JEDNOTKA

Výkres viz příloha 9.1.1.

Zobrazovací jednotka tvoří společně s řídicím počítačem rozhraní mezi uživatelem a systémem. Na rozdíl od PC, ke kterému by měl přistupovat pouze administrátor, zobrazovací jednotku využívá běžný uživatel. Tato jednotka měří, zobrazuje a umožňuje nastavit teplotu v místnosti, kde je instalována. Dále je možné zobrazit jednu další teplotu. Zpravidla to je teplota vně objektu. (Uvádí se zde sice měření a nastavení teploty, ale z principu se může jednat o jakékoliv veličiny. Z podstaty systému se ale předpokládá právě teplota.)

Jádrem zobrazovací jednotky je 8 bitový mikroprocesor ATmega325 z řady Atmel AVR. Tento procesor řídí veškerou činnost jednotky a je taktován externím

krystalovým oscilátorem na frekvenci 1.843 MHz. Dále je doplněn blokovacími kondenzátory podle doporučení v katalogovém listě [3].

Zobrazovací jednotka je navržena pro montáž do rozvodné elektroinstalační krabice o minimální hloubce 15 mm. Jako čelní panel se použije upravený zaslepovací kryt krabice stejné typové řady jako ostatní prvky použité v budově.

3.2.1 Displej

Viz příloha 9.1.1 B4, B5

Pro zobrazení údajů je použit čtyřmístný 7-segmentový LCD displej. Tento displej není vybaven řadičem, a proto jsou všechny jeho segmenty buzeny přímo mikroprocesorem. Toto řešení sice vyžaduje procesor s více vývody, ale je velmi výhodné z hlediska spotřeby. Pro případ používání v noci je displej podsvětlen čtyřmi modrými LED diodami.

3.2.2 Tlačítkové vstupy

Viz příloha 9.1.1 C1

Ovládání zobrazovací jednotky zajišťují 3 tlačítka SW1 – SW3. Levé (SW1) slouží pro pohyb v menu směrem doleva a dekrementaci nastavených hodnot a prostřední (SW2) pro pohyb směrem doprava a inkrementaci nastavených hodnot. Právě tlačítko (SW3) je potvrzovací „OK“ tlačítko.

3.2.3 Teplotní vstup

Viz příloha 9.1.1 D5

Pro měření teploty je na zobrazovací jednotce umístěn teplotní vstup. Tvoří jej integrovaný teploměr DALLAS DS1631 [11]. Tento obvod sám provádí měření teploty a komunikuje s nadřazeným mikroprocesorem pomocí sběrnice I2C. Z důvodu minimalizace vlivu tepla vytvářeného ostatními součástkami a především zdrojem, je tento obvod umístěn na hliníkovém profilu, který zajišťuje přechod tepla z okolního prostředí. K desce zobrazovací jednotky připojen pohyblivým spojem. Ten se připojuje do konektoru J5.

Rozlišitelnost použitého teploměru je 12 bitů s absolutní přesností $\pm 0,5^{\circ}\text{C}$.

3.2.4 Univerzální vstup

Viz příloha 9.1.1 C5

Univerzální vstup zobrazovací jednotky tvoří sběrnice I2C vyvedená na konektor J4. Fyzicky se jedná o jiné vodiče než v případě teplotního vstupu, aby se v případě poruchy tyto vstupy neovlivňovaly. Prostřednictvím tohoto vstupu může být připojen jakýkoliv obvod komunikující po sběrnici I2C, například další teploměr. Primárně se ale předpokládá připojení čidla vlhkosti vzduchu a rosného bodu pro využití při regulaci klimatizace.

3.2.5 Analogový vstup

Viz příloha 9.1.1 A3

Stejně jako silová jednotka má i zobrazovací jednotka zabudované měření vlastního napájecího napětí.

3.2.6 Napájecí zdroj

Viz příloha 9.1.1 A1, A2

Zobrazovací jednotka je navržena pro napájecí napětí 5V. V případě, že by bylo nutné z důvodu zmenšení ztrátového výkonu a tím snížení ohřívání jednotky napájecí napětí snížit, je možné po výměně převodníku RS232 TTL – RS485 za jiný typ snížit napájení na 3,3V.

Pro získání stejnosměrného stabilizovaného napětí je použit obvod MC34063A ve snižujícím zapojení s hodnotami součástek doporučenými v katalogovém listě [9]. Dále je napájecí napětí blokováno keramickými kondenzátory o kapacitě 100 nF.

Přítomnost napájecího napětí indikuje LED s popisem PWR. Z důvodu snížení spotřeby lze tuto indikaci vypnout vyjmutím zkratovací propojky J7.

3.2.7 Komunikační a programovací rozhraní

Viz příloha 9.1.1 B1, D1

Jako převodník mezi sběrnici RS232 TTL a sběrnici RS485 byl použit stejně jako v případě silové jednotky obvod sn65hvd3082 [10]. Indikace stavu převodníku je zajištěna LED diodou s označením TxD. Svítící LED indikuje, že převodník je ve

stavu vysílání a tedy blokuje sběrnici. Z důvodu snížení spotřeby lze tuto indikaci vypnout vyjmutím zkratovací propojky J6. V případě nutnosti použití napájecího napětí 3,3V je nutné osadit převodník umožňující činnost s tímto napětím, např. obvod ADM3493 firmy Analog Devices.

Programování mikroprocesoru v zobrazovací jednotce se provádí stejně jako v případě silové jednotky technikou ISP skrz rozhraní SPI. Signálové vodiče MOSI, MISO a SCK tohoto rozhraní jsou spolu s napájením a signálem RST vyvedeny na programovací konektor. Ze stejných důvodů jako v případě silové jednotky je i zde implementován systém uploadu firmwaru pomocí bootloaderu, který aplikační a konfigurační část firmwaru naprogramuje přes komunikační sběrnici z řídicího počítače.

3.3 ŘÍDICÍ PC

Jako řídicí počítač byl použit typ VIA Eden 800 MHz. Jedná se o průmyslový počítač určený k zabudování do racku.

Pro komunikaci využívá sériový port s redukcí RS232/RS485 vyrobenou firmou CODEP s.r.o. Tato redukce převádí jeden port RS232 na dva RS485. Redukce umožňuje několik módů provozu. V tomto případě pracuje tak, že na oba kanály RS485 vysílá data přijatá z kanálu RS232 a data přijatá z obou kanálů RS485 logicky sčítá a odesílá na kanál RS232. Protože je použita dvouvodičová sběrnice, je nutné řídit směr toku dat. To zajišťuje signál RTS kanálu RS232.

3.4 AKČNÍ ČLENY

Ve stávající verzi se k výkonové desce připojují 3 typy akčních členů. Jedná se o topná tělesa a oběhová čerpadla spínaná triaky S1-S3 a servomotor. Ten je řízen analogovým výstupem přizpůsobeným konkrétnímu typu a slouží především k nastavování klapky vzduchotechniky a směšovacích ventilů topné vody. V budoucích instalacích se předpokládá využití typu LMC24-SR-F firmy Belimo [8].

V jedné ze stávajících instalací je z důvodu zpětné kompatibility použit starší typ servomotoru téhož výrobce, který ale nemá integrován snímač polohy. Ten musel být vyroben pomocí LED diod a fototranzistoru. Snímač polohy byl připojen na

univerzální výstupy A-DO, ve kterých musely být odstraněny výstupní RC filtry. Tato verze servomotoru je navíc napájena střídavým napětím 230 V, přičemž směr otáčení se volí připojením fáze na jeden ze dvou vodičů. K tomu byly využity spínače S1 a S2, které tak už nemohou být použity ke standardnímu spínání, protože nejsou řízeny jako výstupní kanály z PC. Jejich spínání je řízeno regulátorem naprogramovaným v řídicím mikroprocesoru modulu. To samozřejmě vyžaduje úpravu firmware daného modulu.

4. SW REALIZACE

Celková koncepce systému je striktně modulární. To platí i o software, který je navržen s co možná nejvyšší úrovní abstrakce. Jednotlivé moduly obsahují různé vstupy a výstupy. Tyto vstupy a výstupy jsou z hlediska uživatele oddělené a pohlíží se na ně jako na vstupní či výstupní „kanály“. Při návrhu regulačních algoritmů potom nezáleží na tom, na kterém modulu se daný kanál fyzicky nachází. Z hlediska uživatele je reprezentován svým názvem a z hlediska systému virtuální adresou. Při výběru kanálu uživatelem podle jeho názvu se uloží jeho virtuální adresa a v okamžiku použití tohoto kanálu se vypočítává virtuální adresa modulu z adresy kanálu. To se ale děje „na pozadí“ a uživateli to umožňuje pracovat s daným kanálem pouze podle jeho názvu, který mu přidělil při konfiguraci systému bez znalosti fyzického rozložení modulů.

4.1 KOMUNIKAČNÍ PROTOKOL

4.1.1 Formát paketu

Použitý komunikační protokol vychází z protokolu SDLC vyvinutého společností IBM. Formát datového paketu je znázorněn na obrázku Obrázek 4.1. Protože mikroprocesory použité na jednotlivých jednotkách systému jsou osmibitové, základní komunikační datová jednotka z pohledu linkové vrstvy je byte. Jednomu bytu také odpovídá jedno políčko v grafickém znázornění formátu paketu. Protože je ale osmibitová informace nedostačující, byl zvolen jako základní komunikační datová jednotka z pohledu aplikační vrstvy 16-ti bitový datový typ WORD. Kromě vyhrazených řídicích bytů (v obrázku zvýrazněných modrou barvou) jsou všechny informace přenášeny ve dvou bytech. Proto všechny údaje o délce dat budou dále uváděny jako počet dvoubytových slov, nebude-li explicitně uvedeno jinak.

Seznam vyhrazených řídicích bytů poskytuje Tabulka 4.1. Vyhrazené byty mají svůj jedinečný význam, s výjimkou prefixu se nemohou vyskytovat v adresové a datové části ani v CRC a slouží k synchronizaci vysílacího a přijímacího modulu.

Pokud je třeba odeslat data, ve kterých se vyskytuje kombinace odpovídající některému z vyhrazených bytů, je tento odesílaný byte nahrazen dvojicí „prefix“ a zástupný znak. Při dekódování se „prefixy“ převádí na příznaky a následný zástupný znak na odpovídající hodnotu.

Protože se předpokládá rozšiřování systému o další moduly, které by mohly vyžadovat velké množství dat, jsou v systému možné dva typy zpráv. Tyto zprávy jsou rozlišeny hlavičkou paketu. Jedná se o krátké zprávy, jež slouží ke komunikaci mezi řídicím počítačem a silovým, popř. ovládacím modulem, a dlouhé zprávy, jejichž využití zatím není pevně určeno a závisí na dalších modifikacích systému. Maximální délka prvního typu zpráv je 256 bytů dat, tedy 128 [WORD] kvůli omezené velikosti bufferů v datové RAM paměti použitých mikroprocesorů. Délka druhého typu zpráv je zatím omezena prakticky jen rozsahem proměnných o rozměru 16-ti bitů, tedy na 65 536 [WORD] dat. Rozdílná hlavička se používá z důvodu ochrany vstupních bufferů proti přetečení u zařízení, která nemohou přijímat dlouhé pakety. Tato zařízení paket s nepodporovanou hlavičkou dále nezpracovávají.

Hlavička paketu	Virtuální adresa bit 15-8	Virtuální adresa bit 7-0	Začátek datové části	Délka datové části Bit 15-8	Délka datové části Bit 7-0	Data 1 Bit 15-8	~
~	Data 1 Bit 7-0	...	Data n Bit 15-8	Data n Bit 7-0	CRC Bit 7-0	CRC Bit 15-8	Konec paketu

Obrázek 4.1: Formát paketu

CRC je odesíláno v opačném pořadí. Využívá se vlastnosti, že pokud se spočte CRC takto uspořádané zprávy včetně CRC ve zprávě, tak musí být v případě bezchybného přenosu CRC nově vzniklé zprávy nulové.

Význam	Kód (hexadecimálně)	Zástupný znak (hexadecimálně)
Začátek datové části paketu	0x7A	0x5A
Konec paketu	0x7B	0x5B
Hlavička dlouhého paketu	0x7C	0x5C
Prefix	0x7D	0x5D
Hlavička krátkého paketu	0x7E	0x5E

Tabulka 4.1: Seznam vyhrazených bytů komunikačního protokolu

4.1.2 Adresový prostor systému

V rámci systému se používají dva adresové prostory. Jedná se o jedinečná výrobní sériová čísla, podle nichž lze jednoznačně identifikovat modul, a virtuální adresu, kterou má každý ze vstupních či výstupních kanálů.

4.1.2.1 Výrobní sériová čísla

Sériové číslo je modulu přiřazeno při programování bootloaderu klasickým hardwarovým programátorem. Je zapsáno v programové flash paměti v sektoru bootloaderu a nelze jej softwarově měnit. Toto číslo je dvoubytové a je jedinečné v rámci všech vyrobených modulů. Ze sériového čísla lze také určit typ modulu, kterému patří. Seznam rozsahů sériových čísel a jejich význam shrnuje Tabulka 4.2.

Rozsah sériových čísel (hexadecimálně)	Typ modulu
0x0000	Identifikace broadcastové zprávy
0x0001 – 0x00FF	Vyhrazené pro servisní účely
0x0100 – 0x3FFF	Silový modul
0x4000 – 0x7FFF	Zobrazovací modul
0x8000 – 0xFFFF	Volné pro budoucí rozšíření

Tabulka 4.2: Seznam sériových čísel

4.1.2.2 Virtuální adresy

Virtuální adresa je jedinečná v rámci jedné instalace systému a je možné ji softwarově měnit. Toho lze využít při výměně modulu v nainstalovaném systému. Novému modulu s jiným sériovým číslem se přiřadí stejná virtuální adresa jako odebranému modulu a není nutné měnit další nastavení systému.

Uživatel může každému modulu určit virtuální adresu z rozsahu 1 – 2000. Prakticky je možné použít rozsah virtuálních adres až do hodnoty 2047 včetně, ale posledních 47 adres nelze standardně používat. Tyto adresy jsou vyhrazeny pro servisní a ladicí účely. Protože jsou všechny vstupy a výstupy jednotlivých modulů chápány jako oddělené kanály, mají své virtuální adresy. Tyto adresy už nejsou zadány uživatelem, ale jsou určeny na základě virtuální adresy modulu, kterou zadal uživatel. Virtuální adresa modulu se nachází v adresovém slově na pozici horních 11 bitů, adresa kanálu na pozici dolních 5 bitů (viz Obrázek 4.2). Relativní adresy jednotlivých kanálů vzhledem k adrese modulu silové jednotky jsou v tabulce Tabulka 4.3, zobrazovací jednotky potom v tabulce Tabulka 4.4. Absolutní adresa kanálu se určí sečtením virtuální adresy modulu a relativní adresy kanálu. Adresy 28-31 jsou vyhrazeny pro posílání servisních povelů.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Adresa modulu										Adresa kanálu					

Obrázek 4.2: Formát virtuální adresy kanálu

Adresa	Kanál	I/O
0	Virtuální adresa modulu	I+O
1	Měření napájecího napětí modulu	I
2	Silový spínač 1	O
3	Silový spínač 2	O
4	Silový spínač 3	O
5	Silový spínač 1+2+3	O
6	Digitální vstup 1	I

7	Digitální vstup 2	I
8	Digitální vstup 3	I
9	Digitální vstup 4	I
10	Digitální vstup 1+2+3+4	I
11	Zpětná vazba servomotoru	I
12	Žádaná poloha servomotoru	O
13	Analogový vstup	I
14	Univerzální vstup/výstup	I/O
15	Režim funkce univerzálního vstupu/výstupu	I
16	Digitální výstup 1	O
17	Digitální výstup 2	O
18	Digitální výstup 3	O
19	Digitální výstup 1+2+3	O
20	Teplotní vstup 1	I
21	Teplotní vstup 2	I
22	Teplotní vstup 3	I
23	Nastavení bezpečných hodnot a nastavení	O
24	Načtení bezpečných hodnot a nastavení	I
25		
26		
27		
28		
29	Povel k přepnutí do servisní části firmwaru	O
30	Povel k načtení verze firmwaru	I
31		

Tabulka 4.3: Virtuální adresy kanálů – silový modul

Adresa	Kanál	I/O
0	Virtuální adresa modulu	I+O
1	Měření napájecího napětí modulu	I
2	Hlavní teplotní vstup	I
3	Pomocný vstup (teplota, vlhkost)	I
4	Uživatelsky nastavená hodnota – standardní režim	I
5	Uživatelsky nastavená hodnota – úsporný režim	I
6	Hlavní zobrazovaná hodnota	O
7	Pomocná zobrazovaná hodnota	O
8		I
~	~	~
28		O
29	Povel k přepnutí do servisní části firmwaru	O
30	Povel k načtení verze firmwaru	I
31		

Tabulka 4.4: Virtuální adresy kanálů – zobrazovací modul

4.1.3 Komunikační událost

Zpracování přijatých dat v modulech se provádí pod přerušením. To je vyvoláno v okamžiku přijetí bytu, což by bylo možné považovat za komunikační událost. Za takovou událost se ale považuje až přijetí celého nepoškozeného paketu.

Komunikační událost probíhá následujícím způsobem: Jakmile modul přijme hlavičku odpovídající typu zpráv, který má přijímat, nastaví všechny proměnné linkové vrstvy na výchozí hodnoty a čeká na přijetí zbytku paketu. Po přijetí ukončovacího znaku se provede kontrola CRC součtu celého paketu a pokud byl paket přijat nepoškozen, provede se porovnání virtuální adresy modulu s virtuální adresou přijatou v paketu. Pokud adresy odpovídají, je paket určen pro daný modul, jenž jej dále zpracovává.

