

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
(Fakulta elektrotechnická / Katedra řídicí techniky)



AK. R. 2007/08

BAKALÁRSKA PRÁCA

**SADA PRO DLOUHODOBÉ MĚŘENÍ A ZVIDITELNĚNÍ
PROCESŮ VE VYTÁPĚNÝCH PROSTORECH**

autor: MICHAL ANDREJCO
vedúci práce: Ing. Martin Hlinovský, Ph. D.
oponent: Ing. Ladislav Šmejkal, CSc.

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra řídicí techniky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Michal Andrejco**

Studijní program: Elektrotechnika a informatika (bakalářský), strukturovaný
Obor: Kybernetika a měření

Název tématu: **Sada pro dlouhodobé měření a zviditelnění procesů ve vytápěných prostorech**

Pokyny pro vypracování:

Cílem je navrhnout a realizovat měřicí sadu pro dlouhodobé měření a názorné zviditelnění procesů v regulovaných tepelných soustavách, která je vhodná pro potřeby výuky a identifikace.

- seznamte s PLC systémy Tecomat, distribuovaným instalačním systémem Inels 2, s vývojovým systémem Mosaic a SCADA systémem Reliance
- analyzujte tepelné procesy v technice budov a zvolte veličiny, které je účelné sledovat,
- seznamte se se zásadami instrumentace procesů s těmito systémy a navrhnete uspořádání sady pro měření s dlouhodobé sledování zvolených veličin,
- realizujte program PLC pro měření, popř. pro archivaci a názornou grafickou prezentaci,
- realizujte a ověřte na reálné soustavě (alespoň částečně) a popř. zpracujte metodickou dokumentaci.

Seznam odborné literatury:

Dodá vedoucí práce

Vedoucí: Ing. Martin Hlinovský, Ph.D.

Platnost zadání: do konce zimního semestru 2008/2009

prof. Ing. Michael Šebek, DrSc.
vedoucí katedry



doc. Ing. Boris Šimák, CSc.
děkan

V Praze dne 25. 2. 2008

Abstrakt

Cieľom tejto bakalárskej práce je zoznámiť sa s riadiacimi systémami FOXTROT a softvérovými prostriedkami pre ich programovanie a poznatky spracovať ako metodickú dokumentáciu ktorá bude využívaná ako návod pre zoznámenie sa so softvérom a PLC systémami. Na základe týchto znalostí vytvoriť kompaktný monitorovací systém teplôt pre vytápané priestory, ktorý bude schopný dlhodobej archivácie nameraných dát, a ich spätnej interpretácie prostredníctvom časového priebehu vývoja teploty v určitom mieste a v zvolenom čase. Aktuálne data by mohli byť prístupné aj z webového rozhrania.

Abstract

The aim of this bachelor thesis is to familiarize with the FOXTROT control systems and with support software for programming it. This knowledge could be processed as a methodical documentation and used for acquainting with software and PLC systems. On the basis of this knowledge the compact temperature monitoring system for the heating areas could be made. It will be able to archive the measured data for a long time and it will interpret them as a time course of temperature development at chosen time and at chosen place. The current data could be available also from web-pages.

Čestné prehlásenie

Prehlasujem, že som zadanú bakalársku prácu spracoval samostatne a použil som len podklady (literatúru, projekty, SW atd.) uvedené v priloženom zozname.

V Prahe, dňa.....

.....

podpis

Obsah

Abstrakt.....	3
Čestné prehlásenie.....	4
Úvod.....	7
1. Kapitola.....	9
1.1 Teoretická časť.....	9
1.1.1 Model tepelného systému.....	9
1.1.2 Straty v tepelných systémoch.....	10
1.1.3 Dvojstavová regulácia.....	10
2. Kapitola.....	12
2.1 PLC Systémy.....	12
2.1.1 Popis logického automatu.....	12
2.1.2 Štruktúra logického automatu.....	12
2.1.3 Programovanie logického automatu.....	13
2.2 Systémy FOXTROT CP1004.....	14
2.2.1 Hardvérové prostriedky.....	14
2.2.2 Konfigurácia systému.....	15
2.3 Inštalačné systémy INELS 2.....	16
2.3.1 Popis systémov.....	16
2.3.2 Inštalácia a konfigurácia systémov.....	17
3. Kapitola.....	18
3.1 Softvérové prostriedky.....	18
3.1.1 Popis vývojového softvéru MOSAIC.....	18
3.1.2 Popis softvéru RELIANCE 4.....	18
3.1.3 Nastavenie komunikácie s PLC v RELIANCE.....	19
4. Kapitola.....	20
4.1 Monitorovací systém.....	20
4.1.1 Voľba veličín k sledovaniu.....	20
4.1.2 Umiestnenie senzorov.....	22
5. Kapitola.....	23
5.1 Návrh riadiaceho programu pre PLC.....	23
5.1.1 Dekompozícia problému.....	23
5.1.2 Funkcie využité v programe.....	25
5.1.3 Popis funkcií využitých v programe.....	27
5.1.4 Vývojový diagram hlavného algoritmu PLC.....	29
5.2 Návrh riadiaceho programu v RELIANCE.....	31
5.2.1 Dekompozícia problému.....	31
5.2.2 Popis hlavných skriptov.....	32
5.3 Vizualizácia.....	33
5.3.1 Popis vizualizačných okien.....	33
5.3.2 Ukážky vizualizácií.....	34
6. Kapitola.....	36
6.1 Problémové sekcie.....	36
7. Kapitola.....	38
7.1 Návrhy na vylepšenie.....	38
7.2 Záver.....	39
Prílohy.....	40
A .Pripojenie PLC k RELIANCE	40

B .Pripojenie PLC k MOSAIC	41
C .Výsledky merania.....	43
Referencie.....	45
Pod'akovanie.....	46

Úvod

Téma bakalárskej práce vychádza z nápadu praktického overenia prebiehajúcich tepelných procesov vo vytápaných priestoroch. Problém práce v sebe zahŕňa praktické znalosti z oblasti programovania riadiacich systémov PLC (*Programable Logic Controller*), znalosti prezentačných a monitorovacích systémov SCADA (*Supervisory Control And Data Acquisition*), tak ako znalosti prevádzkovania sietí a praktických znalostí jednoduchých databázových systémov. Výstupom tejto práce je kompaktný autonómny monitorovací systém riadený programovateľným automatom. Systém bude akumulovať a prezentovať aktuálne údaje o stave daného monitorovaného priestoru. Okrem toho bude schopný interpretovať už archivované data vo forme vhodnej pre ďalšie spracovanie. V tomto prípade prichádza ako najatraktívnejšie riešenie forma časového priebehu.