Za účelem dosažení co největší jednoduchosti komunikace je použita hierarchie master/slave. Řídící počítač je masterem, který zasílá jednotlivým

modulům zprávy a ty mu zpětně odesílají odpovědi. Konkrétní podoba je závislá na kanálu, tedy virtuální adrese kanálu, se kterým řídicí počítač komunikuje. V případě výstupního kanálu počítač zasílá na virtuální adresu kanálu modulu data, jaká má modul zapsat na výstup, a modul již zpětně neodpovídá. V případě čtení hodnoty ze vstupního kanálu řídicí počítač zašle na virtuální adresu vstupního kanálu žádost a modul odešle zpět paket s aktuální hodnotou vstupního kanálu. Zvláštní případ je komunikace s „kanálem“ s relativní virtuální adresou 0. V tomto případě řídicí počítač žádá o hodnoty všech vstupních kanálů a modul je odesílá zpět v jednom paketu.

4.2 SOFTWAREVÁ PODOBA SYSTÉMU

Jak bylo uvedeno na začátku kapitoly, byla zvolena stejná modularita softwaru jako hardwaru. Software byl vyvinut v prostředí Borland Builder v jazyce C++ s maximálním využitím informací a doporučení uvedených v [17]. Systém byl navržen tak, aby byl co nejméně závislý na použitém prostředí Borland C++ Builder a bylo možné jej použít opakovaně s minimálním množstvím úprav. Celý systém je navržen na základě objektově orientovaného programování, což se ukázalo jako velmi výhodné.

Předprogramované formuláře prostředí Borland poskytují pouze grafické rozhraní aplikace. V hlavním formuláři se nachází instance třídy cSystem, která systém reprezentuje.

Software pro moduly byl vytvořen v jazyce C. V souvislosti s vlastním programovacím jazykem byla použita publikace [18], k jeho použití na konkrétní mikrokontroléry potom publikace [19].

4.2.1 Řídicí počítač

4.2.1.1 Třída cSystem

Třída cSystem je z hlediska softwaru základem celého systému. Jedná se v podstatě o softwarový ekvivalent nainstalovaného hardwaru. Nadřazenému objektu poskytuje veškeré funkce, které je systém schopen vykonávat. Obsahuje instance podřazených tříd popsanych dále a volá jejich členské funkce.

Hlavním prvkem třídy je tabulka sériových čísel modulů nainstalovaných v systému. Při komunikaci s kanálem určeným virtuální adresou třída cSystem přesměruje komunikaci do některé z instancí tříd zastupujících jednotlivé moduly. Pomocí sériových čísel se určuje, který typ softwarového modulu má být volán. Rozsahy sériových čísel určující typ modulu jsou uvedeny v tabulce Tabulka 4.2.

Třída obsahuje dále pole instancí třídy cServoData a pole instancí třídy cRoomUnitData, které jsou stěžejní pro funkci systému, a několik dalších objektů. Funkce některých z nich je popsána v následujících kapitolách.

Výčet a popis funkcí třídy cSystem a tím i všech veřejně přístupných funkcí systému je uveden v příloze v kapitole 9.3.

4.2.1.2 Třída cRS485regtherm

Třída cRS485regtherm je softwarovým protějškem fyzické komunikační linky použité v systému a tvoří linkovou vrstvu ISO/OSI modelu. Momentálně se využívá sériového COM portu na straně řídicího počítače a sběrnice RS485 na straně modulů. Tato třída reprezentuje použitý převodník RS232/RS485 a umožňuje fyzicky komunikovat.

Poskytuje funkce k otevření, uzavření a zjištění stavu portu a pro odesílání a přijímání paketů. Zajišťuje dále výpočet kontrolního CRC součtu odesílaných paketů a kontrolu správnosti CRC přijatých paketů použitím funkcí ze skupiny „crc16“.

Vzhledem k použité objektové formě lze v případě potřeby v budoucnu jednoduše pouhou úpravou této třídy přejít na jiný typ portu na straně řídicího počítače (např. USB).

4.2.1.3 Skupina funkcí „crc16“

Pro zabezpečení přenášených dat proti chybám vzniklým během přenosu se používá 16-ti bitový kontrolní součet CRC s generujícím polynomem 0xA001. Funkce pro výpočet CRC jsou k dispozici v souboru crc16.c a lze použít dvě verze. V prvním případě se komunikační funkci předá buffer dat a jeho délka a funkce vrátí CRC součet tohoto bufferu. Druhá funkce přijímá jako parametr data a aktuální

hodnotu CRC a provádí pouze jeden výpočetní krok. V rámci systému je z důvodu organizace dat použita druhá „průběžná“ verze.

4.2.1.4 Třída cServoData

Třída cServoData reprezentuje silový modul v softwarové části systému. Obsahuje členské proměnné, které odpovídají vstupům a výstupům na silovém modulu. Tyto proměnné jsou navíc navrženy jako pole, ve kterých se uchovává historie hodnot na vstupech a výstupech. Velikost tohoto pole je u všech kanálů stejná a je určena jednou globální proměnnou, která umožňuje snadno změnit délku historie ukládání. Výstupní kanály jsou navíc reprezentovány dvourozměrným polem, kde je v jedné části uložena historie žádaných hodnot na výstupech a v druhé části historie hodnot, které na výstupu fakticky byly. To umožňuje snazší identifikaci místa výskytu chyby v regulačním algoritmu, komunikaci či v hardwaru modulu nebo připojeného zařízení.

Kromě historie hodnot kanálů je pro každý výstupní kanál ve třídě zahrnuta proměnná, ve které je uložena „bezpečná“ výstupní hodnota. Jedná se o hodnotu, ve které je výstup neaktivní. Tyto bezpečné hodnoty se zapisují do paměti modulu a v okamžiku, kdy modulu po stanovenou dobu nepříjde zpráva od řídicího PC, modul zapíše bezpečné hodnoty na výstup, aby nedošlo k nehodě v případě výpadku komunikace. Společně s „bezpečnými“ hodnotami se zapisují do paměti modulu nastavení módu univerzálního vstupně výstupního kanálu a nulové pozice servomotoru, které jsou ve třídě také uloženy v odpovídajících proměnných. „Bezpečné“ hodnoty jsou zapsány na výstupy modulu také v okamžiku jeho spuštění připojením napájecího napětí.

Pro každý kanál třída dále obsahuje proměnnou, ve které je uložen název tohoto kanálu. Ten je nastavitelný uživatelsky a právě jím je kanál pro uživatele v celém systému reprezentován.

Pro identifikaci svojí instance v rámci systému obsahuje třída dvě další proměnné: virtuální adresu a textový název modulu. Obě jsou zobrazeny v servisním okně systému a používají se hlavně při nastavování nebo rekonfiguraci systému.

Pro snadné zjištění funkce systému se ukládá i počet po sobě jdoucích neúspěšných pokusů o komunikaci. Pokud je systém spuštěn, zobrazuje se tato hodnota v poli „status“ v servisním okně systému a nuluje se v okamžiku úspěšné komunikační události.

4.2.1.5 Třída *cRoomUnitData*

Třída *cRoomUnitData* softwarově reprezentuje zobrazovací jednotku. Obdobně jako *cServoData* uchovává ve svých členských proměnných historii hodnot na vstupech a výstupech kanálů.

Pro identifikaci svojí instance v rámci systému obsahuje také proměnné, v nichž je uložena virtuální adresa a textový název modulu. Počítadlo „timeout“ má stejnou funkci jako v případě třídy *cServoData*, tedy určení počtu po sobě jdoucích neúspěšných komunikačních událostí.

Navíc obsahuje proměnnou, která v sobě nese informaci, zda je aktivní úsporný nebo standardní režim. Ta umožňuje indikaci a lokální změnu aktivního režimu zobrazovací jednotkou.

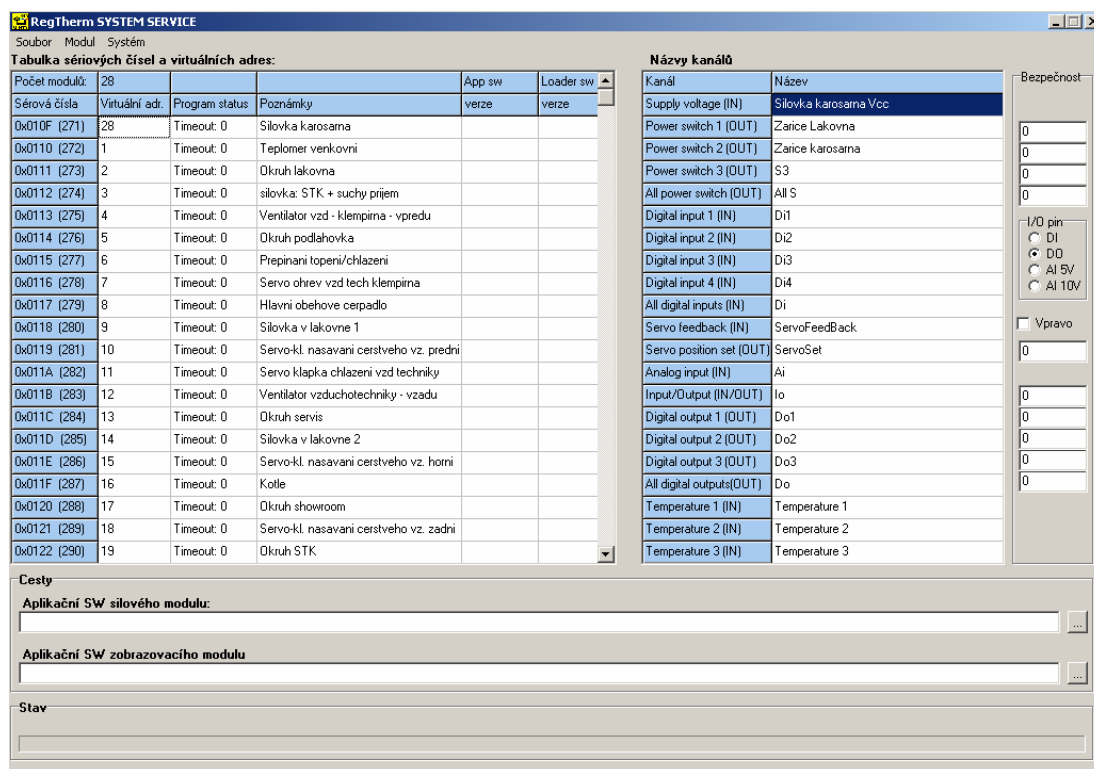
4.3 REŽIMY CHODU

4.3.1 Servisní režim

Servisní režim je implicitní pro každý modul po připojení systému k napájecímu napětí. V tomto režimu se provádí program uložený v bootovací části paměti programu. Při startu se zapíše na výstup „bezpečné“ hodnoty uložené v datové paměti EEPROM a režim umožňuje modulu komunikovat s řídicím počítačem přes globální servisní adresu 0x0000. Pomocí této adresy se odesílají broadcastové pakety a adresování jednotlivých modulů se potom provádí pomocí sériových čísel těchto modulů. Původní záměr byl používat virtuální adresy i v servisním módu, ale z důvodu omezené velikosti bootovací části paměti programu jej nebylo možné realizovat. Pro skenování sběrnice, které je vysvětleno dále, je nutné použít globální adresu a přidání algoritmu pro virtuální adresy by zabralo více paměti programu, než je k dispozici.

Ze servisního režimu lze do aplikačního přejít dvěma způsoby. Pokud je zvoleno automatické přepnutí, řídící počítač po spuštění systému a načtení konfigurací odešle příkaz pro přepnutí do aplikačního režimu, procesor v modulu provede všechny inicializační úkony, spočte CRC součet aplikační části paměti programu, porovná ho s uloženým a předá řízení aplikační části programu. Druhá možnost je přepnutí na příkaz uživatele. Přepnutí má stejný průběh, jen je zadáno ručně.

Grafické rozhraní používané k programování a nastavení modulu v servisním režimu znázorňuje Obrázek 4.3.



Obrázek 4.3: Servisní okno systému

V tomto okně je nutné provést všechna nastavení po instalaci nebo rekonfiguraci systému. Okno obsahuje standardní nabídku menu „Soubor“, které nabízí standardní položky „Uložit“ a „Exit“. Dále se zde nachází položky „Načíst kompletní konfiguraci“ a „Načíst pouze kanály“, jejichž význam bude vysvětlen

dále. Nabídka „Modul“ a „Systém“ poskytují de-facto stejné příkazy. Jediný rozdíl je v tom, že se příkaz zvolený v nabídce „Modul“ provede pouze ve vztahu ke zvolenému zařízení a příkaz z nabídky „Systém“ se aplikuje na všechna zařízení v systému.

V levé části servisního okna na obrázku Obrázek 4.3 se nachází seznam modulů instalovaných v systému. Tento seznam se vytvoří příkazem „Skenovat sběrnici“ z nabídky „Modul“ nebo „Systém“, který spustí algoritmus skenování sběrnice. Algoritmus automaticky detekuje všechny moduly připojené do systému, načte jejich sériová čísla a seřadí je do tabulky. Celkový počet nalezených modulů je uveden v záhlaví tabulky. Uživatel musí přiřadit jednotlivým modulům jedinečné virtuální adresy zapsáním do sloupce „Virtuální adr.“ a může zadat doplňující popis do sloupce „Poznámky“. Ve sloupci „Program status“ se vypisuje aktuální stav programu v modulu (servisní/aplikační) na příkaz „Zjisti stav“ z nabídky „Modul“ nebo „Systém“ nebo počet neúspěšných pokusů o spojení automaticky za běhu systému. Sloupce „App sw verze“ a „Loader sw verze“ slouží ke zjištění aktuální verze software nahraného v modulu. Vyčtení verze software se provádí příkazem „Zjisti verzi sw“.

Seznam modulů instalovaných v systému lze samozřejmě načíst ze souboru. K tomu se používá položka menu „Načíst kompletní konfiguraci“. Pomocí ní se načte veškeré servisní nastavení a systém může být spuštěn. To se také provádí v případě automatického načítání a spouštění systému. Pokud je ale nutná rekonfigurace systému či výměna některého modulu, provede se skenování sběrnice a poté se vybere položka „Načíst pouze kanály“. Tímto příkazem se otevře konfigurační soubor a z něj se přiřadí nastavení pouze těm modulům aktuálně nainstalovaným v systému, které mají odpovídající sériová čísla. Pokud byl detekován nový modul, zůstane jeho nastavení implicitní, a pokud byl načten ze souboru takový modul, jehož sériové číslo se v systému nenachází, je vypsán za detekovanými moduly. Tím se neztratí jeho nastavení a fyzicky není dále používán. Pokud došlo k výměně modulu, stačí označit starý modul, pomocí pravého tlačítka myši vyvolat pop-up menu, zvolit položku „Kopírovat“, poté označit nový modul a zvolit položku vložit. Tím se přenesou veškeré nastavení starého modulu na nový

a díky zachování virtuální adresy není nutné provádět jakékoliv další změny v systému. Pak už stačí označit starý modul a zvolit položku „Smazat“.

V pravé části okna na obrázku Obrázek 4.3 je seznam vstupních a výstupních kanálů modulu. Ve sloupci „Kanál“ je vypsán název kanálu a v závorce jeho vstupně výstupní charakter. Ve sloupci „Název“ jsou uživatelem definované názvy kanálů. Těmito názvy je potom kanál reprezentován uživateli v celém systému. Nastavení názvů se provádí separátně pro každý modul a seznam názvů kanálu se aktualizuje při výběru modulu v seznamu modulů v levé části okna.

Napravo od seznamu názvů kanálu jsou editační pole pro zadání „bezpečných“ hodnot výstupů kanálů. Svým umístěním odpovídají kanálu, vedle jehož názvu jsou uvedeny. Nachází se zde i nastavení univerzálního vstupně výstupního kanálu a volba nulové pozice servomotoru. Pro nastavení nebo načtení „bezpečných“ hodnot v modulech je nutné zvolit odpovídající příkaz z nabídky „Modul“.

Největší předností servisního módu je možnost upgrade aplikačního softwaru pomocí řídicího počítače bez použití klasického programátoru. To se provádí zadáním cesty k binárnímu souboru s aplikačním programem do odpovídajícího editačního pole ve skupině „Cesty“ a zadáním příkazu z nabídky. Pro nalezení souboru lze samozřejmě použít klasického otevíracího dialogu stisknutím tlačítka „...“ napravo od editačního pole. Po zadání cesty k souboru se zvolí příkaz „Naprogramovat vybraný modul“ z nabídky „Modul“ pro naprogramování jednoho modulu nebo „Naprogramovat vše“ z nabídky „Systém“ pro naprogramování všech modulů. Během tohoto programování se modulu zároveň přiřazuje virtuální adresa zadaná v seznamu nainstalovaných modulů. Při přenosu dat paměti programu z binárního souboru do paměti programu v modulu jsou data zabezpečena proti vzniku chyby dvojím CRC součtem. Na úrovni linkové vrstvy se standardně zabezpečují pakety. Aby se zabránilo případnému výpadku paketu nebo vzniku chyby, která by nebyla detekována CRC součtem na úrovni linkové vrstvy, je před začátkem přenosu spočten CRC součet binárního souboru. Aplikační paměť programu se naprogramuje a CRC součet binárního souboru je uložen v poslední stránce paměti programu, která se využívá k ukládání různých informací o modulu.

Před každým spuštěním aplikačního programu se kontroluje CRC součet paměti programu s hodnotou uloženou v poslední stránce aplikační programové paměti a pokud nesouhlasí, není aplikační program spuštěn.

V dolní části servisního okna je pole „Stav“, které zobrazuje stav právě prováděných operací. Používá se hlavně během programování modulů, kdy vypisuje sériové číslo modulu, který je právě programován, a graficky znázorňuje průběh programování.

Programování aplikační části paměti programu jednotlivých modulů lze provést pouze v servisním režimu. Naopak nastavení bezpečných hodnot se provádí v aplikační části programu.

4.3.2 Aplikační režim

4.3.2.1 Aplikační režim v modulech

Z hlediska modulů má aplikační režim pouze jednu variantu. Modul přijímá pakety jemu adresované a podle virtuální adresy kanálu provádí jednotlivé operace. Funkce modulu je omezena na sběr dat ze vstupních kanálů a jejich odeslání do řídicího počítače a příjem dat z řídicího počítače a jejich zápis na výstupní kanály. U vstupních dat je možné použít různé filtrace a předzpracování dat, ale to záleží na jednotlivých aplikacích a je to nutné provést změnou aplikačního software v modulu.

4.3.2.2 Aplikační režim v řídicím počítači

Aplikační režim v řídicím počítači lze dělit na dvě části podle toho, zda je spuštěna hlavní funkční smyčka. V okamžiku, kdy není spuštěna funkční smyčka, lze provádět různá nastavení systému. Jedná se například o nastavení v servisním okně, nastavení regulačních algoritmů, nastavení žádaných hodnot a režimů funkce. Jakmile je tato smyčka spuštěna, řídicí počítač zasílá výstupní data modulům a přijímá z nich data vstupní.

Hlavní funkční smyčka zajišťuje spouštění všech komunikačních a regulačních funkcí. Aby byla odezva uživatelského rozhraní aplikace rychlá, je hlavní funkční smyčka umístěna ve vlastním „funkčním“ vlákně.

V okamžiku vzorkování je nastaven příznak, který spustí tělo vlákna, v němž je smyčka umístěna. Nejprve se nastaví příznak o probíhající komunikaci a poté se otevře komunikační port. V případě úspěšného otevření portu se nejprve zapíše hodnoty vypočtené v předchozím regulačním kroku na výstup, poté se načtou aktuální vstupní a výstupní hodnoty, uzavře se komunikační port a spustí se regulační výpočetní algoritmy. Po dokončení výpočtů se vykreslí naměřené vstupní hodnoty do grafů ve vizualizačním okně. Pokud je nastaveno ukládání naměřených dat do souboru, jsou data uložena na konci funkční smyčky.

Opačné pořadí zápisu na výstupní kanály a čtení ze vstupních kanálů je zvoleno proto, aby nedocházelo k posunu dat v čase. Protože při čtení dat z modulu jsou zasílány i aktuální hodnoty výstupních kanálů, jsou tyto kanály nejprve nastaveny na požadovanou hodnotu, která je poté vyčtena a může být použita k ladění a kontrole.

4.4 REGULAČNÍ SOFTWARE

4.4.1 Regulační smyčka

Protože se předpokládá využití systému k regulaci jednoduchých soustav, vychází se z předpokladu, že se v těchto soustavách vyskytují pouze SISO systémy, nebo lze systém dekomponovat právě na SISO systémy. V případě nutnosti ale lze využít možnosti kombinace více SISO systémů za účelem vytvoření jednoduchého MISO systému.

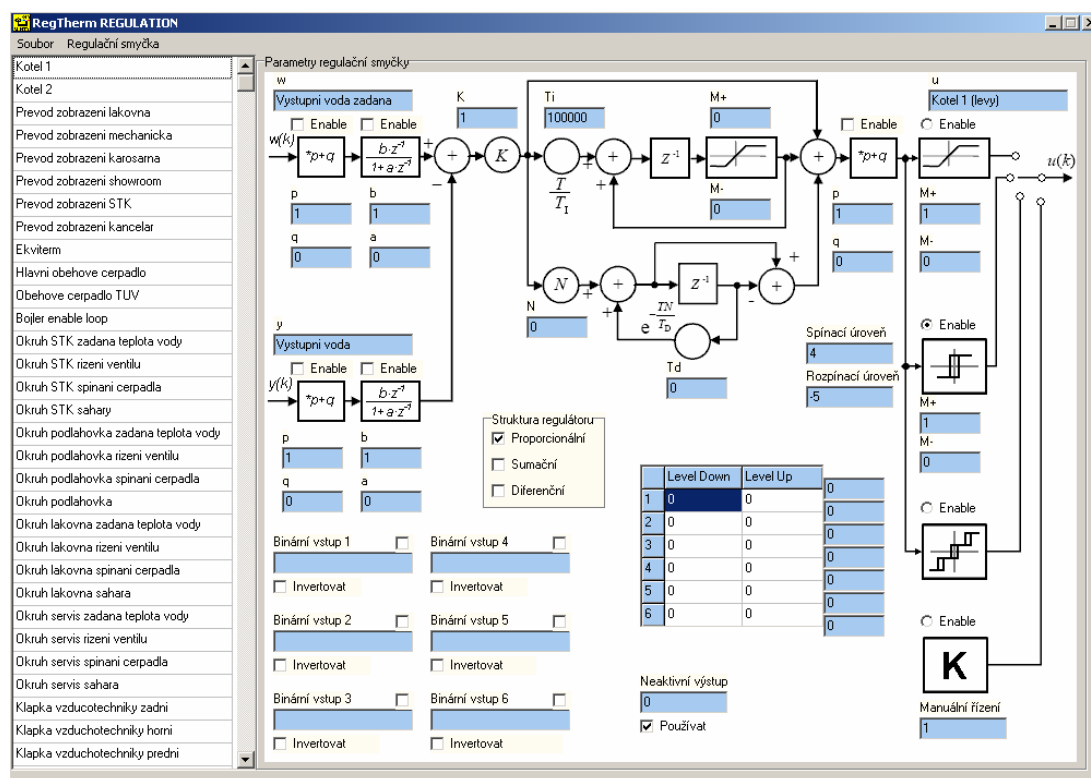
Právě z tohoto důvodu a s ohledem na maximální univerzálnost regulačního softwaru byl zvolen za základní univerzální regulační prvek PSD regulátor. Ten je reprezentován třídou `cRegulator`, která v sobě nese veškerá potřebná data. Byl implementován PSD regulátor s filtrací derivační složky, který je popsán v [21], a pro potřeby tohoto systému byl doplněn o několik funkcí popsaných dále.

4.4.1.1 Okno grafického rozhraní regulační smyčky

Uživatelské rozhraní pro nastavení regulační smyčky tvoří okno `RegTherm REGULATION` znázorněné na obrázku Obrázek 4.4. Pro snazší a co možná nejvíce intuitivní nastavování parametrů regulátoru byla zvolena forma blokového schématu

použitého regulátoru doplněného editačními poli k zadávání regulačních konstant umístěnými v blízkosti bloku, jehož parametr určují. Protože v systému může být v podstatě libovolné množství regulačních smyček a jejich počet lze měnit i v době běhu systému (myšleno kdykoliv po instalaci a konfiguraci, ovšem ne v okamžiku vykonávání regulačního algoritmu), obsahuje okno ve své levé části seznam všech regulačních smyček v systému a v pravé části parametry vybrané smyčky. Regulační smyčka se přidává vybráním položky „Přidat“ v menu „Regulační smyčka“. Tímto se vytvoří další řádek v seznamu smyček a je vyplněn přednastaveným textem. Uživatel si může zadat libovolný název smyčky, pod kterým se s ní bude dále pracovat a pod nímž bude uložena do souboru. Regulační smyčky lze samozřejmě jednoduše mazat pouhým označením v seznamu a vybráním položky „Smazat“ v menu „Regulační smyčka“. Dále zbývá nastavit parametry regulátoru. Volba nastavované regulační smyčky se provede jednoduchým kliknutím na její název v seznamu regulačních smyček. Při zadávání konstant stačí umístit kurzor do editačního pole a zadat požadovanou hodnotu. Systém kontroluje zadávané hodnoty a provádí korekci nesmyslných hodnot. Při zadávání vstupního nebo výstupního kanálu se po dvojitým kliknutí zobrazí seznam všech kanálů v systému a výběr kanálu z tohoto seznamu se provede opět dvojitým kliknutím.

Okno RegTherm REGULATION poskytuje standardní menu „Soubor“. V něm jsou obsaženy položky „Nový“, která smaže všechny aktuální regulační smyčky a vytvoří jednu novou s výchozím nastavením, dále položky „Otevřít“, „Uložit“ pro načtení a uložení do souboru a položka „Nastavit jako výchozí“. Ta provede uložení aktuálních regulačních smyček do souboru a tento soubor nastaví jako výchozí pro načtení při automatickém spouštění systému.



Obrázek 4.4: Nastavení regulační smyčky

4.4.1.2 Vstupy regulátoru

Do regulátoru standardně vstupují žádaná hodnota regulované veličiny a zpětná vazba od skutečné hodnoty výstupní veličiny a vystupuje z něj požadovaná velikost akčního zásahu.

Protože vstupy a výstupy regulátorů mohou být jakékoliv kanály, jejichž hodnoty mohou nabývat různých vzájemně nesouměrných hodnot, jsou z důvodu co největší univerzálnosti oba vstupy i výstup regulátoru doplněny normalizačním blokem s přenosem podle rovnice [4.1]. Protože se jedná pouze o lineární operace

$$y = p * x + q \quad [4.1]$$

násobení a součtu bez paměti, je použit zápis bez z-operátoru. Použití jednotlivých bloků lze povolit zatržením položky „Enable“ nad příslušným blokem a jeho parametry se nastavují v editačních polích „p“ a „q“ pod ním.

V případě přítomnosti šumu ve vstupních signálech lze do vstupních kanálů zařadit filtr prvního řádu s přenosem podle rovnice [4.2]. Použití filtru se zvolí stejně jako v případě normalizačního bloku zaškrtnutí políčka „Enable“ a jeho konstanty se zadávají do editačních polí „a“ a „b“.

$$F(z) = \frac{b * z^{-1}}{1 + a * z^{-1}} \quad [4.2]$$

4.4.1.3 PSD regulátor

Po úpravě vstupních signálů následuje rozdílový člen a za ním vlastní tělo regulátoru s operátorovým přenosem podle rovnice [4.3]. Za účelem zabránění

$$F_R(z) = K \left(1 + \frac{Tz^{-1}}{T_I(1 - z^{-1})} + N \frac{1 - z^{-1}}{1 - e^{-\frac{TN}{T_D}} z^{-1}} \right) \quad [4.3]$$

přeintegrování, tzv. wind-up jevu, je zde implementována i saturace integrační složky s uživatelsky nastavitelnými mezemi. Všechny konstanty regulátoru jsou uživatelsky nastavitelné pomocí odpovídajících editačních polí. Jedinou konstantně danou hodnotou je vzorkovací perioda. V současné verzi je nastavena na $T_s = 1$ min a nemůže být uživatelsky měněna, neboť její minimální hodnota je závislá především na cyklu sítě a dalších vlastnostech systému. V současných instalacích se jedná o vhodnou délku vzorkovací periody a v případě nutnosti její změny v některé budoucí instalaci systému bude záležet na konkrétních potřebách a požadavcích a nelze předem odhadnout řešení tohoto problému. Rovnice [4.3] popisuje standardní přenos PSD regulátoru v plné konfiguraci. V tomto systému mohou být použity i jednoduché regulátory P, S a D (i když D samostatně nemá význam, lze jej z principu použít) a jejich vzájemné kombinace. Použití jednotlivých složek se určí zatržením příslušného políčka ve skupině „Struktura regulátoru“.

4.4.1.4 Výstup regulátoru

Výstup vlastního PSD regulátoru je tvořen sumačním členem, který sčítá dílčí akční zásahy od jednotlivých složek regulátoru, následovaný výše popsáním normalizačním členem. Protože topná tělesa, motory čerpadel a ventilátorů a další

běžné akční členy mají většinou binární řízení a například u servomotorů bývá vhodné omezit počet akčních zásahů z důvodu zmenšení opotřebení zařazením pásma necitlivosti nebo hystereze, byla na výstup regulátoru zařazena možnost volby mezi více druhy výstupních bloků. Jedná se o lineární výstup se saturací, dvoustavové relé s hysterezí, sedmistavové relé a konstantní hodnotu. Lineární výstup poskytuje klasický výstup regulátoru a je omezen saturací s uživatelsky nastavitelnými mezemi. Dvoustavové relé poskytuje možnost binárního řízení zapnuto/vypnuto a má nastavitelné hodnoty pro sepnutý a rozepnutý výstup i horní a dolní meze hystereze (spínací a rozpínací úrovně). Sedmistavové relé má více využití. V prvním případě připojení více topných těles k jednomu silovému modulu umožňuje sdružením všech tří silových spínacích výstupů řídit celkový výkon těchto těles. V tomto případě se ale vyskytla chyba, protože relé by mělo být správně osmistavové, aby bylo schopné pokrýt plně rozsah tří spínačů (0-7). Ve všech případech ale zatím plně postačilo 7 výstupních úrovní, takže nebylo nutné zasahovat do již vyzkoušené a odladěné části systému. V druhém případě se využívá tohoto výstupu k nastavování polohy servomotorů řídicích klapky a směšovací ventily. Otevření těchto prvků není nutné řídit s přesností na jednotky procent, tudíž se k nastavení jejich polohy využívá sedmistavového relé. Poloha se nastavuje pouze v několika diskrétních krocích, čímž se výrazně zmenší počet změn polohy potažmo doba běhu motoru, což vede k prodloužení životnosti servomotoru. Konstantní hodnotu výstupu lze využít k ladicím a testovacím činnostem nebo k vyřazení regulátoru z činnosti.

Volba použitého typu výstupu se provádí označením položky „Enable“ nad patřičným výstupním blokem.

4.4.1.5 Binární vstupy regulátoru

Regulátor je dále doplněn o šestici binárních vstupů zapojených v logickém součinu. Funkce regulátoru je podmíněna logickým výsledkem součinu těchto vstupů. Pokud je nenulový, provádí se standardní výpočet akčního zásahu. V případě nulového výsledku závisí chování regulátoru na přepínači „Používat“ u editačního pole „Neaktivní výstup“. Pokud je políčko „Používat“ zatrženo, je do výstupního

kanálu zapsána hodnota uvedená v editačním poli „Neaktivní výstup“. Pokud zatržené není, ponechává se hodnota výstupního kanálu beze změny. První možnosti se využívá například v případě řízení elektrokotle, jehož napájení je několikrát denně vypnuto pomocí HDO signálu vysílaného energetickou společností. V okamžiku nepřítomnosti napájení nemá smysl spínat stykače. Z důvodu zamezení stavu, kdy jsou zapnuté stykače a není dodáván výkon, dojde k vypnutí výstupů. Druhé možnosti lze s výhodou využít, pokud se o jeden akční člen dělí dva regulátory. To je případ přepínání funkce řízení vzduchotechniky mezi topným a chladicím režimem. Jeden ventil pouští do výměníků buď teplou nebo studenou vodu, přičemž každá funkce potřebuje jinak nastavený regulátor (v podstatě inverzně) a přepínání těchto regulátorů může provádět obsluha nebo může být řízeno automaticky například na základě teploty vně objektu.

Tyto vstupy se sice nazývají binární, protože operace s nimi i její výsledek mají binární charakter, ale vstupovat sem mohou jakékoliv kanály. K přetypování na binární hodnotu dochází až po načtení hodnoty kanálu uvnitř třídy regulátoru.

4.4.1.6 Matematicko-výpočetní jednotka

Standardní funkce regulátoru je taková, že načítá hodnoty ze vstupních kanálů a na základě nastavených konstant a vstupních podmínek vypočítává velikost akčního zásahu, kterou následně zapisuje do výstupního kanálu. Použití regulátoru je ale mnohem širší. Pokud se místo výstupního kanálu situovaného na některém z modulů zadá jako výstupní kanál některá z propojovacích proměnných (jejich funkce je popsána v kapitole 4.4.4), může regulátor pracovat jako univerzální výpočetní nástroj například pro vypočítávání žádaných hodnot na základě naměřených hodnot (ekvitermní regulace), určování podmínek pro běh jiných regulátorů (přepínání topení/chlazení) atd. V neposlední řadě je možné tímto způsobem realizovat jednoduchý MISO systém, i když se jedná o poněkud komplikované a ne příliš efektivní řešení.

4.4.1.7 Třída cRegulator

Je nutné si uvědomit, že výše uvedené odstavce sice popisují vzhled a funkce okna grafického rozhraní regulační smyčky a postupy při nastavování jejich parametrů, ale že veškerá výpočetní činnost se provádí na úrovni třídy cRegulator, která je členskou proměnnou třídy cSystem. Výše popsané okno pouze vytváří rozhraní mezi uživatelem a třídou cRegulator, což je přísně v souladu s filozofií návrhu co nejvíce omezit závislost na prostředí Borland. Uživatelské okno pouze volá členské funkce této třídy a předává jí uživatelem zadané hodnoty nebo vypisuje obsah proměnných třídy do editačních polí uživatelského okna.

Každá regulační smyčka v seznamu je vlastně instancí třídy cRegulator a třída cSystem v sobě uchovává pole těchto instancí, které se v okamžiku přidání nebo smazání regulační smyčky ze seznamu dynamicky mění.

4.4.2 Žádané hodnoty a režimy úspory energie

V následujících dvou kapitolách 4.4.3 a 4.4.4 jsou popsány možnosti zadávání žádaných hodnot do systému. Předtím je ale vhodné popsat přístup k těmto hodnotám. Ten je stejný jako v případě všech ostatních hodnot v systému. Pohlíží se na ně jako na kanál, ze kterého lze číst a v některých případech do něj lze i zapisovat.

Systém podporuje dva režimy funkce: klasický režim a režim úspory energie, zkráceně denní a noční. Noční režim se projeví snížením žádaných hodnot vybraných veličin. Nejedná se o nic jiného, než že každá z žádaných hodnot nebo konstant je v systému obsažena dvakrát, jednou pro denní režim a jednou pro noční režim. Podle aktivního režimu se při čtení z kanálu načte odpovídající hodnota. Tímto způsobem je možné ovlivnit chování každého kanálu zvlášť podle aktuálních potřeb. Aktivní režim pro daný kanál může být nastaven třemi způsoby: manuálně, časově, jiným kanálem. Standardní přepínání režimů se provádí časově, je popsáno v kapitole 4.4.3 a má nejnižší prioritu. Vyšší prioritu má manuální nastavení a je popsáno také v kapitole 4.4.3. Obě tyto možnosti přepínání režimů mají globální platnost a řídí aktivní režim všech kanálů, které nemají nastaveno řízení režimu jiným kanálem. Řízení režimu jiným kanálem má pouze lokální platnost pro ten kanál, pro který je

nastaveno, a má nejvyšší prioritu. Detailní nastavení tohoto řízení režimu je popsáno v obou následujících kapitolách 4.4.3 a 4.4.4.