Celá koncepcia práce sa dá rozdeliť do viacerých kompaktných celkov. Spočíva v spojení problému monitoringu, archivácie a prezentácie. Úmyslom je, aby monitorované data boli prístupné jednak zo zdroja, na ktorom je systém inštalovaný, rovnako tak aby boli prístupné z webového rozhrania aspoň data, ktoré boli namonitorované aktuálne. S ohľadom na to, že systém by mohol byť použitý aj u koncového užívateľa pre monitoring stavu budovy (pre potreby identifikácie modelu meraného objektu s cieľom zvýšenia efektivity vytápania), je snaha navrhnúť programové prostredie efektívne a jednoducho. Na riešenie každého z vymenovaných problémov je potreba iného nástroja a zoznámenia sa s ním, aspoň v rozsahu zadania bakalárskej práce. V tomto prípade sa využívajú 2 nástroje: MOSAIC, RELIANCE 4. Pre archiváciu je využitá databáza implementovaná v SCADA systéme RELIANCE 4. Tento nástroj je následne využitý aj pre vizualizáciu celého projektu, ako lokálne, tak aj pre diaľkovú vizualizáciu pomocou JCC (*Java Control Client*). Tento klient má jednu značnú nevýhodu, a to potrebu stále zapnutého serveru na monitorovacej strane, ktorý sa nenachádza priamo v logickom automate, ale je implementovaný v nástroji RELIANCE 4. To môže byť maximálne nevýhodné pre užívateľov, ktorí PC (*Personal Computer*) alebo NTB (*NoTeBook*) vypínajú, resp. im slúži ako pracovný nástroj a majú potrebu ho využívať aj na cestách. Tento problém je riešený v nástroji MOSAIC, pomocou ktorého je navrhnuté vizualizačné prostredie pre server, ktorý sa nachádza priamo v PLC a má priamy styk s aktuálnym monitorovaným okolím.

Pre realizáciu tohoto systému bude použitý automat firmy TECO a.s. značky FOXTROT s inštaláčnymi komponentami INELS 2. Ako vývojové prostredia sú využívané už avízované vývojové a kontrolné systémy:

- RELIANCE 4 (spomínaný systém typu SCADA)
- MOSAIC (vývojové prostredie pre prácu so systémami od Teco a.s.)

Podrobnejšiemu zoznámeniu s týmito prostriedkami bude venovaná podkapitola v tejto bakalárskej práci. Okrem toho bude značná časť práce venovaná praktickým postupom a metodikám uplatneným v návrhu, taktiež ako aj praktické ukážky vizualizačných okien programu.

Na priloženom CD sú dostupné príručky ktoré slúžili na čerpanie informácií o PLC, a pre zoznámenie sa s programovacími normami.

Od celej práce sa očakáva že jej výsledky pomôžu pri presnej identifikácii vytápaných priestorov.

1. Kapitola

1.1 Teoretická časť

1.1.1 Model tepelného systému

Keďže zmyslom tejto práce nie je identifikácia systému, ale zmeranie charakteristík pre jeho presnú identifikáciu, bude v tejto podkapitole uvedený ako demonštrácia problému len zjednodušený model tepelnej sústavy. Uvažujme miestnosť v objekte, ako tepelný systém, podľa [3]. Tepelná energia prechádza cez steny miestnosti v množstve priamo úmernom rozdielu teplôt pred a za stenou, teda matematicky vyjadrené:

$$q = \frac{1}{R} \cdot (T_1 - T_2) \quad , \quad (1.1)$$

kde: q = tepelný tok [J/s], R = tepelný odpor [$^{\circ}\text{C}/\text{J}\cdot\text{s}$], T = teplota [$^{\circ}\text{C}$].

Celkový tok tepelnej energie do látky ovplyvňuje teplotu látky podľa vzťahu:

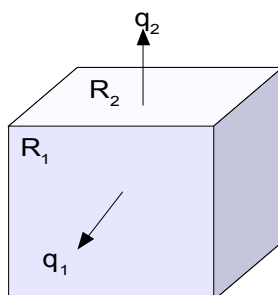
$$\dot{T} = \frac{1}{C} \cdot q \quad , \quad (1.2)$$

kde C označuje tepelnú kapacitu [-].

POZN.: Pre dostatočne izolovanú stenu by platilo, že $1/R = 0$, teda pre zjednodušenie uvažujme že v miestnosti budú nedostatočne izolované len dve steny (obr. 1.1). Aplikáciou rovníc (1.1) a (1.2) dostaneme diferenciálnu rovnicu, ktorá takýto systém popisuje:

$$\dot{T}_I = \frac{1}{C_I} \cdot \left(\frac{1}{R_1} + \frac{1}{R_2} \right) \cdot (T_O - T_I) \quad , \quad (1.3)$$

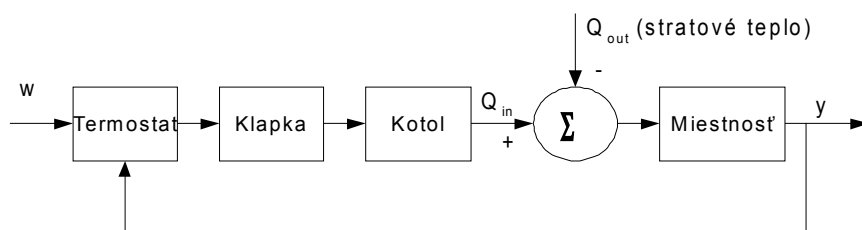
kde: C_I = tepelná kapacita vzduchu miestnosti, R_1 = tepelný odpor steny, R_2 = tepelný odpor stroopy, T_I = vnútorná teplota, T_O = vonkajšia teplota.



Obrázok 1.1: Modelová miestnosť s dvoma nedostatočne izolovanými stenami

1.1.2 Straty v tepelných systémoch

Predstavme si miestnosť popísanú o odstavce vyššie, ktorá je vytápaná pomocou kotla. Chovanie takého systému by sa dalo popísať pomocou blokovej schémy¹ z obrázku (obr. 1.2). Termostat predstavuje regulačný člen. Vzduchová klapka s kotlom je v tomto prípade akčný člen, ktorým sa ovplyvňuje výstup systému. Miestnosť je regulovaná sústava a teplota miestnosti predstavuje regulovanú veličinu. Ako je z obrázku patrné, teplo ktoré kotol vyvinie na ohrev miestnosti sa k samotnej sústave nedostane v plnej miere, ale jeho časť sa spotrebuje, resp. unikne konvenčným prúdením do okolia. Tento jav je známy ako tepelné straty. K tepelným stratám sa započítava aj teplo, ktoré slúži k ohrevu stien.