4.4.3 Tabulka žádaných hodnot v bytových jednotkách a řízení režimů

Žádané hodnoty výstupních veličin se mohou v systému vyskytovat ve dvou podobách. První možností je zadání hodnoty do tabulky konstant a propojovacích proměnných (viz kapitola 4.4.4), kde může být hodnota změněna pouze prostřednictvím řídicího počítače. Druhou možností je tabulka žádaných hodnot v bytových jednotkách (viz Obrázek 4.5), která nabízí uživateli pohodlnější kontrolu nad žádanými hodnotami systému.

4.4.3.1 Žádané hodnoty v bytových jednotkách

Protože je systém určen v první řadě k regulaci teploty v prostorách obývaných lidmi, je vhodné, aby si tito lidé mohli nastavit teplotu, jakou by rádi v dané místnosti měli. Nebo to naopak vhodné není, pokud se jedná o pracoviště s určitými normami a teplotu určuje vedoucí pracovník, v jehož zájmu není toto svým podřízeným umožnit. Pro případ, že si obyvatelé regulovaných prostor mohou sami zvolit, jakou teplotu chtějí ve svém okolí mít, poskytují zobrazovací moduly kromě funkcí měření a zobrazení teploty také funkci jejího nastavení. Pomocí tlačítek inkrementace, dekrementace a potvrzení volby lze v jednoduchém menu vybrat žádanou teplotu pro denní nebo noční režim a nastavit její velikost. Nutno podotknout, že se opět jedná o obecný vstupní kanál a jeho hodnota může reprezentovat jakoukoliv fyzikální veličinu, nikoliv jen teplotu. Aby bylo ale možné provést nastavení žádaných hodnot ve zobrazovacích modulech i centrálně z řídicího počítače a nebylo nutné zadávat tyto hodnoty ručně na všech modulech, což by bylo obzvlášť v případě rozlehlých instalací značně nepohodlné, byla vytvořena tabulka žádaných hodnot v bytových jednotkách.

RegTherm Tabulka žadaných hodnot v bytových jednotkách

Soubor

kanál	Den [°C]	Noc [°C]	Kanál řídicí úsporný režim	Inv (Y/N)
Teplota karosama	0	0		N
Teplota lakovna	0	0		N
Teplota servis	0	0		N
Teplota STK	0	0		N
Teplota kancelar	0	0		N
Teplota showroom	0	0		N

Nastavení režimů

☒ Automaticky **DEN**

Denní režim start Noční režim start

Pondělí

Denní režim od: 04:00 Noční režim od: 15:00

Úterý

Denní režim od: 04:30 Noční režim od: 15:00

Středa

Denní režim od: 04:30 Noční režim od: 15:00

Čtvrtek

Denní režim od: 04:30 Noční režim od: 15:00

Pátek

Denní režim od: 04:30 Noční režim od: 15:00

Sobota

Denní režim od: 05:00 Noční režim od: 12:00

Neděle

Denní režim od: 00:00 Noční režim od: 00:05

Obrázek 4.5: Tabulka žadaných hodnot v bytových jednotkách

Tato tabulka je generována po naskenování sběrnice nebo po načtení konfigurace ze souboru a je pevně svázána se zobrazovacími moduly – bytovými jednotkami, kde každému zobrazovacímu modulu nainstalovanému v systému odpovídá jeden řádek v tabulce. Každý zobrazovací modul obsahuje fyzicky dva vstupní kanály, jejichž hodnoty může uživatel nastavit, přičemž se zpravidla jedná o hodnoty pro denní a noční režim, a jeden fiktivní vstupní kanál, který načítá data z jednoho z těchto kanálů podle aktivního režimu. Obecně nemusí mít dvojice zadávaných hodnot nic společného, ale potom použití třetího „přepínaného“ kanálu ztrácí smysl. Názvy všech tří kanálů se zadávají při konfiguraci systému v servisním okně a do tabulky žadaných hodnot v bytových jednotkách se načítá automaticky název třetího fiktivního kanálu.

Jak již bylo uvedeno, jednotlivé řádky tabulky odpovídají jednotlivým zobrazovacím modulům v systému. Také hodnoty pro denní a noční režim

odpovídají hodnotám uloženým v těchto modulech. Vzniká zde ale problém se směrem toku informací. Kanály pro zadávání žádaných hodnot jsou z hlediska systému chápány jako vstupní, protože informace proudí od uživatele skrz zobrazovací modul do řídicího počítače. Ovšem v okamžiku nastavování žádaných hodnot z řídicího počítače se směr toku informace otáčí. Data jsou odeslána z řídicího počítače do modulu, tam mohou být zobrazena a následně jsou standardně načítána regulačními algoritmy. Z tohoto důvodu jsou v menu „Soubor“ položky „Načíst hodnoty z modulů“ a „Uložit hodnoty do modulů“. Načítání se sice provádí automaticky, ale doporučuje se před nastavováním hodnot v řídicím počítači provést synchronizaci dat načtením z modulů. Naopak při odesílání nových hodnot do modulů je volba položky v menu jedinou možností jak toho dosáhnout.

Menu „Soubor“ dále nabízí standardní položky pro ukládání dat do souboru „Načíst“, „Uložit“ a „Nastavit jako výchozí“. Funkce těchto příkazů je ve všech modulech totožná.

4.4.3.2 Kanál řídicí režim žádané hodnoty

Nastavení kanálu řídicího režim dané žádané hodnoty se provádí stejně jako nastavení všech kanálů v systému: dvojitým kliknutím ve sloupci „Kanál řídicí úsporný režim“ v řádku odpovídajícím nastavované žádané hodnotě a výběrem kanálu ze zobrazeného seznamu. Smazání tohoto řídicího kanálu se provádí příkazem „Smazat řídicí kanál“ v pop-up menu. Přepínání režimu v tomto případě probíhá tak, že pokud je hodnota řídicího kanálu nenulová a invertování je neaktivní, je nastaven noční režim. V případě, že je invertování řídicího kanálu aktivní, je funkce opačná. Invertování se nastaví zadáním hodnoty „Y“ do odpovídajícího řádku a sloupce „Inv (Y/N)“.

4.4.3.3 Nastavení globálního řízení režimů

V okně s tabulkou žádaných hodnot v bytových jednotkách se nachází i ovládací panel k nastavování globálního přepínání režimů. V okamžiku, kdy u některé z žádaných hodnot, konstant nebo propojovacích proměnných (jejich funkce je vysvětlena v kapitole 4.4.4) není nastaven kanál řídicí režim, je režim této

hodnoty řízen globálně. Standardně se využívá přepínání režimu na základě času. Okno ve své pravé části nabízí pro každý den dvě editační pole k nastavení začátku denního a nočního režimu. Pokud je zatrženo políčko „Automaticky“, probíhá přepínání režimů na základě těchto nastavených časů. V horní části ovládacího panelu je vypsán aktuální aktivní režim. Uživatel si může manuálně vynutit změnu režimu na denní stiskem tlačítka „Denní režim start“ a na noční analogicky „Noční režim start“. Tato změna ale není trvalá a má platnost pouze do okamžiku nejbližšího časového přepnutí režimu. Dále se potom přepínání řídí opět časově podle nastavených hodnot. To platí ovšem pouze v případě zatržení pole „Automaticky“. Pokud toto pole zatrženo není, je volba globálního režimu plně v rukou uživatele.

4.4.3.4 Skutečná destinace dat

Stejně jako všechna ostatní okna v tomto systému poskytuje okno s Tabulkou žádaných hodnot v bytových jednotkách pouze uživatelské rozhraní a nenese v sobě žádná data. V případě nastavených žádaných hodnot se jedná o rozhraní mezi uživatelem a instancemi třídy `cRoomUnitData`, které v sobě nesou tyto hodnoty uložené po dobu běhu systému. Ovládací panel přepínání režimů tvoří nadstavbu přímo instance třídy `cSystem`, která provádí správu a přepínání režimů.

Přestože jsou tato data v době běhu systému uložena v různých třídách, jsou na disk ukládána do jednoho společného souboru.

4.4.4 Tabulka konstant a propojovacích proměnných

Tabulka konstant a propojovacích proměnných je zobrazena na obrázku Obrázek 4.6. Slouží jako seznam žádaných hodnot regulovaných veličin v systému, které ale mohou být měněny pouze prostřednictvím řídicího počítače a to jak uživatelem, tak automaticky výpočtem.

RegTherm Konstanty a propojovací proměnné

Soubor

Název	Den	Noc	Kanál řízení spotřeby - řízení módu Den/Noc	Inv [Y/N]
NULL	0	0		N
Výstupní voda zadána	40	40		N
Tepl. lakovna zadána	19	17		N
Top. voda lakovna vypočtena	40	20		N
Tepl. karosarna zadána	19	17		N
Vzduch karosarna vypočt.	40	20		N
Tepl. servis zadána	19	17		N
Top. voda servis vypočtena	40	20		N
Tepl. STK zadána	19	17		N
Top. voda STK vypočtena	40	20		N
Tepl. kancelar zadána	22	22		N
Top. voda kanc. vypočtena	40	40		N
Tepl. showroom zadána	21	19		N
Top. voda showroom vypočtena	40	30		N
Tepl. bojler zadána	40	40		N
Bojler enable	1	1		N
Chod ventu vzdtech	1	0		N
Chod ventu vzdtech karosarna	1	0		N
Teplota vzduchu zadána	30	15		N
Teplota vzd. karos. zadána	30	15		N
Vzdtech karos. enable	0	0		N

Přidat Smazat Smazat kanál řídicí mód Exit

Obrázek 4.6: Tabulka konstant a propojovacích proměnných

Konstant a propojovacích proměnných může být v systému libovolné množství. Lze je přidávat a odebírat pomocí tlačítek „Přidat“ a „Smazat“, jen při mazání je nutné provést kontrolu, aby nedošlo ke smazání používané konstanty, protože by tím došlo k narušení funkčnosti celého systému. Doporučený postup je nastavit konstanty a proměnné v okamžiku konfigurace systému a v případě nutnosti pozdějších úprav konstanty přidávat na konec seznamu.

Stejně jako v případě žádaných hodnot v bytových jednotkách je každá konstanta zadávána dvakrát. Jedna hodnota pro denní režim a druhá pro noční režim. I v tomto případě lze nastavit kanál pro řízení režimu dané konstanty stejným

postupem: dvojitým kliknutím na sloupec „Kanál řízení spotřeby...“ a výběrem kanálu ze seznamu. Funkce sloupce „Inv (Y/N)“ je stejná jako v předchozím případě.

4.4.4.1 Konstanty

Z hlediska systému je konstanta vstupním kanálem, jenž mění nebo nemění svoji hodnotu na základě aktivního režimu. Konstantou je tento kanál nazýván proto, že je mu uživatelem přiřazena dvojice hodnot a tyto hodnoty mohou být měněny opět jen zásahem uživatele.

Lze jich využít jako konstantních žádaných hodnot nebo těch žádaných hodnot, které může měnit jen některý z uživatelů systému (zpravidla administrátor), různých kalibračních nastavení, korekčních hodnot atd.

4.4.4.2 Propojovací proměnné

Propojovací proměnná je v době svého vzniku konstantou. Vytvoří se standardně přidáním položky do tabulky tlačítkem „Přidat“ a nastavením jejího názvu a hodnot pro denní a noční režim. V této chvíli je stále konstantou, která může být použita jako vstupní kanál pro regulátory nebo vizualizaci.

Propojovací proměnnou se konstanta stává v okamžiku, kdy je do ní nasměrován výstup nějakého výpočetního bloku, kterým je ve stávající verzi pouze regulátor. Tím, že se stane výstupním kanálem jednoho regulátoru, spojí de facto výstup tohoto regulátoru se vstupem jiného, pokud byl nastaven. Odtud vznikl název „Propojovací proměnná“. Jedná se o velmi výhodnou vlastnost, neboť lze tímto způsobem řadit kaskádně jednotlivé regulátory a vzhledem k jejich univerzálnosti popsané v kapitole 4.4.1 tím získat potřebné matematické operace.

Tato proměnná ale i nadále podporuje denní a noční režim. Zápis do ní coby do výstupního kanálu se provádí tím způsobem, že se nejprve vypočte difference hodnot pro denní a noční režim a poté se uloží zapisovaná hodnota do části udávající denní režim. Nakonec se do části pro noční režim uloží rozdíl hodnoty pro denní režim a na začátku vypočtené difference. Tímto způsobem se dosáhne toho, že uživatel v době vytváření proměnné zadá hodnoty pro denní a noční režim a rozdíl

mezi nimi bude zachován po celou dobu běhu systému. Při čtení proměnné jako vstupního kanálu se do kanálu vkládá hodnota odpovídající aktivnímu režimu.

4.4.5 Vizualizační okno, hlavní okno aplikace

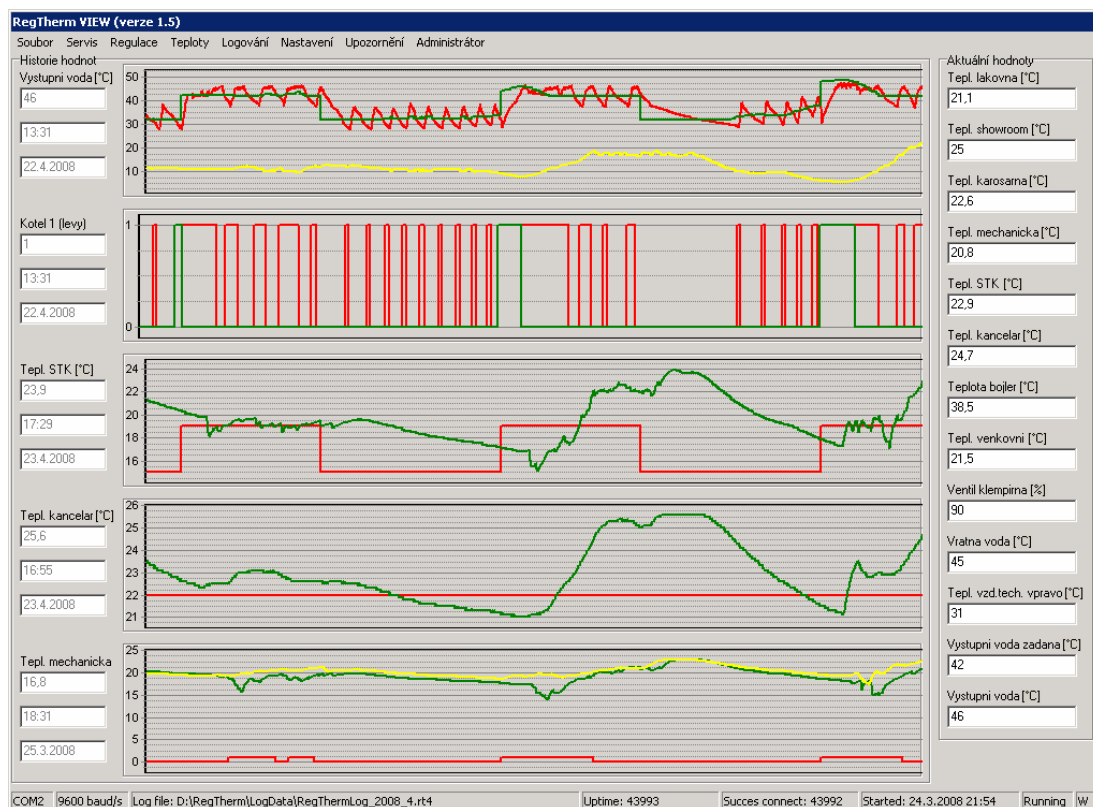
4.4.5.1 Vizualizace

Aby bylo možné sledovat a kontrolovat průběhy hodnot jednotlivých kanálů a následně nastavovat regulační algoritmy, nabízí systém okno RegTherm VIEW znázorněné na obrázku Obrázek 4.7. K potřebám vizualizace procesu nabízí toto okno 5 grafů s historií zobrazení 3 dny a 13 zobrazovacích polí pro výpis libovolných aktuálních hodnot. Každý z grafů může navíc obsahovat až 5 křivek různých průběhů, aby bylo možné jednotlivé průběhy snadno porovnávat jak v čase, tak ve výchylce. Zobrazit lze jakýkoliv kanál obsažený v systému.

V případě některého ze 13 polí zobrazujících aktuální hodnoty se zobrazovaný kanál nastaví výběrem ze seznamu zobrazeného dvojitým kliknutím na patřičné zobrazovací pole. V těchto polích jsou zobrazeny aktuálně naměřené a vypočtené hodnoty a také aktuální reálné fyzické hodnoty výstupů. Nad každým z nich je navíc uveden název zobrazovaného kanálu.

Pro nastavení zobrazovaných průběhů do grafu je k dispozici pop-up menu, přičemž každý z pěti grafů nabízí vlastní menu vztahující se ke křivkám zobrazovaným v daném grafu. Všechna tato menu mají ale stejnou strukturu: V první úrovni nabízí položky „Line 1“ až „Line 5“ odpovídající křivkám 1 – 5. V druhé úrovni potom nabízí ke každé z křivek položku „Select“ pro zobrazení seznamu kanálů a následný výběr zobrazovaného kanálu, položku k zatržení „Enabled“, která určuje, zda má být daná křivka zobrazována, a položku „Marks“ zobrazující u křivky ‚vlaječky‘ s hodnotami. V případě zobrazení jedné a více křivek musí první křivka být vždy zobrazena. Pro snazší odečítání hodnot z grafu jsou umístěna v levé části okna před každým z grafů tři zobrazovací pole. Do těch se v okamžiku kliknutí myši na křivku průběhu vypíše hodnota daného kanálu v tomto bodě a čas a datum, ke kterému se tato hodnota vztahuje. Nad první z těchto polí se dále vypíše název tohoto kanálu. Je-li to třeba, lze data zobrazovaná v grafech vymazat volbou nabídky

„Smazat historii zobrazovaných dat“ v menu „Soubor“. Tímto se vymaže, respektive nastaví na hodnotu nula, veškerá historie dat uložená v instancích tříd cServoData a cRoomUnitData.



Obrázek 4.7: Okno vizualizace procesu

Tloušťku zobrazovaných křivek lze nastavit v rozsahu 1 – 5 volbou příslušné položky z menu „Nastavení“ → „Tloušťka křivek“. Výchozí tloušťka je 2.

Nastavení zobrazovaných kanálů lze samozřejmě uložit do souboru a následně z něj i vyčíst pomocí standardních příkazů „Načíst nastavení“, „Uložit nastavení“ a „Nastavit jako výchozí“.

4.4.5.2 Ovládání aplikace

Okno RegTherm VIEW je navíc z pohledu operačního systému Windows hlavním oknem aplikace RegTherm, a proto je tedy z oken aplikace spouštěno jako

první. To je výhodné z toho důvodu, že u nastaveného a odladěného systému není zpravidla nutné otevírat další okna a uživateli stačí pouze sledovat aktuální průběhy.

Z hlediska hierarchie softwaru je důležité, že toto okno obsahuje instanci třídy cSystem, jež je softwarovým jádrem celého systému. Jedná se tedy o hlavní okno aplikace, které provádí potřebné inicializace a spouštění a navíc umožňuje uživateli přístup ke všem dalším oknům. Ta se otevírají vybráním patřičné položky z menu. Mohou být otevírána v režimu náhledu nebo editace, což je ale předmětem kapitoly 4.4.8.

V menu „Soubor“ se nachází dva velmi důležité příkazy „Start“ a „Stop“. Těmito příkazy se spouští a zastavuje vykonávání hlavního vlákna aplikace, tedy veškerá měřicí a regulační činnost.

4.4.5.3 Stav aplikace

Vizualizační okno navíc zobrazuje i stav základních komponent nutných pro správný běh systému. Jedná se o zvolený komunikační port, kterým je v současné době sériový port COM1 – COM9, nastavenou přenosovou rychlost tohoto portu nebo text „Error“ v případě, že se nepodařilo port otevřít, a úplný název souboru pro logování dat (viz kapitola 4.4.6).

Dále je zde uveden počet vzorkování od spuštění systému, počet úspěšných komunikačních událostí, datum a čas spuštění systému, stav spuštění funkčního vlákna a kód stavu jeho provádění.

Z hlediska uživatele sice tyto údaje nejsou podstatné, ale jsou vhodné pro ladění a testování systému.

4.4.6 Ukládání naměřených a vypočtených hodnot

4.4.6.1 Logování

Aby bylo možné zpětně kontrolovat chování systému, zjišťovat parametry regulovaných soustav nebo sledovat okolní podmínky, nabízí systém možnost ukládání dat do souboru – logování. To probíhá tím způsobem, že se v okamžiku vzorkování přidávají aktuálně naměřené a vypočtené hodnoty do zvoleného logovacího souboru, jehož obsah lze následně zobrazit. V aktuální verzi se ukládají data ze všech

kanálů v systému. To se ale ukázalo jako nevýhodné, protože velké množství kanálů v modulech není fyzicky připojených, tudíž tato data nemají žádný význam a pouze zabírají místo v paměti. Zabírání místa na disku je ale bezvýznamné ve srovnání s rostoucí dobou nutnou pro práci s logovacím souborem. Momentálně je tato situace řešena rozdělením logovaných dat do více souborů, což bude vysvětleno dále. V budoucích verzích systému se ale předpokládá změna formátu ukládaných dat a ukládání hodnot pouze uživatelem vybraných kanálů.

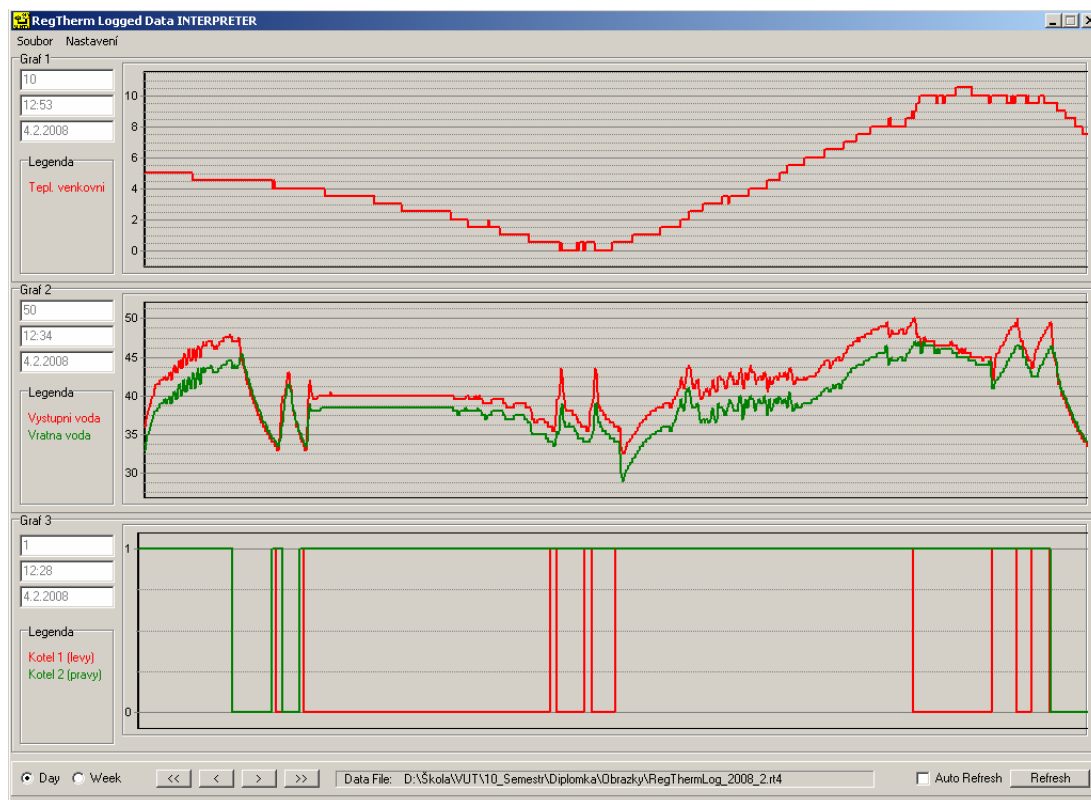
4.4.6.2 Přepínání logovacího souboru

Z výše zmíněného důvodu prodlužování doby práce souborem a kvůli zpřehlednění, jednodušší orientaci a interpretaci dat jsou průběžně vytvářeny nové logovací soubory. V aktuální verzi se automaticky vytváří vždy nový soubor pro nový kalendářní měsíc. Tento soubor má standardně předdefinovaný název ve tvaru „RegThermLog_rrrr_mm.rt4“, kde „rrrr“ udává rok a „mm“ měsíc vzniku dat. I v případě měsíční změny logovacího souboru ale velikost tohoto souboru při větším počtu modulů nainstalovaných v systému neúměrně roste, takže bude nutné v budoucnu ještě zkrátit tento interval nebo provést změnu formátu a množství ukládaných dat.

4.4.6.3 Interpretace uložených dat

K interpretaci uložených dat slouží okno „RegTherm Logged Data INTERPRETER“ zobrazené na obrázku Obrázek 4.8. Jeho funkce je obdobná jako funkce hlavního vizualizačního okna aplikace. Obsahuje ale pouze 3 grafy, aby bylo možné dosáhnout většího rozlišení v ose výchylky, přičemž maximální počet pěti křivek v každém grafu zůstal zachován. Pro zvětšení rozlišení ve směru časové osy lze zvolit délku zobrazení jeden den nebo jeden týden. To se provádí zatržením jednoho z polí „Day“ nebo „Week“ v levé dolní části okna. Nalevo od každého z grafů jsou umístěna stejně jako ve vizualizačním okně zobrazovací pole pro výpis hodnoty, data a času při kliknutí na křivku průběhu a pod nimi je umístěno pole „Legenda“. Do něj se vypisují názvy zobrazených kanálů. Barva písma názvu kanálu odpovídá barvě vykresleného průběhu hodnot tohoto kanálu. Výběr zobrazovaných

kanálů se provádí stejně jako ve vizualizačním okně volbou v pop-up menu každého z grafů.



Obrázek 4.8: Okno interpretace dat

Volba souboru k zobrazení dat se provádí příkazem „Načíst“ z menu „Soubor“ a pomocí standardního otevíracího dialogu systému Windows se zvolí požadovaný soubor. Na základě dat z tohoto souboru se vytvoří seznam kanálů, které jsou v tomto souboru uloženy, a jednotlivé kanály se pak zobrazují standardně výběrem z tohoto seznamu. Název vybraného souboru se i s cestou zobrazuje ve spodní části interpretačního okna. Pohyb v zobrazovaných kanálech se provádí pomocí tlačítek „<“ a „>“ o den a „<<“ a „>>“ o týden. Kromě položky „Soubor“ nabízí menu ještě položku nastavení, pomocí níž lze zvolit tloušťku zobrazovaných křivek v rozsahu 1 – 5, přičemž výchozí tloušťka je 2.

4.4.7 Bezpečnostní systém

Aby bylo zmenšeno riziko havárie, podporuje systém několik bezpečnostních mechanismů a to buď automatických nebo vyvolávajících reakci uživatele. V následujících kapitolách 4.4.7.1, 4.4.7.2 a 4.4.7.3 jsou popsány tři základní implementované úrovně ochrany.

Stávající zabezpečení systému proti selhání a upozornění uživatele na toto selhání je použito v minimální možné konfiguraci tak, aby regulační systém nezačal působit škody. Navíc je ochrana zaměřena především na komunikaci mezi řídicím počítačem a jednotlivými moduly, jelikož se právě komunikace ukázala jako nejrizikovější část systému. Pokud ale dojde k selhání jednotlivých částí systému jako například senzorů nebo akčních členů, popřípadě selhání řídicího počítače, uživatel tento stav zjistí až tím způsobem, že celý systém nebo jeho část neplní požadovanou funkci, což se v tomto případě systému regulace vytápění projeví tak, že netopí. To není ale z hlediska požadavků na automatický systém příliš vhodné, proto byly navrženy další zabezpečovací mechanismy popsané v kapitolách 4.4.7.4, 4.4.7.5 a 4.4.7.6. Tyto mechanismy nejsou ve stávající verzi implementovány, ale počítá se s jejich nasazením v dalších verzích, protože výrazně zrychlí odhalování závad a v určitých případech jim mohou pomoci předejít.

4.4.7.1 Vyřazení regulátoru z provozu

Vyřazení regulátoru z provozu chrání systém a regulovanou soustavu před havárií, která by mohla nastat vlivem změny nějakého měřitelného stavu v systému. V tomto případě se využívá binárních vstupů regulátoru popsaných v kapitole 4.4.1.5. V okamžiku vzniku rizikového stavu dojde k odpojení regulátoru zapsáním předem nastavené neaktivní hodnoty jeho výstupu.

Tato ochrana je určena proti změnám v regulovaných soustavách či změnám provozních podmínek a předpokládá plnou funkčnost regulačního systému. V případě selhání systému selhává i tato ochrana.

4.4.7.2 Nastavení bezpečných hodnot výstupů

Ochrana nastavením bezpečných hodnot výstupů je hlavním bezpečnostním prvkem systému. Je určena proti selhání regulačního systému a má dvě úrovně jištění.

Nižší úroveň je splněna již na úrovni hardwaru tím, že v okamžiku výpadku napájecího napětí modulu přejdou všechny jeho výstupy do rozepnutého nebo neaktivního stavu. Vyšší úroveň zabezpečení je tvořena jakýmsi ‚komunikačním watch-dogem‘. Modul má uloženy bezpečné hodnoty výstupů zadané uživatelem v servisním okně a pokud neproběhla po určenou dobu úspěšná komunikační událost nastavující nové stavy výstupů, jsou na tyto výstupy zapsány právě tyto bezpečné hodnoty. V případě úspěšné komunikační události se výstupy nastaví podle nově přijatých hodnot a systém pokračuje dále v činnosti.

Tato ochrana je na uživateli naprosto nezávislá a není možné změnit její nastavení nebo ji vypnout. Chrání regulovanou soustavu před výpadkem komunikace v systému ať již vlivem přerušení vodičů nebo například chybou řídicího počítače i před celkovým selháním řídicího počítače. V případě přerušení komunikace se spouští bezpečnostní mechanismus popsany v kapitole 4.4.8. V případě selhání řídicího počítače se pouze odpojí výstupy a je nutný zásah administrátora systému.

4.4.7.3 Systém upozornění na selhání

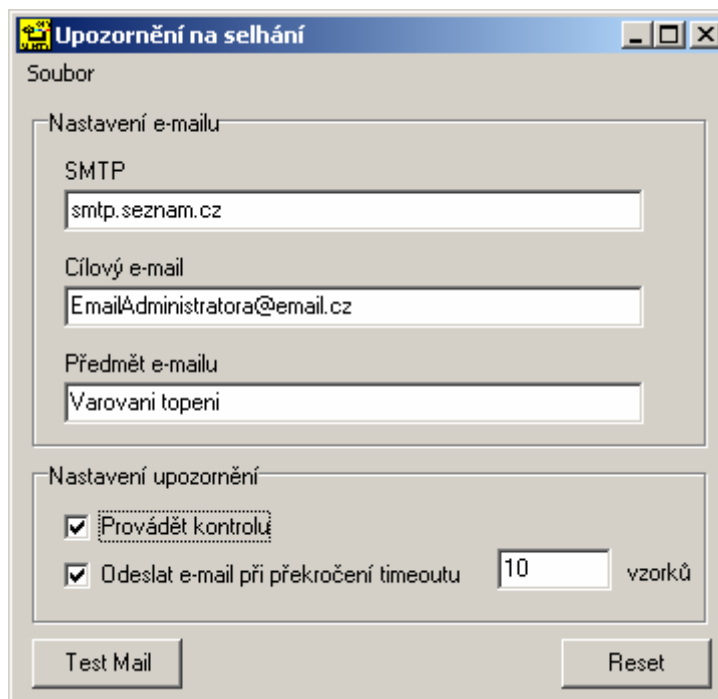
Ve stávající verzi systému je implementován jednoduchý mechanismus informování uživatele o selhání systému. V případě výskytu chyby je uživateli odeslán informační email s jejím popisem. Okno s nastavením tohoto emailu zobrazuje Obrázek 4.9. Je nutné nastavit adresu cílového emailu, na niž má být zpráva odeslána, a předmět této zprávy. Její tělo je vyplněno informací o vzniklé chybě. SMTP server se doporučuje zanechat stávající, protože systém využívá vlastní emailovou schránku zřízenou pro jeho potřeby

Ve spodní části okna je ovládací panel k nastavení kontrolovaných událostí. Aktuálně je podporována pouze kontrola komunikační rutiny pro jednotlivé moduly. Zatřetím pole „Provádět kontrolu“ se globálně povolí kontrolování nastavených událostí. Druhou položkou je zatrhávací pole „Odeslat email při překročení

timeoutu“ a editační pole pro nastavení počtu timeoutů, po kterém má být odeslán informační email. Funkce je taková, že pokud je komunikační událost s některým z modulů po nastavený počet po sobě jdoucích vzorků neúspěšná, je odeslán email obsahující název, adresu a počet timeoutů daného modulu. Odesláním emailu dojde k zablokování této kontroly, aby nedošlo v případě velkého výpadku k zahlcení schránky administrátora. I kdyby došlo k dalším selháním, předpokládá se, že administrátor provede kontrolu celého systému. Po opravě vzniklé závady je nutné tlačítkem „Reset“ opět povolit provádění kontroly.

Tlačítko „Test Mail“ slouží k odeslání kontrolního testovacího emailu na zadanou adresu, aby bylo možné ověřit funkčnost nastaveného spojení.

Standardní menu „Soubor“ umožňuje uložit a načíst nastavení ze souboru a nastavit aktuální soubor jako výchozí.



Obrázek 4.9: Nastavení informačního emailu

4.4.7.4 Kontrola hodnoty kanálu

Kontrola hodnoty kanálu vychází z předpokladu, že hodnoty naměřené nějakým čidlem se mohou ať již z fyzikálních důvodů nebo z důvodu možných

provozních podmínek pohybovat pouze v určitém intervalu a výstupní hodnota tohoto čidla je časově proměnná vlivem proměnných okolních podmínek vstupujících do systému z regulačního hlediska většinou jako poruchy.

Jedním stupněm kontroly hodnoty kanálu je nastavení mezí intervalu, ve kterém se hodnoty tohoto kanálu mohou pohybovat. Tyto meze jsou samozřejmě různé pro jednotlivé regulované soustavy i měřené fyzikální veličiny, ale z principu mohou být vždy nastaveny. Potom už záleží na nastavení vyhodnocovacího algoritmu. Může být sestavován jakýsi statistický přehled a při dosažení určitého počtu překročení nastavených mezí generována informační zpráva nebo v jednodušší verzi může být zpráva vytvořena při dosažení nastaveného počtu překročení nastavených mezí za sebou. Způsob vyhodnocení je závislý na povaze měřené veličiny a na požadavcích na zabezpečení konkrétního systému.

Druhým stupněm je kontrola průběhu naměřených hodnot. Vlivem změn okolních podmínek regulované soustavy dochází i ke změnám hodnot naměřených veličin. Z tohoto předpokladu vychází úvaha, že hodnota vstupu, která je konstantní po „příliš“ dlouhou dobu, značí poruchu tohoto vstupu a tím generování varovné zprávy. Problémem je definování relativního pojmu „příliš“. Empirickým pozorováním dějů v konkrétním systému je možné tento čas kvalifikovaně odhadnout a nastavit jej do daného algoritmu. Je také možné zvolit „tvrdší“ podmínky pro kontrolu systému a v případě generování falešných poplašných zpráv tyto podmínky zmírnit. Nastavení tohoto algoritmu ale závisí na vlastnostech konkrétní regulované soustavy a je pro každou instalaci systému specifické.