Obrázok 1.2: Interpretácia strát vo vytápanej miestnosti

1.1.3 Dvojstavová regulácia

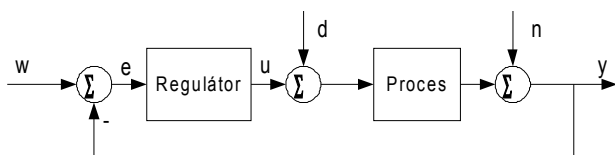
Táto regulácia spojito premennej veličiny je najjednoduchšia a rovnako aj najpoužívanejšia regulácia v oblasti vytápacích systémov. Jedná sa o dvojstavovú reguláciu spojitých systémov. Regulátor v tejto sústave nadobúda na svojom výstupe len dve hodnoty: 1) zapnutý, 2) vypnutý. Príklad zapojenia takého regulátora do regulačnej slučky je na (obr. 1.3a). Všeobecne je základným požiadavkom regulácie, aby regulovaná veličina y v každom okamihu čo možno najpresnejšie sledovala požadovanú hodnotu w nezávisle na pôsobení porúch d a šumu n a na zmenách dynamických vlastností riadeného systému. Idealizovaný popis dvojstavového regulátora je:

$$u = \begin{cases} u_{max}, & e > 0 \\ u_{min}, & e < 0 \end{cases} \quad (1.4)$$

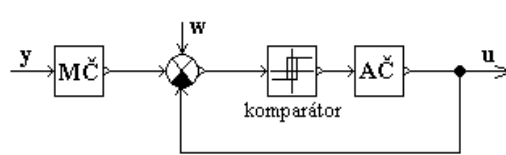
kde u je akčná veličina a $e = y - w$ je regulovaná odchylka. Zo vzťahu (1.4) teda vyplýva, že takýto regulátor generuje maximálny možný akčný zásah vždy v prípade ak požadovaná hodnota y je nižšia ako je w , a minimálny možný akčný zásah, ak je odchylka e záporná. Priebeh výstupu systému s dvojstavovým regulátorom v regulačnej slučke je na (obr. 1.4). V prípade systému vyššieho rádu bude reakcia na vypnutie akčného zásahu oneskorená. To sa prejaví ako prekmit za požadovanú hodnotu a oneskorený pokles hodnoty výstupnej veličiny, v tomto prípade sa uvažuje teplota. Čím bude systém vyššieho rádu, tým viac sa bude výstup podobať sínusovke.

¹ Pre jednoduchosť bude uvádzaná len bloková schéma, ktorá k priblíženiu problematiky postačuje.

V mnohých literatúrach je toto zanedbané a je vždy uvádzaný lineárny systém 1. rádu z (obr. 1.4a), bez poznámky že sa jedná práve o jednoduchý systém. Obrázok (obr. 1.4b) je uvádzaný ako ilustrácia daného problému s vyšším rádom sústavy.

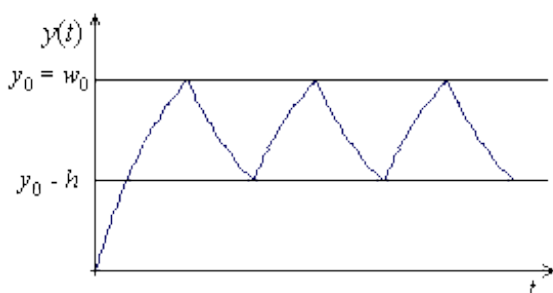


Obrázok 1.3a): Regulačná slučka riadeného procesu.

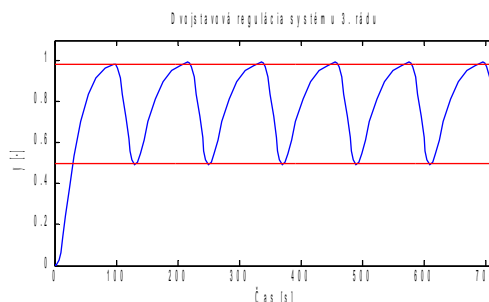


Obrázok 1.3b): Implementácia dvojestavového regulátora

Praktická implementácia dvojestavového regulátora je veľmi jednoduchá. Je znázornená na obrázku (obr. 1.3b). Popísané veličiny v obrázku odpovedajú vyššie uvádzaným, MČ predstavuje merací člen v sústave, AČ predstavuje akčný člen. Elektrická ekvivalencia tohoto regulátora sa dá predstaviť ako komparátor s hysteréziou (viď [1]), ktorý porovnáva nastavenú žiadanú hodnotu w so skutočnou hodnotou regulovanej veličiny y na výstupe a podľa toho zapína, resp. vypína akčný člen, ktorý generuje akčný zásah u .



Obrázok 1.4a): Výstup sústavy 1. rádu regulovanej dvojestavovým regulátorom.



Obrázok 1.4b): Výstup sústavy 3. rádu regulovanej dvojestavovým regulátorom.

Z profesionálneho pohľadu vyzerá použitie takéhoto regulátora ako nezmyselné, pretože akčný zásah nie je spojito premenlivý podľa požiadavku. No nesmierna výhoda týchto regulátorov je, že sú extrémne jednoduché, majú dlhú životnosť a sú nenáročné na údržbu čo je výhoda hlavne pre koncového spotrebiteľa. Okrem toho pri použití v systémoch s pomalou dynamikou (ako napríklad tepelné systémy) dosahujú uspokojivých výsledkov. Ich presnosť pri riadení tepelných systémov býva $0,5\text{ }^{\circ}\text{C}$. Táto odchylka býva pre obývané priestory pomerne malá a pre tepelnú pohodu človeka predstavuje zanedbateľnú zmenu.

2. Kapitola

2.1 PLC Systémy

2.1.1 Popis logického automatu

Programovateľný logický automat (ďalej len PLC) je číslicový riadiaci elektronický systém určený pre riadenie procesov v priemyselnom prostredí. K uloženiu užívateľského programu a užívateľsky orientovaných inštrukcií sa využíva programovateľná pamäť typu EEPROM (*Electrically Erasable Programmable Read Only Memory*). Tieto inštrukcie sú využívané k realizácií špecifických funkcií, ktoré prostredníctvom číslicových alebo analógových vstupov a výstupov automatu riadia rôznorodé procesy. Viacej podrobností okolo riadiacich automatov je možné získať z literatúry [2].