4.4.7.5 Kontrola sběrnice nezávislým modulem

Jak bylo uvedeno v kapitolách 4.4.7.2 a 4.4.7.3, je prováděna kontrola komunikace z části řídicím počítačem a z části jednotlivými moduly. Řídicí počítač v případě souvislého komunikačního výpadku odesílá administrátorovi systému varovnou zprávu, zatímco jednotlivé moduly nastavují svoje výstupy do neaktivních úrovní. Nabízí se však možnost provádět kontrolu komunikace mezi řídicím počítačem a jednotlivými moduly speciálním k tomu určeným modulem. Tento modul by byl připojen na komunikační sběrnici RS485 a monitoroval by pohyb

paketů po sběrnici. Dále by byl doplněn nezávislým GSM modulem pro možnost odesílání a příjmu SMS zpráv, GPRS modulem pro připojení k Internetu a komunikaci například prostřednictvím emailu, popřípadě kombinací těchto řešení. Tímto způsobem je potom možné velmi komfortně zasílat administrátorovi informace o provozu sítě. Je možné vytvářet statistiku správných a chybných paketů, statistiku výpadků atd. Vzhledem ke komunikační hierarchii master/slave je snadné určit, který z modulů zasílá vadné pakety, popřípadě nezasílá pakety žádné, či který vlivem nějaké poruchy nedodrжуje například časování komunikace.

Veškerá výše popsaná kontrola by mohla být prováděna i řídicím počítačem. To však vyžaduje bezchybnou činnost tohoto počítače, což nikdy není možné zaručit. Doplněním systému zvláštním kontrolním modulem se vytvoří jistá míra redundance, která umožní kontrolovat i činnost řídicího počítače. V případě poruchy tohoto počítače ustane veškerý datový pohyb na sběrnici, což je modulem okamžitě detekováno. Další nespornou výhodou je použití nezávislého GSM či GPRS modulu. V případě výpadku internetového připojení řídicího počítače nebude odeslání varovné zprávy možné a administrátor o vzniklé situaci nebude informován. Toto řešení s sebou sice nese zvýšení nákladů na provoz systému ve formě poplatku telefonnímu operátorovi za využívání této služby, ale výhoda rapidního zvýšení zabezpečení systému proti selhání a informování o případném selhání je nesporná.

4.4.7.6 Kontrola a ovládání řídicího počítače nezávislým modulem

V kapitole 4.4.7.5 byla navržena možnost kontroly komunikační sběrnice nezávislým modulem. Funkce tohoto modulu ale může být mnohem větší. Nabízí se možnost, aby tento modul zároveň komunikoval s řídicím počítačem prostřednictvím některého z rozhraní tohoto počítače, například RS232 nebo USB. Tímto by bylo možné vytvořit jakýsi systémový watch-dog. Regulační aplikace by odesílala prostřednictvím nového komunikačního rozhraní stav svého hlavního funkčního vlákna a kontrolní modul by kontroloval, zda ke změnám stavu dochází v pravidelných přesně určených intervalech. V okamžiku přerušení běhu hlavního vlákna modul okamžitě reaguje odesláním varovné zprávy. V krajním případě může provést restart řídicího počítače.

Další funkcí kromě kontroly vývoje stavu hlavního vlákna aplikace může být vytvoření redundantního informačního kanálu k uživateli. V okamžiku, kdy pokus o odeslání informace pomocí standardního internetového připojení řídicího počítače selže, je možné využít některý z komunikačních modulů GSM nebo GPRS instalovaných na kontrolním modulu a odeslat pomocí něj zprávu uživateli.

Standardní formou vzdálené správy systému je využití služby systému Windows „Vzdálená plocha“, kdy se uživatel připojí k řídicímu počítači a pracuje s ním, jakoby u něj fyzicky byl. Tato snadná forma vzdáleného přístupu ke všem funkcím systému byla také jedním z důvodů jeho koncepce na platformě systému Windows. Vyžaduje však, aby uživatel měl k dispozici internetové připojení jak na svojí straně, tak na straně řídicího počítače. Použitím kontrolního modulu s GPS modulem by bylo možné zadávat řídicímu počítači povely na dálku bez internetového připojení například z dopravních prostředků či z dovolené. Tato činnost vyžaduje pouze mobilní telefon, který je dnes běžným standardem.

Další možností je situace, kdy v místě instalace řídicího počítače není vůbec k dispozici internetové připojení. V tomto případě se zadávání povelů pomocí GSM modulu stává jedinou možností vzdálené správy systému, pokud tedy pomineme GPRS modul, pomocí nějž by musel být připojen přímo řídicí počítač. V případě jeho instalace do kontrolního modulu by musel být vytvořen zvláštní standard ovládání, což není výhodné. Správa systému pomocí GSM modulu sice není zdaleka tak komfortní jako připojení k řídicímu počítači pomocí „Vzdálené plochy“, nicméně nabízí jednoduchou možnost realizace vzdálené správy.

4.4.8 Uživatelský a administrátorský mód

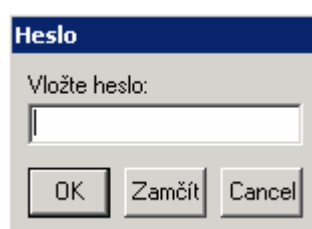
Jelikož řídicí počítač může být umístěn i v prostorách přístupných lidem nezaškoleným pro práci se systémem, nebo systém může spravovat více uživatelů, z nichž jen někteří mají kvalifikaci jeho nastavování, systém nabízí dvě úrovně ovládání: uživatel a administrátor.

V úrovni administrátor jsou přístupná všechna nastavení a volby. Systém může být konfigurován pouze v tomto režimu. Režim administrátora je chráněn heslem. Standardně může být na základě znalosti starého hesla vytvořeno heslo nové.

V režimu uživatele nejsou přístupné editační funkce v servisním okně, nelze nastavovat konstanty regulátorů a varovné mechanismy. Nastavovat lze v tomto režimu pouze žádané hodnoty.

Zabezpečení není použito z důvodu ochrany proti úmyslnému útoku či poškození systému, ale spíše jako ochrana proti nežádoucímu neúmyslnému zásahu. Rozdělení na režim administrátora a uživatele je ale převážně předmětem dohody se zákazníkem a bude upravováno podle jeho požadavků.

Volba módu se provádí výběrem položky „Administrátor“ v hlavním menu. Tímto příkazem se otevře okno zobrazené na obrázku Obrázek 4.10. Pro přepnutí do módu administrátora je nutné zadat heslo a potvrdit tlačítkem „OK“, pro přechod do módu uživatele stačí stisknout tlačítko „Zamčít“. Tlačítko „Cancel“ má standardní funkci zrušení volby a zavření okna.



Obrázek 4.10: Okno spuštění módu administrátora

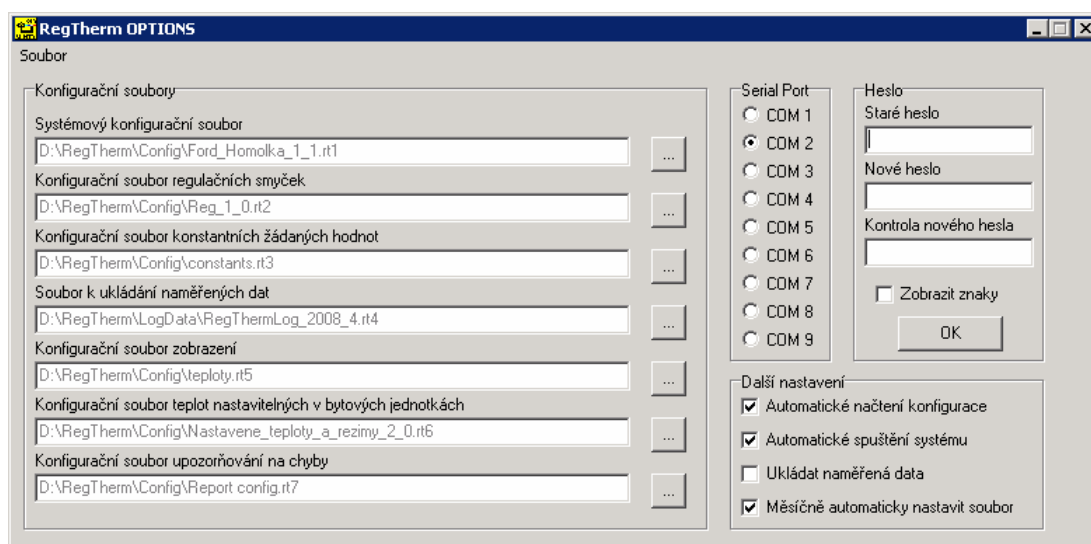
4.4.9 Inicializace a spouštění systému

Systém se spouští stejně jako všechny aplikace systému Windows dvojitým kliknutím na jeho ikonu. Protože by toto ale samo o sobě bylo nedostačující a i v případě krátkodobého výpadku nebo jen poklesu napájecího napětí by došlo k restartu systému Windows a tím i k ukončení aplikace, bylo spouštění systému upraveno následujícím způsobem: Řídící počítač je konfigurován tak, že se v okamžiku připojení k napájecímu napětí sám spouští a automaticky se přihlásí k účtu administrátora počítače. V rámci tohoto účtu je zástupce aplikace regulačního systému umístěn v menu „Po spuštění“, takže dojde k okamžitému spuštění systému. Dále již záleží na nastavení systému. Nakonfigurovaný systém je zpravidla nastaven tak, že okamžitě po svém spuštění načte všechny konfigurace ze zvolených souborů, odešle všem modulům příkaz k přepnutí do aplikačního režimu a spustí hlavní

vlákno aplikace. Tím je zajištěno, že se systém automaticky spustí po jakémkoliv výpadku napájení a bude dále pokračovat ve své činnosti. Systém může být také konfigurován tak, že po svém spuštění pouze načte patřičné konfigurace, nebo že neprovádí žádnou činnost a čeká na povely uživatele. Z hlediska automatického systému ale přichází v úvahu pouze první možnost, zbylé dvě jsou použity jen v době nastavování systému. Způsob tohoto nastavení je popsán v kapitole 4.4.10.

4.4.10 Okno nastavení systému

Na obrázku Obrázek 4.11 je zobrazeno okno nastavení systému. Jsou zde uvedeny položky nezbytně nutné ke správné funkci regulačního systému.



Obrázek 4.11: Okno nastavení systému

V levé části okna je umístěn panel „Konfigurační soubory“. Ten obsahuje editační pole s úplnými názvy souborů s jednotlivými konfiguračními nastaveními systému. Tyto soubory lze zvolit stiskem tlačítka „...“ a následným výběrem souboru ve standardním otevíracím dialogu systému Windows, nebo jsou tyto pole vyplněna automaticky při zvolení položky „Nastavit jako výchozí“ v menu patřičných oken aplikace. Formát jednotlivých souborů je volen podle požadavků na data v něm uložených a každý ze souborů náležících systému má vlastní koncovku složenou z písmen „rt“ doplněných číslem pro rozlišení jednotlivých souborů.

V pravé části je pole pro výběr komunikačního COM portu. Tento port může být změněn v případě přepojení kabelu i za běhu systému. I když sériový port není „plug and play“, systém toto přepojení podporuje. Následkem je pouze výpadek komunikace po dobu přepojování.

Okno dále umožňuje změnu hesla administrátora. K tomu je určen panel v pravé horní části okna a změna hesla se provádí standardně zadáním starého hesla, poté dvojím zadáním nového hesla a potvrzením tlačítkem „OK“. Heslo je standardně zobrazováno zástupnými znaky „*“, avšak pokud si uživatel není jistý jeho správným zadáním, může jej zobrazit zatržením pole „Zobrazit znaky“.

V pravé dolní části okna je umístěn panel s nastavením chování systému po spuštění. Pole „Automatické načtení konfigurace“ a „Automatické spuštění systému“ povolují funkce popsané v kapitole 4.4.9, přičemž funkce automatického spuštění nelze nastavit bez automatického načtení konfigurace. Další dvě položky se vztahují k ukládání hodnot kanálů do logovacího souboru. Zatržením pole „Ukládat naměřená data“ se spustí funkce logování dat a zatržením pole „Měsíčně automaticky nastavit soubor“ se nastaví přepínání logovacího souboru popsané v kapitole 4.4.6.

Veškeré volby nastavené v tomto okně jsou uloženy spolu s informacemi o velikostech a pozicích jednotlivých oken v souboru „RegTherm.ini“ umístěném v kořenovém adresáři aplikace. Tento soubor je nezbytně nutný ke spuštění a funkci aplikace.

5. NASTAVENÍ REGULÁTORŮ

Všechny předcházející kapitoly se věnovaly provedení systému jako regulačního nástroje. Byla vysvětlena filozofie návrhu systému a jeho provedení, byly uvedeny nedostatky, které se objevily až následkem používání systému, a byla navržena zlepšení, která budou do systému dále přidávána. Tato kapitola se ale zabývá regulací ve smyslu nastavení regulátorů a toho se týkajících praktických poznatků ze stávajících instalací systému.

5.1 DISKRÉTNÍ AKČNÍ ZÁSAHY

Regulátor použitý v rámci tohoto systému a popsany v kapitole 4.4.1 provádí výpočty s proměnnými datového typu „double“, což je typ dat s plovoucí desetinnou čárkou. Stav tohoto regulátoru i jeho výstup mohou být tedy vzhledem k rozlišení tohoto typu považovány za spojité. (Myšleno v ose výchylky, nikoliv v čase.) Pojem „spojitě“ bude v rámci této kapitoly chápán právě ve směru osy výchylky. Protože se ale k vytápění používají zpravidla akční členy s ovládáním zapnuto/vypnuto nebo s nastavením dodávaného výkonu pouze v několika diskrétních krocích, nelze naplno využít možností tohoto regulátoru. Ve stávajících instalacích byly coby akční členy použity plynové nebo elektrické kotle, plynové zářiče, motory ventilátorů a čerpadel a servomotory pro nastavování polohy směšovacích ventilů a vzduchotechnických klapek. Z tohoto výčtu lze spojitě zadávat pouze polohy servomotorů. Ovšem z hlediska jejich životnosti je žádoucí měnit jejich polohu co možná nejméně, což je navíc podpořeno faktem, že ve stávajících systémech bylo ověřeno, že není nutné nastavovat polohy jednotlivých ventilů a klapek s velkou přesností a stačí nastavení pouze ve 4 – 5 krocích. Tímto dochází k situaci, kdy je z regulátoru využit pouze rozdílový člen a jedno z výstupních relé. V případě některých smyček je výstup sice tvořen pomocí relé, ale tomu je předřazen PS regulátor za účelem omezení regulační odchylky.

5.2 PORUCHY A OMEZENÍ

V obou dosavadních případech nasazení systému je regulována teplota v budovách středních rozměrů, přičemž v jednom z nich se jedná o autoservis. Na základě poznatků o těchto objektech, tedy regulovaných soustavách, jsou největším problémem při nastavení regulačních algoritmů okolní podmínky a provozní omezení. Obzvláště nasazení regulačního systému ve zmíněném autoservisu je komplikované. Jedná se o regulaci poměrně rozlehlých prostor, kde není příliš kvalitní tepelná izolace. Dále dochází k častému otevírání vrat autoservisu, takže ochlazování regulovaných prostor je poměrně velké. Protipólem toho je místnost showroom, která je kompletně prosklená a během slunečných dnů se silně ohřívá právě slunečním zářením. Dalším problémem je malý výkon instalovaných plynových kotlů. Pokud teplota klesne pod bod mrazu, tyto kotle nemají dostatečný výkon k tomu, aby vytopily všechny prostory. Malý výkon těchto kotlů vede navíc k tomu, že přechod z nočního na denní režim je poměrně pomalý a vytápění musí začít s velkým předstihem před začátkem pracovní doby. Proto jsou v některých místnostech nainstalovány plynové zářiče, které se spínají při poklesu vnější teploty pod 0 °C a pomáhají dosáhnout požadované teploty. V době sepnutí těchto zářičů ale dochází k uzavření ventilů topných těles v těchto místnostech, aby byl k dispozici dostatek výkonu pro zbývající prostory v objektu. Použitý regulátor je tedy nutné nastavit podle těchto podmínek využitím jeho binárních vstupů popsanych v kapitole 4.4.1.5. Některé z podmínek musí být navíc vypočítávány použitím regulátoru jako univerzální výpočetní jednotky (viz kapitola 4.4.1.6).

Ve druhém případě nasazení systému se jedná o prostory firmy zabývající se vývojem a výrobou elektronických zařízení. Zde není situace tak komplikovaná jako ve zmíněné budově autoservisu. Avšak v místnosti automatické výroby se nachází dvě osazovací linky, z nichž každá má výkon přibližně 25 kW. Jelikož je celá budova vytápěna elektrickými kotli o celkovém výkonu 30 kW, z regulačního hlediska se jedná o nezanedbatelnou hodnotu vstupující poruchy.

Z výše popsaných důvodů se ale poněkud mění pohled na nastavení optimálních parametrů regulátoru. Namísto hledisek rychlosti a kvality přechodového děje, překmitu a minimální regulační odchylky vystupuje jednoznačně

priorita takového nastavení regulátorů, které bude co možná nejrobustnější a bude fungovat i při omezeních a změnách okolních podmínek. Jedná se v podstatě o takový kompromis nastavení, aby vlastnosti regulátorů byly pokud možno co nejstálější a toto nastavení nebylo provedeno s ohledem na jeden „pracovní bod“ na úkor ostatních.

5.3 ADAPTIVITA A IDENTIFIKACE

Původním záměrem bylo doplnit systém i nějakým autoadaptačním mechanismem. Mělo by se jednat o algoritmus, který provede identifikaci regulovaných prostor a na jejím základě vypočte vhodné parametry regulátorů. Tato činnost by byla prováděna průběžně a bylo by tak možné reagovat na změny v regulovaných soustavách a jejich okolí. Ovšem vzhledem k problémům popsáním v kapitole 5.2 se implementace takového mechanismu jeví jako velmi komplikovaná. Navíc je vhodné provádět identifikaci systému v okamžiku změny žádané hodnoty jeho výstupu, což ale v případě teploty regulované na konstantní hodnotu lze obtížně provést. Bylo by sice možné provádět identifikaci v okamžiku přechodu z nočního na denní režim, ale to se děje ještě před začátkem pracovní doby a tudíž jsou zařízení, která do systému vnášejí poruchy, většinou vypnutá.

Proto bylo prozatím od implementace identifikačních a autoadaptačních mechanismů upuštěno a jejich případné doplnění bude záležet na konkrétních podmínkách a požadavcích dalších použití systému.