2.1.2 Štruktúra logického automatu

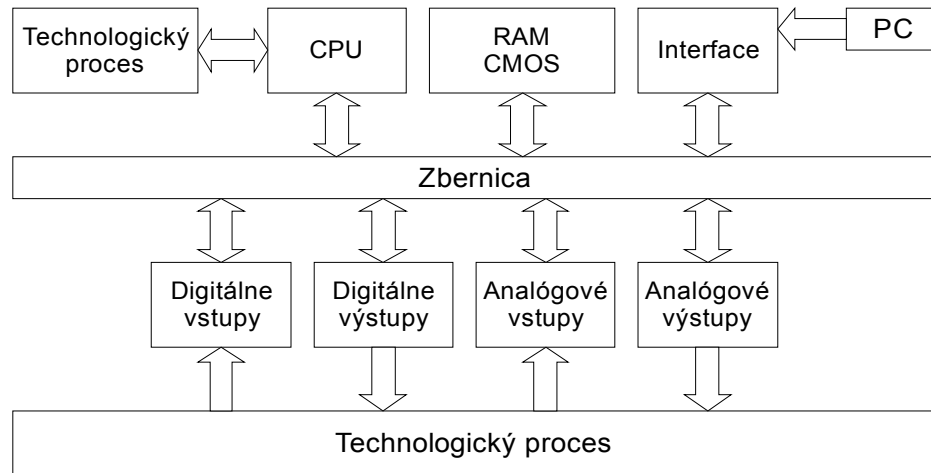
Keďže sa jedná o programovateľný elektronický systém, schopný autonómnej činnosti podľa zadaného programu, tak jeho základnou časťou je centrálna jednotka s procesorom a pomocnou sadou registrov, zbernica, pamäť pre uloženie aktuálnych výsledkov a rozhranie pre komunikáciu medzi PC a PLC, PLC a ovládacím panelom alebo medzi PLC navzájom. Schématické znázornenie je na (obr. 2.1).

Pamäť býva väčšinou zápisníková typu SRAM (*Static Random Acces Memory*), z dôvodu jej krátkej prístupovej doby k uloženým dátam. Pre udržanie výsledkov aj po výpadku napájania mávajú automaty núdzový záložný zdroj energie vo forme batérie. Rozhraním pre komunikáciu s programovacím zariadením býva obvyčajne štandard RS 232 a Ethernet. Pre styk s riadeným technologickým procesom slúži sada vstupov a výstupov automatu. Ich počet a prevedenie sa líši v závislosti na type a značke PLC. Digitálne vstupy bývajú galvanicky oddelené pomocou optočlenov. Ak takéto zabezpečenie chýba, je výhodné toto galvanické oddelenie vytvoriť, aby v prípade závady na strane riadeného procesu nevznikla škoda na riadiacom automate.

Analógové vstupy a výstupy slúžia pre pripojenie analógových snímačov, ako sú termistory pre meranie teplôt, rôzne tenzometrické snímače pre meranie tlaku, a pod., na ktorých výstupe je zaznamenávaný spojité signál (zvyčajne sa merá ich V-A charakteristika). Analógové výstupy sa využívajú pre riadenie rôznych elektronických členov resp. sústav, ktoré pre svoju činnosť využívajú spojité signál.

Základnou vlastnosťou oproti PC má PLC tú, že po výpadku napájania si musí pamätať

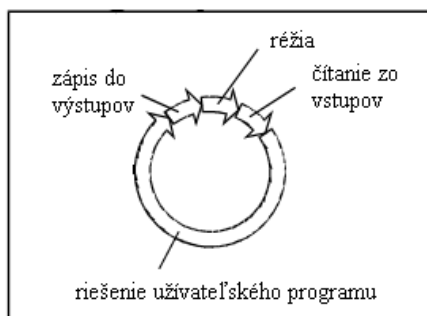
posledný stav vykonávania ištrukcie, tak ako aj stav zásobníku a operandov, v čase, kedy došlo k výpadku. To zvyšuje nároky na réžiu pamäte.



Obrázok 2.1: Vnútoraná štruktúra riadiaceho automatu

2.1.3 Programovanie logického automatu

Základom pre správny návrh programu je pochopenie princípu vykonávania programu v PLC. Narozdiel od klasického konceptu PC, sa program v PLC vykonáva cyklicky bez toho, aby to užívateľ definoval v jeho riadiacom algoritme. Jedná sa o takzvaný cyklus programu, ktorý je znázornený na (obr. 2.2). Toto cyklické vykonávanie a jednorázová aktualizácia stavu má veľkú výhodu v tom, že eliminuje vznik hazardných stavov. Riziko ich vzniku by bolo najväčšie pri zmene vstupných premenných v priebehu výpočtu algoritmu riadenia. Viacej informácií o hazardných stavoch v automatoch je možné získať v [2].



Režim RUN

Obrázok 2.2: Vykonávanie riadiaceho algoritmu v PLC

Tento fakt je potrebné si uvedomiť už pred započatím vývoja riadiaceho programu. Ušetrí to množstvo práce pri hľadaní eventuálnych vzniknutých chýb. Okrem toho automat má nadefinované 2 režimy vykonávania riadiaceho algoritmu. Režim vykonávania celej slučky programu – *RUN* a režim ladenia algoritmu, kedy je vykonávaná len réžia automatu, teda – *HALT*. Vid' [4].

2.2 Systémy FOXTROT CP1004

2.2.1 Hardvérové prostriedky

V tejto podkapitole budú rozoberaté dostupné prostriedky referencovaného riadiaceho automatu, ktoré sú priamo využívané pri tvorbe monitorovacieho systému teplôt podľa zadania. Presný popis dostupných hardvérových prostriedkov pre podporu užívateľského programu je uvedený v [4].

Jedná sa o kompaktné systémy, zložené z viacerých modulov, s možnosťou rozšírenia o prídavné vstupno-výstupné prvky. Tento model automatu má k dispozícii 256 kB CMOS RAM pamäte pre užívateľský program a tabuľky + 256 kB FLASH EEPROM pre ich zálohu. Pamäťový priestor (zápisníková pamäť - viď [10]) je rozdelený na 4 časti, ktoré majú pridelený osobitný význam. Sú to registre pomenované R, S, X, Y. Pre priblíženie je vhodné uviesť ich stručný popis. Podrobný popis a kapacita v konkrétnom prípade viď [4].

R – užívateľský register, označovaný ako remanentná zóna. Je určený na ukladanie užívateľských premenných, realizáciu čítačov, časovačov, dynamických tabuliek a pod.

S – systémový register. Vyhradený pre špecifické použitie systémovým programom automatu. Nie je vhodné ho používať na iný účel. Okrem toho obsahuje na preddefinovaných adresách rôzne príznaky a chybové hlásenia, ktoré je možné využívať pre detekciu chyby.