6. ZÁVĚR

V první fázi vývoje regulačního systému jsem navrhnul jeho hardwarovou část. Jednalo se o silové a zobrazovací moduly. Jejich funkčnost jsem testoval pomocí ladicích verzí programů. Poté jsem během druhé fáze vyvinul komunikační protokol umožňující přenos dat. Po odladění protokolu bylo nutné vyvinout funkce nahrávání paměti programu procesorů jednotlivých modulů z řídicího počítače. Tato fáze byla nejnáročnější, protože bylo nutné zvládnout mechanismus nahrávání „sebe sama“ mikroprocesorů AVR a navíc se ještě objevovaly některé chyby a nedostatky přenosového protokolu, které nebyly při jeho ladění patrné. Během čtvrté fáze jsem vytvořil systém jako celek schopný přijímat, zpracovávat, uchovávat a odesílat signály. Po dokončení této fáze byl systém k dispozici jako regulační nástroj. Pak už jsem musel jen nastavit regulační algoritmy podle soustav, na něž byl systém nasazen. To by se dalo označit pátou fází vývoje, která se ale opakuje při každé další instalaci systému.

Vývoj systému jsem zde sice chronologicky rozdělil do pěti základních částí, ale ve skutečnosti jsem jednotlivé části upravoval a doplňoval již v průběhu vývoje dalších částí, ať již z důvodu oprav chyb nebo doplnění funkcí či vlastností. I tak ale podobu systému nepovažuji za finální a předpokládám jeho další úpravy.

Nejvýraznějším rysem navrženého systému je použití PC coby řídicího prvku. Toto řešení jsem zvolil z důvodu snadné vizualizace a ovládání procesu, jednoduché dostupnosti, velkého výpočetního výkonu a snadného připojení k Internetu, který může být využit k dálkovému ovládání systému.

Z hlediska uživatele je velmi výhodná možnost upgrade softwaru v modulech přes standardní komunikační sběrnici. Společně s připojením počítače k Internetu to umožňuje modifikovat software v modulech bez přítomnosti správce systému v místě instalace, což je výhodné i v případě výměny modulu, která může být provedena elektromechanikem a systémový správce provede softwarovou rekonfiguraci na dálku. Tato vlastnost podstatně snižuje cenu úprav a oprav.

Regulační část systému jsem navrhnul se snahou o maximální univerzálnost. To může být v některých případech výhodou, jindy překážkou. Ve stávajících

instalacích ale bylo možné pomocí navržených nástrojů vyřešit všechny úkoly a splnit požadavky zákazníka. Pokud to bude nutné, je možné jednoduše systém doplnit o další funkční bloky jak hardwarové, tak i softwarové.

Samostatnou částí je zajištění vyšší míry bezpečnosti nebo dokonce vyšší míry funkčnosti systému. V rámci této práce jsem navrhnul některá možná řešení. Vždy ale záleží na konkrétních požadavcích zadavatele a podmínkách aplikace systému a tato zabezpečení vždy zvyšují cenu systému. Jejich budoucí realizace tedy bude záviset na komerční úspěšnosti systému.

Navržený systém je v současné době provozován ve dvou budovách, kde prokázal svoji funkčnost. Ovšem je samozřejmé, že bude dále doplňován podle potřeb a přání zákazníků. Porovnání vlastností systému s konkurenčními produkty není zcela možné, protože výrobci těchto zařízení neposkytují příliš detailní informace o jejich provedení. Jedinou relevantní dostupnou informací je fakt, že si zmiňovaný autoservis nechal vytvořit nabídku na regulační systém od více firem a zde popsaný systém byl nakonec nainstalován v koncové ceně pro zákazníka o 50 % levněji než nejlevnější konkurenční systém. Je nutné podotknout, že sice bylo využito stávajících rozvodů, ale ani při jejich výměně by cena nebyla dvojnásobná. Největší výhodou tohoto systému je tedy bezpochyby jeho flexibilita a z ní vyplývající nízká cena.

Tento projekt svým rozsahem přesahuje běžný rozsah diplomové práce díky tomu, že byl vytvořen na zakázku s předem známým místem použití. Proto jsem se mu věnoval i ve svém volném čase v rozsahu mnohonásobně vyšším, než který je nutný k vypracování projektu pro školní účely.

7. SEZNAM LITERATURY

- [1] ZÁHLAVA, V. *OrCAD 10*, První vydání, Praha, Grada Publishing, 2004. 224s. ISBN 80-247-0904-X
- [2] ATMEL CORPORATION. *ATmega8, ATmega8L*, [on-line]. Poslední aktualizace: 2/2006. [cit. 14.4.2007]. Dostupný z: <www.atmel.com>
- [3] ATMEL CORPORATION. *ATmega325/V, ATmega3250/V, ATmega645/V, ATmega6450/V*, [on-line]. Poslední aktualizace: 5/2005. [cit. 14.4.2007]. Dostupný z: <www.atmel.com>
- [4] DALLAS SEMICONDUCTOR CORPORATION. *DS18S20*, [on-line]. [cit. 14.4.2007]. Dostupný z: <www.maxim-ic.com>
- [5] FAIRCHILD SEMICONDUCTOR CORPORATION. *MOC3041, MOC3042, MOC304*, [on-line]. [cit. 14.4.2007]. Dostupný z: <www.fairchildsemi.com>
- [6] PHILIPS SEMICONDUCTOR. *BT138*, [on-line]. [cit. 14.4.2007]. Dostupný z: <www.alldatasheet.com>
- [7] SHARP. *PC357*, [on-line]. [cit. 14.4.2007]. Dostupný z: <www.alldatasheet.com>
- [8] BELIMO AUTOMATION AG. *LMC24-SR-F*, [on-line]. [cit. 14.4.2007]. Dostupný z: <www.belimo.com>
- [9] STMICROELECTRONICS. *MC34063A, MC34063E*, [on-line]. [cit. 14.4.2007]. Dostupný z: <www.st.com>
- [10] TEXAS INSTRUMENTS. *Sn65hvd3082e*, [on-line]. Poslední aktualizace: 9/2005. [cit. 14.4.2007]. Dostupný z: <www.ti.com>
- [11] DALLAS SEMICONDUCTOR CORPORATION. *DS1631/DS1631A/DS1731*, [on-line]. [cit. 14.4.2007]. Dostupný z: <www.maxim-ic.com>
- [12] REGULUS. *INTELIGENTNÍ REGULÁTOR REGULUS IR09 TC A IR09 KTP*, [on-line]. [cit. 17.4.2007]. Dostupný z: <http://www.regulus.cz/cz/Regulatory-IR09.asp?ID=33>
- [13] REGULUS. *INTELIGENTNÍ REGULÁTOR ELESTA*, [on-line]. [cit. 17.4.2007]. Dostupný z: <http://www.regulus.cz/cz/Regulatory-Elesta.asp?ID=33>

- [14] HONEYWELL. *EXCEL 5000*, [on-line]. [cit. 17.4.2007]. Dostupný z:
<<http://www.honeywell.cz/acs/building/produkty/index.asp?id=DDC&mnu=XL5000>>
- [15] MOELLER. *Radiofrekvenční systém Xcomfort*, [on-line]. [cit. 17.4.2007].
Dostupný z: <http://www.moeller.cz/produkty-domovni_instalace-system_xcomfort-radiofrekvencni_system>
- [16] MOELLER. *Sběrnicový systém NIKOBUS*, [on-line]. [cit. 17.4.2007].
Dostupný z: <http://www.moeller.cz/produkty-domovni_instalace-system_xcomfort-nikobus>
- [17] PRATA, S. *Mistrovství v C++*, Druhé aktualizované vydání, Brno, Computer Press, 2004. ISBN 80-251-0098-7
- [18] KADLEC, V. *Učíme se programovat v jazyce C*, Vydání první, Praha, Computer Press, 2002. 277s. ISBN 80-7226-715-9
- [19] VÁŇA, V. *Mikrokontroléry Atmel AVR – programování v jazyce C*, První vydání, Praha, BEN, 2003, 216s. ISBN 80-7300-102-0
- [20] PIVOŇKA, P. *Optimalizace regulátorů*, [on-line], <<http://www.feec.vutbr.cz>>, 2005-12-10
- [21] PIVOŇKA, P. *Číslicová řídicí technika*, [on-line], <<http://www.feec.vutbr.cz>>, 2003-11-1
- [22] ASTRÖM, K. J., HÄGGLUND, T. *Benchmark systems for PID kontrol*, [on-line], <<ftp://dzin.feec.vutbr.cz>>
- [23] VELEBA, V. *Číslicová řídicí technika – Počítačová cvičení*, [on-line], <<ftp://dzin.feec.vutbr.cz>>, 2005-10-1

8. SEZNAM ZKRATEK

WORD	dvoubytová datová jednotka
I	vstup
O	výstup
I/O	vstup/výstup
CRC	kontrolní součet
SISO	regulační smyčka s jedním vstupem a jedním výstupem
MISO	regulační smyčka s více než jedním vstupem a jedním výstupem
HDO	hromadné dálkové ovládání

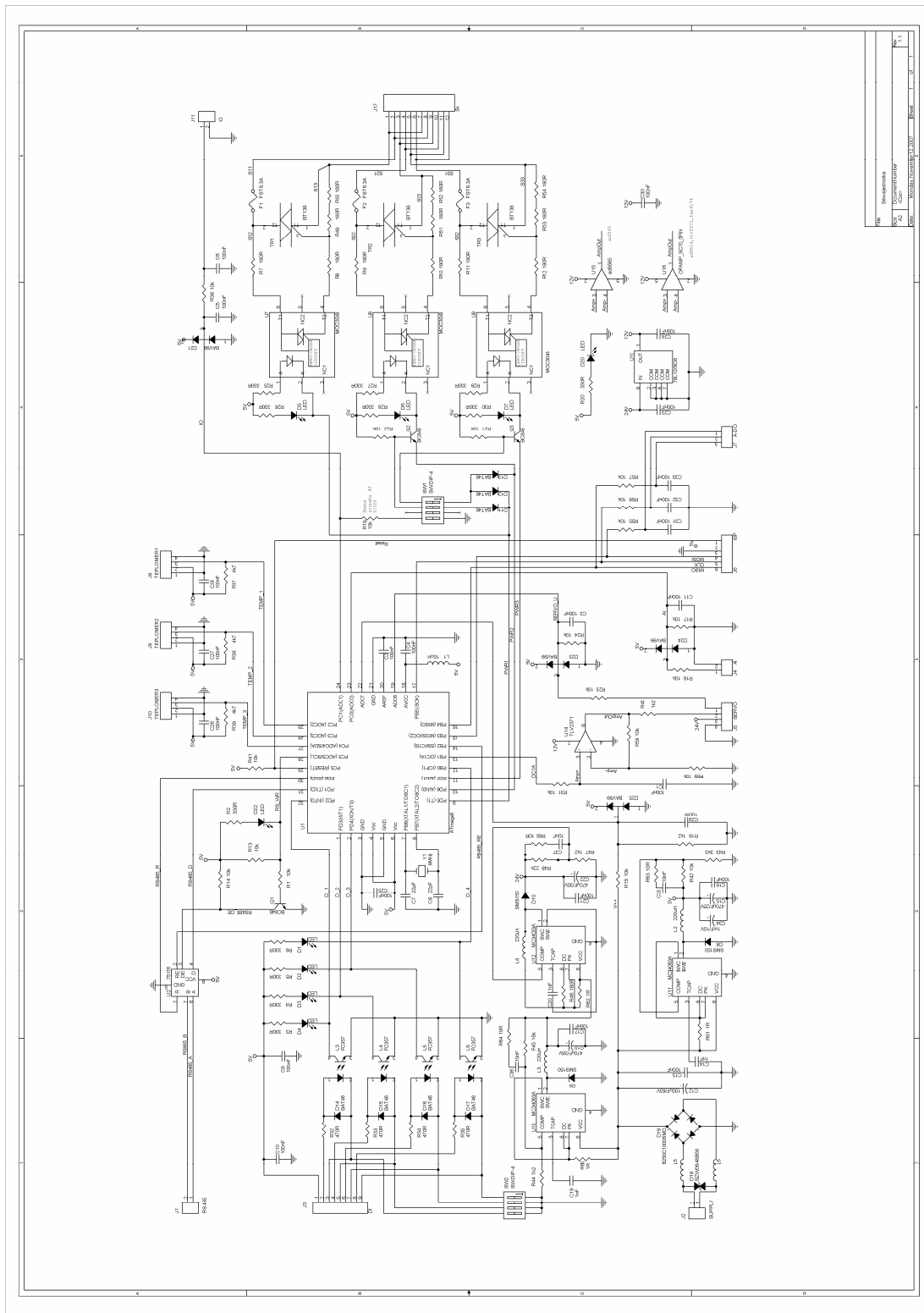
9. PŘÍLOHY

Seznam příloh

- | | |
|-----------|----------------------|
| Příloha 1 | Schémata |
| Příloha 2 | Desky plošných spojů |
| Příloha 3 | Funkce třídy cSystem |

Obrázek 9.1: Schéma zobrazovací jednotky

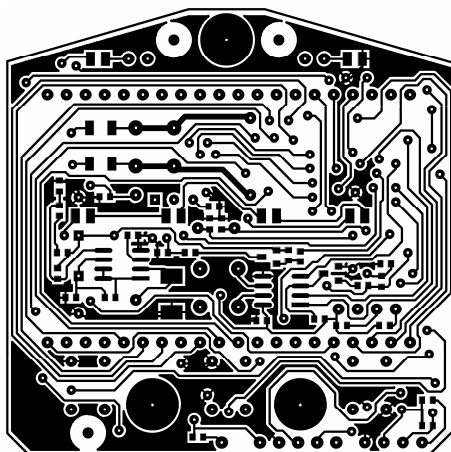
9.1.2 Silová jednotka



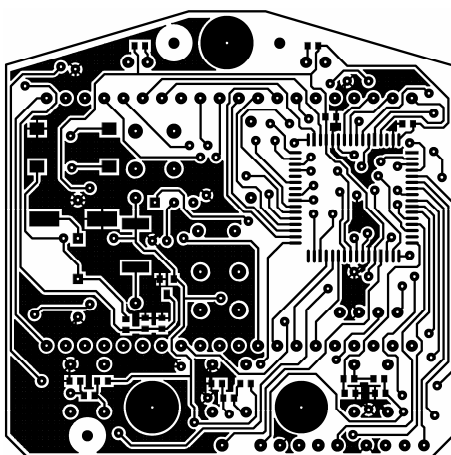
Obrázek 9.2: Schéma silové jednotky

9.2 DESKY PLOŠNÝCH SPOJŮ

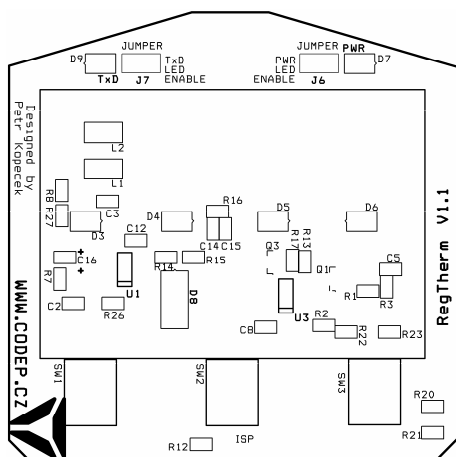
9.2.1 Zobrazovací jednotka



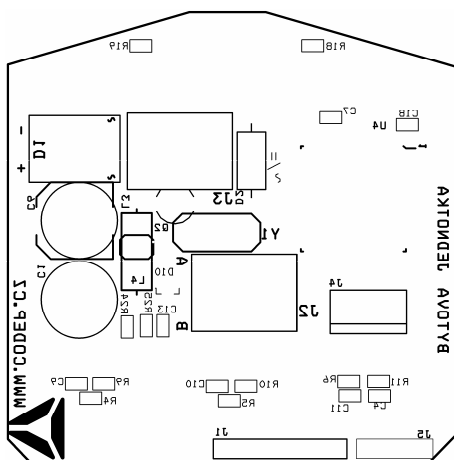
Obrázek 9.3: Zobrazovací jednotka - spoje horní vrstva