X – obraz vstupov automatu. Využíva sa na zapamätanie aktuálneho stavu na vstupe automatu, až do ďalšieho cyklu systémového programu PLC. Rieši sa tak hazard ktorý by mohol nastať pri zmene vstupu behom algoritmu výpočtu užívateľského programu.

Y – obraz výstupov automatu. Je nastavovaný po skončení každého cyklu PLC v závislosti na užívateľskom programe.

Okrem registrov disponuje automat 2 MB záložnej pamäte pre archiváciu celého projektu a 512 kB archivačnej pamäte tzv. DataBox, ktorý je v našom prípade využívaný ako dočasný úložný priestor pre výsledky merania pred ich archiváciou do databázovej tabuľky pomocou prostriedkov Reliance 4. Okrem pamätí by malo PLC disponovať rozširujúcim slotom pre MMC (*Multimedia Memory Card*) kartu, ktorá slúži ako úložný priestor pre webové stránky HTML (*HyperText Markup Language*) serveru, ktorý je v PLC naintegrovaný. S jeho pomocou je teda možné pristupovať k aktuálnym dátam aj z webového rozhrania v reálnom čase. Pre vlastnú implementáciu programu je využívaný čítač a realtimový obvod.

Nie menej dôležitou súčasťou je rozranie Ethernet (pre pripojenie PC) a linka CIB/Inels, ktorá je v tomto prípade využitá pre pripojenie meracích a spínacích komponent typu INELS.

2.2.2 Konfigurácia systému

Ako už bolo vyššie spomínané, PLC komunikuje s počítačom cez RS 232, alebo Ethernet. V tomto prípade bol využitý Ethernet, pretože ako programovací prostriedok bol použitý notebook, ktorý sériovým kanálom COM1 nedisponuje.

Po pripojení napájania na PLC a zicializovaní systému (po prebehnutí tzv. zapínacie sekvencie – vid' [8]) sa začne vykonávať užívateľský program (testovanie výstupov). Pre sprevádzkovanie komunikácie je potrebné zistiť IP adresu a masku siete, ktorá je automatu priradená výrobcom. To sa zistí pridržením tlačidla MODE na prednom panele. Po čase sa na displeji odroluje postupne IP adresa a maska (napr. IP: 192.168.134.176, M: 255.255.0.0). Aby bolo možné sa pripojiť priamo na PLC je potrebné nakonfigurovať aj počítač na rovnakú sieť. Teda v protokole TCP/IP je potrebné nastaviť adresu a masku (napr. IP: 192.168.134.177, M: 255.255.0.0). Po prepojení pomocou RJ45 by sa mala naviazať komunikácia medzi PLC a PC.

POZN.: Nastavenia IP sú potrebné len pre inicializáciu komunikácie. Neskôr je možné nastavenia zmeniť pomocou MOSAIC.

Ak je vytvorená sieť medzi PLC a PC, je možné prisúpiť ku konfigurácii systému. Presný postup konfigurácie v projekte je v [9] a v prílohe časť B tohoto dokumentu. U tohoto typu PLC sa celková konfigurácia systému vrátane zmien systémového času a pod. prevádza programovo pomocou vývojového softvéru MOSAIC. Po nakonfigurovaní a prípadnom reštarte PLC je možné overiť či komunikácia funguje. Po nahratí užívateľského programu (pre vyskúšanie stačí prázdna funkcia main) do PLC je to možné priamym zápisom bitu na výstup jednotky (*ikonové menu – IO – IR_1055 – DO0 až DO5*). Ak sa nastavením obrazového bitu na „1“ daný výstup zopne, komunikácia funguje správne. Správnosť a aktivitu komunikačného kanálu indikuje aj MOSAIC, takže posledný experimentálny krok je možné vynechať.

Keďže program odpovedajúci tejto práci má za úlohu archivovať dáta a následne ich rekonštruovať, je nutné pre potreby archivácie nastaviť v PLC správny dátum a čas, aby nedochádzalo k nesprávnej interpretácii archivovaných dát.

2.3 Inštalačné systémy INELS 2

2.3.1 Popis systémov

Jedná sa o „*INteligentné ELektroinštalačné Systémy*“, ktoré sú využívané hlavne pre spínanie, stmievanie, reguláciu, meranie a monitorovanie udalostí v objektoch (požiar, pohyb, atď.). Tieto elektroinštalačné komponenty sú schopné obojsmernej komunikácie s centrálnou jednotkou PLC po zbernici CIB. O vnútornej štruktúre sa z dôvodu ochrany autorských práv nedá získať veľa informácií. Ale technicky vzaté, ide o spínacie relé s komunikačným rozhraním zbernice, ktoré má pridelenú vlastnú adresu a je ovládané signálami zo zbernice. Tieto sú vysielané centrálnou jednotkou, ktorá vyhodnocuje program a výsledky predáva týmto prvkom po zbernici. Okrem toho prvky INELS v sebe obsahujú vstup pre termočlánok, takže majú aj linearizačné obvody pre prácu s nimi. Pre potreby tejto práce boli využité nižšie uvedené a popísané komponenty.

IM2_80B – je jednotka vstupov pre pripojenie ôsmich zariadení ako sú spínače, tlačidlá, PIR detektory, plynové senzory a podobne. V tejto práci využitá pre pripojenie už spomínaného PIR (infračervený pohybový senzor).

SA2_01B – je jednokanálová jednotka, určená pre spínanie spotrebičov do 16 A. Obsahuje vstup pre pripojenie tepelného senzoru. Pre potreby práce využitý len teplotný vstup.

TZ, TC – sú teplotné čidlá, jedná sa o termistor v kovovom (TZ), alebo plastovom (TC) púzdre zaliaty do teplovodivého tmelu.

JS-20 LARGO – je infračervený detektor pohybu s vysokou citlivosťou. Rozsah detekcie je 120° a vzdialenosť 12 m. Podrobné informácie viď [5].

SA-200 – je magentický dverný kontakt malých rozmerov určený pre montáž na otvárateľné časti objektu.

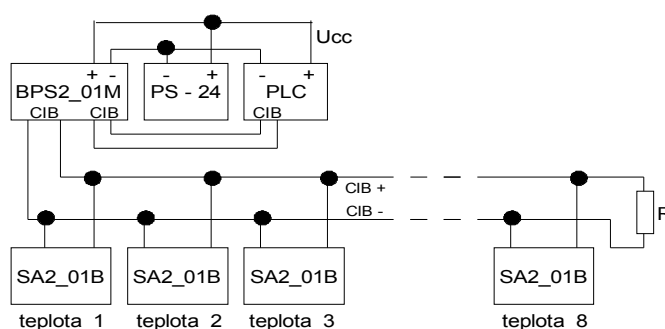
BPS2-01M – je jednotka oddeľovača zbernice. Slúži na odizolovanie napájania pre zbernicu od zdroja, ktorým je napájané PLC. Eliminujú sa tak rôzne rušenia ktoré majú pôvod v zdroji a rozvodnej sieti.