Obrázek 9.4: Zobrazovací jednotka - spoje spodní vrstva



Obrázek 9.5: Zobrazovací jednotka - servisní potisk horní vrstva

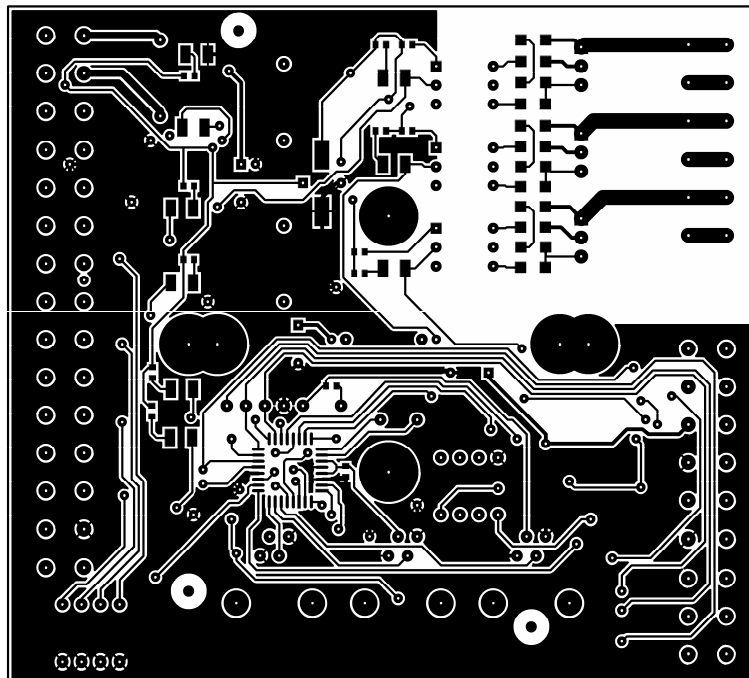


Obrázek 9.6: Zobrazovací jednotka - servisní potisk spodní vrstva

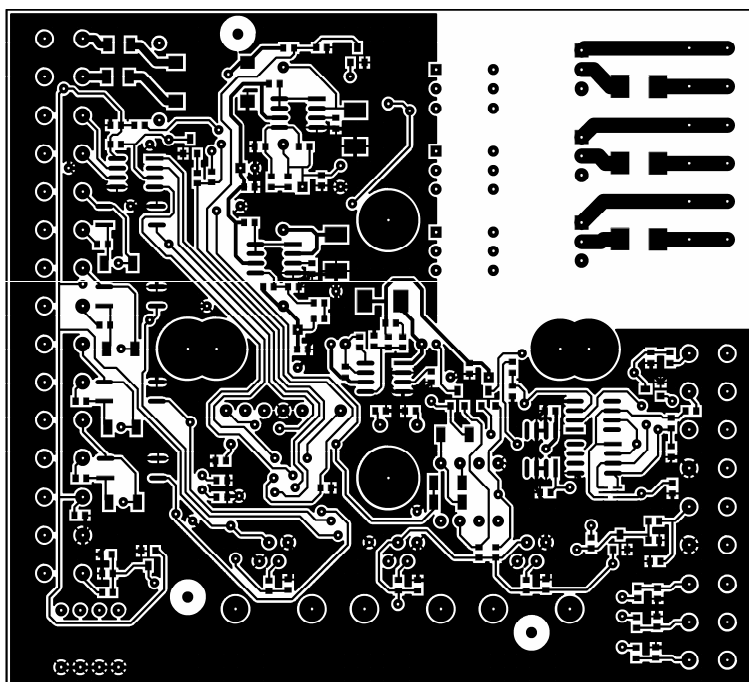


Obrázek 9.7: Senzor

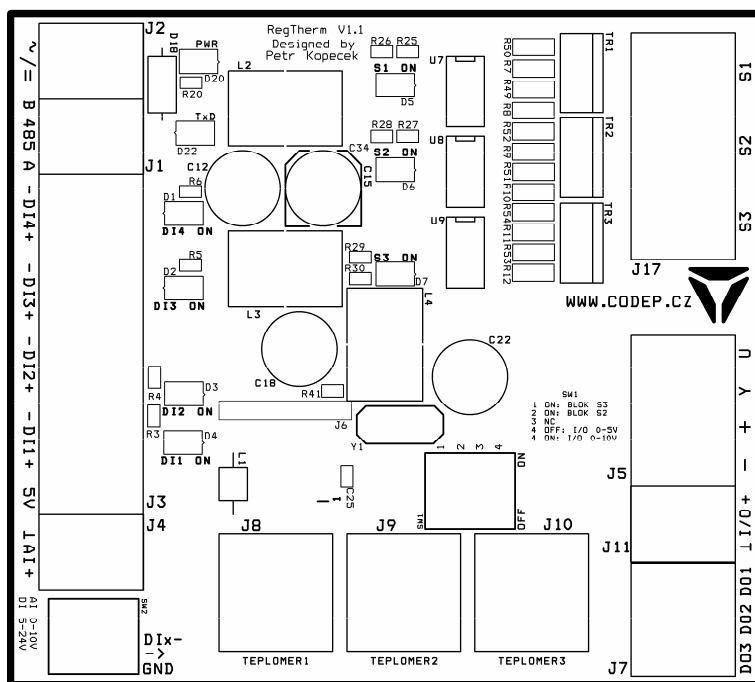
9.2.2 Silová jednotka



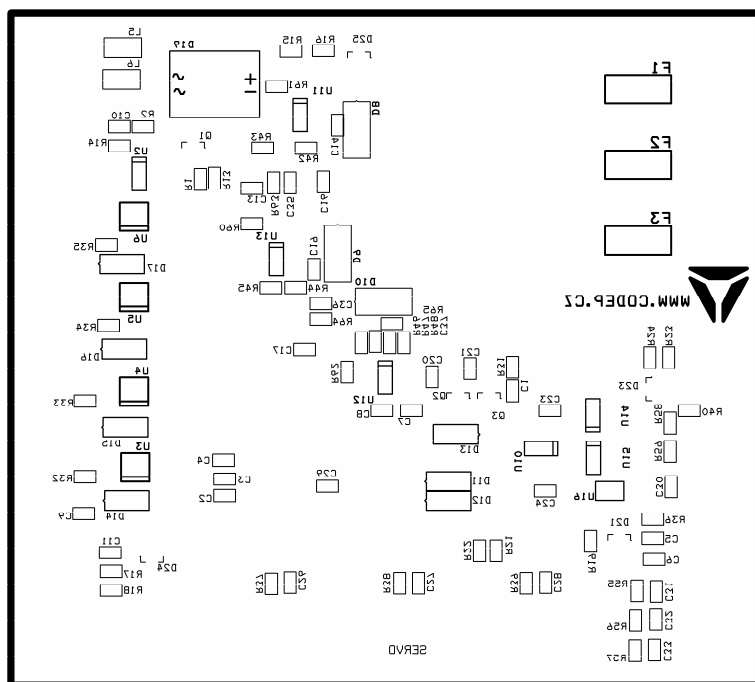
Obrázek 9.8: Silová jednotka - spoje horní vrstva



Obrázek 9.9: Silová jednotka - spoje spodní vrstva



Obrázek 9.10: Silová jednotka - servisní potisk horní vrstva



Obrázek 9.11: Silová jednotka - servisní potisk spodní vrstva

9.3 FUNKCE TŘÍDY CSYSTEM

- *bool **SetCom**(unsigned int uiPortNo);*

Nastaví komunikační port COM1 – COM9. Port je určen parametrem *uiPortNo* (0~COM1, 8~COM9).

- *bool **OpenPort**(void) const;*

Otevře komunikační port. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **ClosePort**(void) const;*

Uzavře komunikační port. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *int **GetComState**(void) const;*

Načte stav portu otevřen/uzavřen.

- *DWORD **GetBaudRate**(void) const;*

Načte nastavenou přenosovou rychlost.

- *char* **GetPortName**(void) const;*

Načte název komunikačního portu.

- *bool **SetCom**(cRS485regtherm *ReceivedPort);*

Nastaví členskou instanci třídy *cRS485regtherm* *port podle přijatého parametru. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *WORD **GetVirtualAdr**(const WORD wIndex) const;*

Vrací virtuální adresu modulu podle indexu *wIndex* z tabulky adres

- *bool **SaveToFile**(const AnsiString asPath);*

Uloží systémovou konfiguraci do souboru. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **LoadFromFile**(const AnsiString asPath);*

Načte systémovou konfiguraci ze souboru. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **ScanBus**(void);*

Provede skenování sběrnice, tj. zjistí počet zařízení připojených ke sběrnici a načte jejich sériová čísla. Návrátová hodnota TRUE značí úspěšně provedenou operaci, FALSE značí chybu při vykonávání.

- *bool **ProgSel**(const WORD index, const AnsiString asServoSwPath, const AnsiString asRoomUnitSwPath, const AnsiString asThirdSwPath, const AnsiString asFourthSwPath);*

Naprogramuje vybraný modul. Funkci se předává index modulu určující jeho pozici v tabulce modulů a cesty ke všem binárním souborům s aplikačním softwarem. Prozatím se využívá pouze dvou typů souborů, třetí a čtvrtý parametr je ignorován.

- *bool **ProgAll**(void);*

Naprogramuje všechny načtené moduly.

- *int **CheckStatusSel**(const WORD wIndex);*

Načte stav software modulu (aplikace, loader, neodpovídá). Modul je určen předaným indexem z tabulky modulů.

- *bool **RunAppAll**(void);*

Spustí aplikační software ve všech modulech. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **RunAppSel**(const WORD wIndex);*

Spustí aplikační software ve vybraném modulu. Modul je určen předaným indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **RunServisAll**(void);*

Spustí servisní software (loader) ve všech modulech. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **RunServisSel**(const WORD wIndex);*

Spustí servisní software (loader) ve vybraném modulu. Modul je určen předaným indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *WORD **GetDeviceCount**(void) const;*
Vrátí počet načtených zařízení.
- *WORD **GetServoCount**(void) const;*
Vrátí počet načtených silových modulů.
- *WORD **GetRoomUnitCount**(void) const;*
Vrátí počet načtených zobrazovacích modulů.
- *WORD **GetUnusedServoCount**(void) const;*
Vrátí počet načtených silových modulů, které byly načteny ze souboru, ale nejsou detekovány na sběrnici.
- *WORD **GetUnusedRoomUnitCount**(void) const;*
Vrátí počet načtených zobrazovacích modulů, které byly načteny ze souboru, ale nejsou detekovány na sběrnici.
- *WORD **GetSerialNumber**(const WORD wIndex) const;*
Vrátí sériové číslo vybraného modulu. Modul je určen předaným indexem z tabulky modulů.
- *bool **GetSerialNumberTable**(WORD *wTable, WORD &wCount) const;*
Načte tabulku sériových čísel do proměnné dané ukazatelem *wTable* a jejich počet do proměnné *wCount*. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.
- *bool **GetServoNames**(const WORD wIndex, AnsiString *asNames, const WORD wSize) const;*
Načte uživatelské názvy kanálů silového modulu do pole určeného ukazatelem *asNames*. Počet načítaných názvů určuje parametr *wSize*. Modul je určen předaným indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.
- *bool **GetRoomUnitNames**(const WORD wIndex, AnsiString *asNames, WORD wSize) const;*
Načte uživatelské názvy kanálů zobrazovacího modulu do pole určeného ukazatelem *asNames*. Počet načítaných názvů určuje parametr *wSize*. Modul je určen předaným indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *AnsiString **GetName**(const WORD wVirtualAdr, bool &bStatus) const;*

Vrátí uživatelský název kanálu určeného virtuální adresou v *wVirtualAdr*. Úspěšnost operace vyjádřená hodnotou typu bool se ukládá do proměnné *bStatus*.

- *bool **SetServoNames**(const WORD wIndex, AnsiString *asNames);*

Nastaví uživatelské názvy kanálů silového modulu podle dat pole určeného ukazatelem *asNames*. Modul je určen předaným indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **SetRoomUnitNames**(const WORD wIndex, AnsiString *asNames) ;*

Nastaví uživatelské názvy kanálů zobrazovacího modulu podle dat pole určeného ukazatelem *asNames*. Modul je určen předaným indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **SetVirtualAdrTable**(AnsiString *asVirtualAdr, AnsiString *asNts, WORD & wErr);*

Vyplní tabulku virtuálních čísel a uživatelských popisů modulů daty předaných poli *asVirtualAdr* a *asNts*. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool, upřesnění chyby je uloženo v proměnné *wErr*.

- *bool **GetVirtualAdrTable**(WORD *wVirtualAdr, AnsiString *asNts) const;*

Načte tabulku virtuálních čísel a uživatelských popisů modulů do polí *asVirtualAdr* a *asNts*.

- *AnsiString **GetNote**(WORD wIndex) const;*

Vrátí uživatelský popis modulu určeného indexem z tabulky modulů.

- *double **GetData**(const WORD wVirtualAdr, const WORD wIndex, bool &bStatus) const;*

Vrátí data kanálu určeného virtuální adresou *wVirtualAdr*. Parametr *wIndex* určuje index z historie. Do proměnné *bStatus* se ukládá úspěšnost operace.

- *bool **GetDataHistory**(const WORD wVirtualAdr, double *dfData, bool &bStatus) const;*

Načte celou historii dat kanálu určeného virtuální adresou *wVirtualAdr* do pole *dfData*. Do proměnné *bStatus* se ukládá úspěšnost operace.

- *bool **SetData**(const WORD wVirtualAdr, double dfData) const;*

Nastaví data kanálu určeného virtuální adresou *wVirtualAdr* na hodnotu *dfData*. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **DeleteData**(const WORD wVirtualAdr) const;*

Vymaže historii dat kanálu určeného virtuální adresou *wVirtualAdr*. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **DeleteData**(void) const;*

Vymaže historii dat všech kanálů v systému. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **ReadAllInputChannels**(void) const;*

Načte hodnoty všech vstupních kanálů z modulů do řídicího počítače. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **WriteAllOutputChannels**(void) const;*

Zapíše hodnoty všech výstupních kanálů z řídicího počítače do modulů. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **SetWValueName**(const WORD wVirtualAdr, const AnsiString asName) const;*

Nastaví název kanálu konstanty nebo propojovací proměnné dané virtuální adresou *wVirtualAdr* na hodnotu *asName*. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **AddWValue**(void);*

Přidá konstantu. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **DeleteWValue**(const WORD wIndex);*

Smaže konstantu určenou indexem z tabulky modulů. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **NewWValues**(void);*

Smaže stávající konstanty a vytvoří nové pole konstant.

- *bool **SetWValues**(const WORD wIndex, AnsiString asName, const double dfDay, const double dfNight);*

Nastaví název konstanty určené indexem z tabulky modulů na hodnotu *asName* a hodnoty konstant na hodnotu *dfDay* pro denní režim a *dfNight* pro noční režim.

- *bool **GetWValues**(const WORD wIndex, AnsiString &asName, double &dfDay, double &dfNight) const;*

Načte název konstanty určené indexem z tabulky modulů do proměnné *asName* a hodnoty konstant pro denní režim do proměnné *dfDay* a pro noční režim do proměnné *dfNight*.

- *WORD **GetWValueCount**(void) const;*

Vrátí počet konstant a propojovacích proměnných.

- *bool **SaveWValues**(const AnsiString asPath) const;*

Uloží konstanty do souboru. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **LoadWValues**(const AnsiString asPath);*

Načte konstanty ze souboru. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **SaveAllValues**(void) const;*

Uloží naměřené hodnoty do logovacího souboru. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *void **SaveNamesToFile**(void) const;*

Uloží názvy kanálů do logovacího souboru.

- *void **NewDataLogFile**(const AnsiString asPath);*

Vytvoří nový logovací soubor na základě cesty *asPath*.

- *void **SetDataLogFile**(const AnsiString asPath);*

Načte logovací soubor na základě cesty *asPath* a nastaví jej jako aktivní.

- *void **SetSystemFilePath**(const AnsiString asPath);*

Nastaví hlavní systémovou cestu na základě cesty *asPath*.

- *AnsiString **GetDataFilePath**(void) const;*
Vrátí hlavní systémovou cestu.
- *bool **SetSafetyValues**(const WORD wIndex, short int *siSafetyValues) const;*
Nastaví bezpečné hodnoty výstupů modulu podle dat v poli *siSafetyValues*. Modul je určen indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.
- *bool **GetSafetyValues**(const WORD wIndex, short int *siSafetyValues) const;*
Načte bezpečné hodnoty výstupů modulu do pole *siSafetyValues*. Modul je určen indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.
- *bool **SetOnBoardSafetyValues**(const WORD wIndex) const;*
Odešle bezpečné hodnoty výstupů do modulu určeného indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.
- *bool **GetOnBoardSafetyValues**(const WORD wIndex) const;*
Přijme bezpečné hodnoty výstupů do modulu určeného indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.
- *int **GetTimeOut**(const WORD wVirtualAdr, bool &bStatus) const;*
Vrátí timeout modulu určeného virtuální adresou *wVirtualAdr*. Do proměnné *bStatus* se ukládá úspěšnost operace.
- *bool **Copy**(const WORD wIndex);*
Kopíruje nastavení modulu určeného indexem z tabulky modulů do vnitřní schránky systému. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.
- *bool **Paste**(const WORD wIndex);*
Vkládá nastavení modulu z vnitřní schránky systému do modulu určeného indexem z tabulky modulů. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.
- *bool **DeleteDevice**(const WORD wIndex);*
Smaže modul určený indexem z tabulky modulů ze systému. Návrátovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **SetServo**(const cServoData &Servo, const WORD wIndex);*

Kopíruje nastavení silového modulu předaného parametrem *Servo* do modulu určeného indexem z tabulky modulů. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **GetServo**(cServoData &Servo, const WORD wIndex) const;*

Kopíruje nastavení silového modulu určeného indexem z tabulky modulů do modulu předaného parametrem *Servo*. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **SetRoomUnit**(const cRoomUnitData &RoomUnit, const WORD wIndex);*

Kopíruje nastavení zobrazovacího modulu předaného parametrem *Servo* do modulu určeného indexem z tabulky modulů. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **GetRoomUnit**(cRoomUnitData &RoomUnit, const WORD wIndex) const;*

Kopíruje nastavení zobrazovacího modulu určeného indexem z tabulky modulů do modulu předaného parametrem *Servo*. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool **LoadChannelsFromFile**(AnsiString asFile);*

Načte nastavení kanálů ze souboru určeného parametrem *asFile*. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *void **SetDayMode**(void);*

Nastaví denní režim.

- *void **SetNightMode**(void);*

Nastaví noční režim.

- *int **GetMode**(void) const;*

Vrátí aktivní režim.

- *bool **SetMode**(const WORD wAdr, const int iModeTemp) const;*

Provede nastavení režimů všech konstant podle řídicích kanálů, pokud jsou nastaveny. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool SetSetValues(void);*

Odešle aktualizované žádané hodnoty do zobrazovacích modulů. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool SetWValuePwrSaveChannel(WORD wIndex, WORD wAdr, bool bInvert);*

Pro konstantu určenou indexem z tabulky modulů nastaví kanál řídicí mód na hodnotu *wAdr* a jeho polaritu na hodnotu *bInvert* . Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool GetWValuePwrSaveChannel(WORD wIndex, WORD &wAdr, bool &bInvert) const;*

Načte virtuální adresu kanálu řídicího mód konstanty určené indexem z tabulky modulů do proměnné *wAdr* a jeho polaritu do proměnné *bInvert*. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *void SetWValueMode(void);*

Provede nastavení módu jednotlivých konstant.

- *int Debug(int iParam);*

Víceúčelová funkce pro ladění systému. Její účel se upravuje podle aktuálních potřeb.

- *bool GetSwVersion(WORD wIndex, AnsiString *asLoader, AnsiString *asApp);*

Načtení verze software z modulů. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool OpenLogFile(AnsiString asFile);*

Otevře logovací soubor k interpretaci dat. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool CloseLogFile(void);*

Uzavře logovací soubor. Návratovou hodnotou je úspěšnost operace vyjádřená hodnotou typu bool.

- *bool IsLogFileOpened(void) const;*

Poskytuje informaci, zda je logovací soubor otevřen k interpretaci dat.