PS-24 (ELKO) – Napájací zdroj pre PLC a zbernicu. Napätie DC 24V, max. zaťaženie 1250 mA.

POZN.: Pre podrobné informácie o týchto produktoch viď [5].

2.3.2 Inštalácia a konfigurácia systémov

Inštalácia sa prevádza z dôvodu bezpečnosti do inštalačných krabíc. V tomto prípade sa krabice vynechali a spínacie jednotky sú inštalované len voľne. Keďže sa nevyužíva silový obvod týchto prvkov, nie sú porušené zásady bezpečnosti a nebezpečenstvo úrazu elektrickým prúdom je vylúčené. Fyzické parametre spínacích prvkov vid' katalógové listy [5]. Tieto prvky, komunikujú s riadiacou jednotkou po zbernici. Pre zapojenie bola zvolená zbernicová topológia, podľa odporúčaní výrobcu funkčných jednotiek. Ako fyzické médium pre vytvorenie zbernice bola použitá krútená tienená dvojlinka (twisted-pair). Usporiadanie jednotiek je na obrázku (obr. 2.3). Ku každej jednotke pripojenej na zbernici je pripojený tepelný senzor (termistor), ktorý mení stav svojho výstupu (rezistencia) v závislosti na teplote. Jednotka automaticky vyhodnocuje pomocou vnútornej logiky stav tohoto senzoru a na zbernicu vysiela výsledok prepočítaný už priamo v °C. Pri inicializácii systému PLC automaticky zisťuje stav na zbernici. PLC zistí prítomnosť nových jednotiek na zbernici, ale obrazy ich výstupov nenamapuje automaticky do svojich vnútorných registrov. To je úloha programátora. Celková konfigurácia PLC je robená v programe MOSAIC. Praktický postup s obrázkom je uvedený v prílohe časť B. Po pripojení prvkov INELS sa automaticky vygeneruje program, ktorý zabezpečuje komunikáciu medzi PLC a jednotkami na zbernici. Pre komunikáciu s jednotkami a pre uloženie obrazov ich vstupov resp. výstupov je v registri R (remanentná zóna) vyhradených prvých 110 Bytov, teda oblasť s absolútnou adresou %R0 až %R110. Toto pamäťové miesto musí zostať vyhradené práve pre tento účel. Po prípadnom alokovaní premennej do tejto zóny hlási prekladač chybu. Pre využívanie obrazov tepelných vstupov spínacích jednotiek je výhodné priradenie symbolického mena každému z nich. Zabezpečí sa tak vyššia prehľadnosť kódu než použitím absolútnych operandov. Po priradení symbolických mien (*Zobrazit – Manažer projektu – HW – Konfigurace HW - Nastavení V/V*) je potrebné preloženie programu. Po jeho následnom vyslaní do programového priestoru PLC sú jednotky zinicializované a hodnoty ich V/V sú namapované na adresách s priradeným symbolickým menom. Pre odskúšanie komunikácie je možné manuálne zapnúť a vypnúť každý réleový výstup spínacej jednotky, čím dostaneme prehľad aj o adrese mapovania svojho výstupu každej jednotky. Ak komunikácia funguje a je možné ovládať každú jednotku, nastavenie je kompletne.



Obrázok 2.3: Pripojenie komponentov Inels2 k PLC

3.Kapitola

3.1 Softvérové prostriedky

3.1.1 Popis vývojového softvéru MOSAIC

Je vývojové prostredie určené pre programovanie riadiacích systémov firmy Teco a.s. . Prostredie v sebe integruje viacero programovacích štandardov, ktoré sa využívajú pri tvorbe algoritmov pre riadiace systémy. Sú to štandardy podľa normy *IEC 61 131-3*, ktorá ich definuje štyri. Textové jazyky ako *IL (Instruction List)*, *ST (Structure Text)* a grafické jazyky ako je *LD (Ladder Diagram)*, *FBD (Function Block Diagram)*. Pre potreby tejto práce bol využitý jazyk *ST*. O syntaxi a štruktúre zápisu kódu v tomto dokumente nebude zmienené, ale podrobnosti (tak ako aj podrobnosti o ostatných spomínaných štandardoch) sú dostupné v [6] .

3.1.2 Popis softvéru RELIANCE 4

Software *RELIANCE 4* patrí medzi takzvané *SCADA* systémy, umožňujúce ako tvorbu vizualizácií k projektom, tak riadenie technologického procesu prostredníctvom zmien technologických premenných ovplyvňujúcich vývoj riadiaceho algoritmu. Pre zlepšenie predstavy o *SCADA* vid' (obr. 3.1). Systém *Reliance* má štyri programové komponenty, a to:

- *RELIANCE DESIGN* (hlavná komponenta, slúži pre návrh)
- *RELIANCE CONTROL SERVER* (zobrazenie a projektu + klientský server)
- *RELIANCE CONTROL* (zobrazenie projektu s možnosťou riadenia)
- *RELIANCE VIEW* (zobrazenie projektu bez práv na riadenie)

V tejto práci sa využívali komponenty *DESIGN*, *CONTROL* a *CONTROL SERVER*. Systémy podporujú štandard databázy pre archiváciu ako je *SQL*, čím umožňujú ukladať data centrálné na server, a sprístupniť ich tak viacerým užívateľom súčasne. To je výhodné pri implementácii webových klientov, kedy má klient prostredníctvom vizualizácie prístupný aktuálny stav riadeného resp. monitorovaného procesu, a chce pristupovať k archívnym dátam.

Software-ové prostredie je vyvinuté intuitívne, a má dobre spracovaný help, takže jeho zvládnutie predstavuje primeranú časovú záťaž. Podrobnosti o týchto programových komponentách sú dostupné v [7].

3.1.3 Nastavenie komunikácie s PLC v RELIANCE

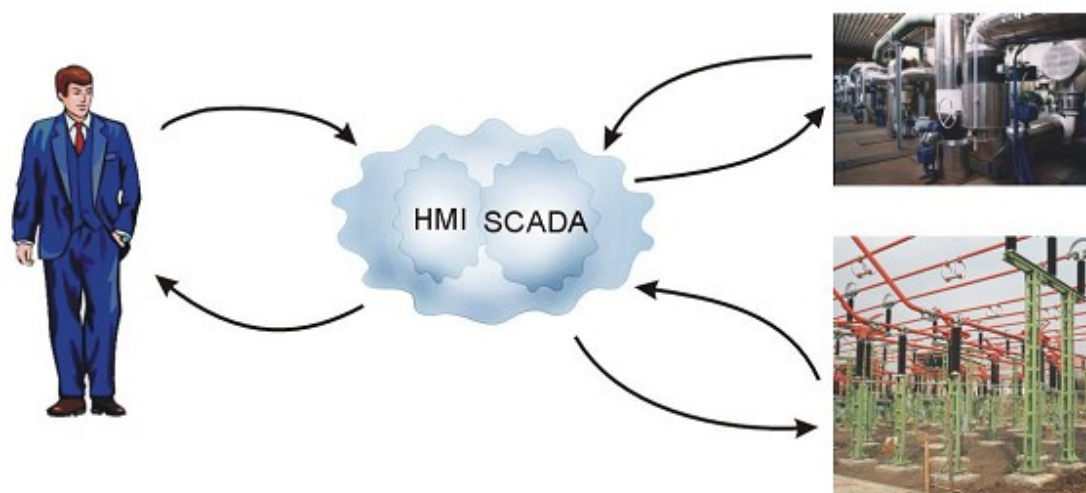
V odstavci bude popísaný podrobný postup ako nadviazať komunikáciu medzi PLC a RELIANCE. Najprv je potrebné si v RELIANCE založiť nový projekt. Po úspešnom založení projektu postupujeme podľa nižšie uvedených krokov, ku ktorým sú priložené ilustrácie. Ilustrácie sa nachádzajú v sekcii prílohy, časť A. Konfigurácia je vytváraná pre prípad, kedy je PLC pripojené k počítaču cez ethernet a je problémové nadviazať priame spojenie R4 a PLC.

Krok 1) V roletovom menu sa zvolí položka *Správci - Správce stanic*. Objaví sa okno so zoznamom pripojených staníc k PC.

Krok 2) Kliknutím na ikonu pridať stanicu sa objaví preddefinovaný zoznam PLC, ktoré RELIANCE podporuje. Vyberieme stanicu TECO.

Krok 3,4,5) Nastavíme parametre podľa obrázkov.

POZN.: Po nastavení parametrov v kroku 4 sa objaví dialógové okno o pripojení stanice k PC. To je potrebné potvrdiť a stanica sa tak pripojí k PC.



Obrázok 3.1²: Všeobecný pohľad na funkciu SCADA systému

2 Obrázok prebratý z [11].

4. Kapitola

4.1 Monitorovací systém

Pre získanie dlhodobějších výsledkov ktoré poslúžia k demonštrácii funkčnosti systému by mal byť inštalovaný v obytnom priestore so zvýšeným výskytom osôb, aby bolo možné čo najlepšie sledovať dynamické deje.

4.1.1 Vol'ba veličín k sledovaniu

Pre potreby tohoto merania je výhodné konrolovať hneď niekoľko dejov, ktoré sa vo vytápaných priestoroch odohrávajú. Účelné je sledovanie teplôt v určitých lokálnych oblastiach, ako sú topné telesá, oblasti okolo okien, oblasť dverí a referenčná oblasť, za ktorú je možné považovať stred miestnosti. Aby bolo možné vyhodnocovať deje prebiehajúce vo vnútri priestoru v závislosti na vonkajších vplyvoch (počasie a pod.), je taktiež vhodné umiestniť snímače aj do otvoreného vonkajšieho priestoru.

Sledovanie teplôt v okolí topných telies, prípadne priamo na nich, je vhodné z dôvodu identifikácie topného systému. Za najrozumnejšie miesto na topnom telese, kam uložiť tepelné čidlá, sa pokladá prítokové a odtokové potrubie. Zistí sa tým koľko konvenčného tepla sa odovzdá okoliu za určitý čas, poprípade doba nábehu teploty na požadovanú hodnotu od zapnutia kotla. Taktiež môžu poslúžiť k identifikácii vnútorného systému topnej sústavy, teda myslí sa tým okruh kotla, potrubia, a topných telies, ktorý by pomohol pri prípadnom neskoršom návrhu inteligentnej regulácie, postavenej na základe praktického chovania tohoto podsystemu.

Návrh na sledovanie teplôt v oblasti okien je postavený na základe teórie o prúdení tepla. Klasická topná sústava je založená na zvyšovaní teploty v miestnosti prostredníctvom ohrevu okolitého vzduchu. Ako už bolo spomenuté jedná sa o konvenčné prúdenie. Narozdiel od infračerveného vytápania, kedy sú telesá ohrievané radiáciou infračerveného svetla topného telesa, konvenčný ohrev vykazuje rozdiely teplôt v rôznych výškach. Topné telesá bývajú umiestnené v spodnej časti miestnosti. Z teórie hustoty plynov je známe že ohriaty vzduch má tendenciu stúpať hore, a naopak, ochladený vzduch sa drží pri podlahe. Pri otvorenom okne sa zvyšujú straty topnej sústavy, čo je spôsobené zväčšením plochy vytápaného priestoru a hlavne únikom teplého vzduchu do vonkajšieho okolia a prúdením chladného vzduchu do miestnosti. Dôsledky takejto výmeny sú, že topné teleso musí odovzdať svoju energiu na zohriatie chladného vzduchu. To sa vo vodnom okruhu topnej sústavy prejaví ako zníženie teploty topného média (zvyčajne voda). Teda je zaujímavé sledovať súvislosti medzi teplotou prúdiaceho vzduchu

a teplotou vytápaného telesa.

Oblasť dverí je zaujímavá z podobných dôvodov ako u okien. Rozdiel oproti prúdeniu vzduchu z okien je ten, že pri ich otvorení vznikne prúdenie s vyššou rýchlosťou. To spôsobí, že pri krátkom otvorení dverí sa vymení v miestnosti väčší objem vzduchu, ako za rovnaký čas otvoreného okna. Okrem toho je zaujímavé sledovať kombináciu týchto dvoch javov, a ich dopad na zmenu okolitej teploty miestnosti a zmenu teploty topného telesa, ktorá podľa predpokladu bude z dôvodu mernej tepelnej kapacity C topného média menej badateľná.

Sledovanie referenčnej oblasti je navrhnuté z dôvodu, že sa predpokladá najvyššia stálosť teploty oproti predchádzajúcim oblastiam, ktoré boli v blízkosti tepelného zdroja, alebo „zdroja chladu“. Táto oblasť je od všetkých týchto zdrojov rovnomerne vzdialená a je vhodná na vyhodnocovanie chovania tepelného systému, teda miestnosti ako celku.

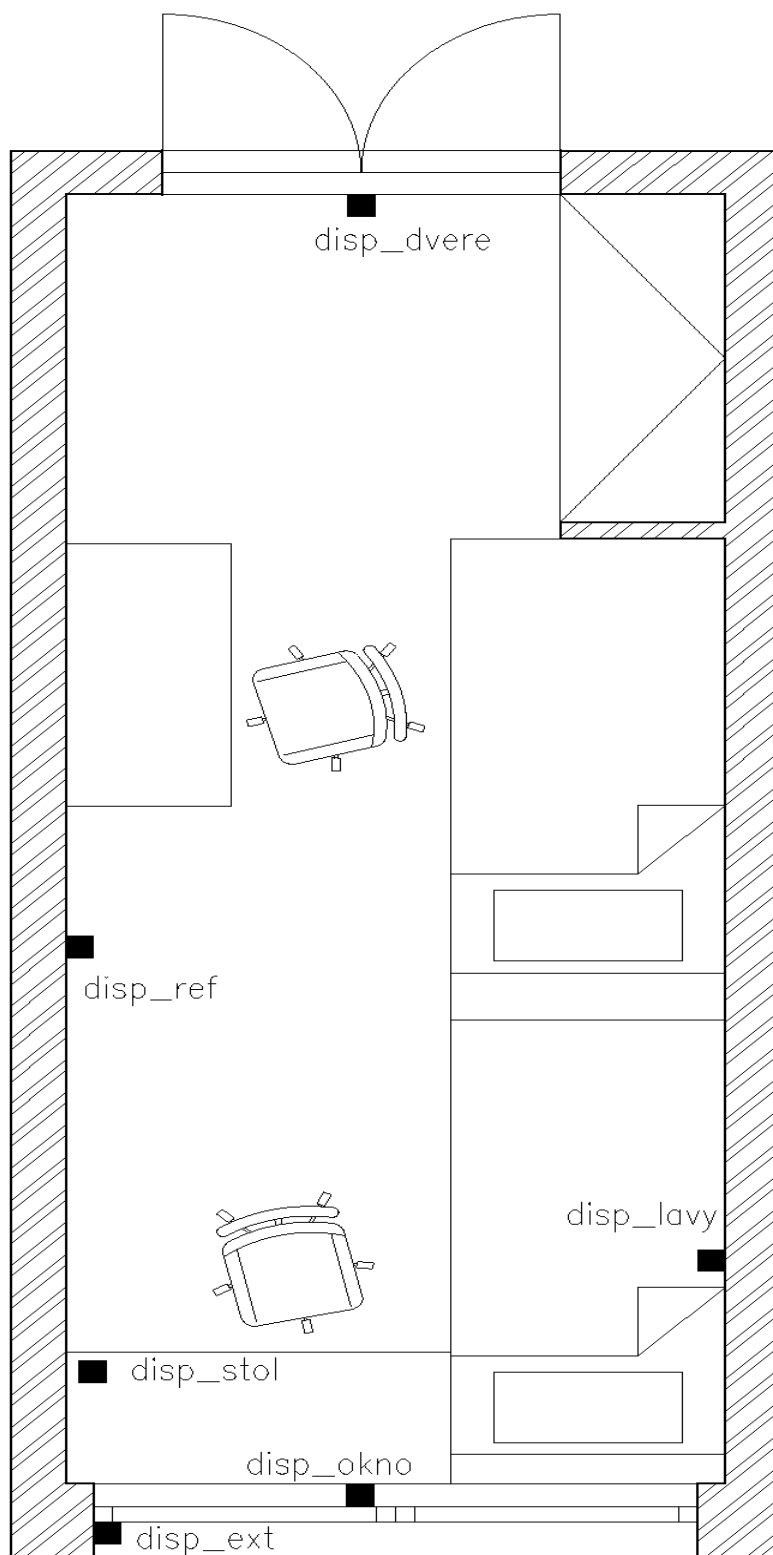
Okrem snímačov teplôt je vhodné použitie spínacích senzorov, ktoré detekujú otvorenie okien a dverí, aby sa zachytila príčina zmeny teploty v danom priestore. Popri snímačoch stavu okna resp. dverí je vhodné sledovanie pohybu v monitorovanom priestore. Ľudské telo o priemernej ploche 2 m^2 (čo odpovedá priemernému dospelému človeku) predstavuje zdroj tepla. Podľa [12] priemerný človek schopný vyprodukovať až cca. 90 W tepelného výkonu. Táto hodnota nespôsobí za prítomnosti jedného človeka badateľnú interakciu medzi monitorovaným priestorom z hľadiska príspevku tepelnej energie. Pri jeho pohybe však nastáva zvýšené prúdenie vzduchu v miestnosti a to môže badateľne ovplyvniť rozdiely teplôt v meraných oblastiach. Okrem toho pri zvýšenom výskyte osôb v tomto priestore už môže tento tepelný výkon badateľne ovplyvniť stav referenčnej teploty, za predpokladu uzavretých dverí a okna.

Ďalším z neprehliadnuteľných faktorov pri výbere meraných veličín je vplyv slnečnej energie na stav teploty. Slnko ohrieva objekty radiáciou, čo je vlastne šírenie energie prostredím (ako aj vákuom) pomocou vlnenia. To znamená že ak sa bude merať referenčná teplota vzduchu, slnko nie je práve tou príčinou, ktorá spôsobí priamo ohriatie. Teplo sa po miestnosti bude šíriť znova len konvenčne a to od okolitých pevných telies, ktoré absorbovali slnečnú energiu a tým sa zvýšila ich teplota. Zvýšenie teploty telesa spôsobí pokles entropie v miestnosti. Tento pokles musí byť podľa vety o termodynamickú rovnováhu vyrovnaný, to znamená, že teplo ktoré získa teleso prostredníctvom radiácie od slnka predá okolnému prostrediu. To má za následok zvýšenie referenčnej teploty v miestnosti.

Pre sledovanie slnka by mohol byť použitý diferenčný senzor, ktorý zisťuje intenzitu žiarenia z rozdielu teplôt bielej a čiernej plochy, ktoré vzniknú z dôvodu absorpcie žiarenia. Ďalšou variantou je použitie klasických senzorov a meranie intenzity aplikovať ako výpočetný algoritmus PLC.

4.1.2 Umiestnenie senzorov

Zámer a odôvodnenie pre jednotlivé miesta pre senzory boli popísané v časti 4.1.1 tohoto dokumentu.



5. Kapitola

5.1 Návrh riadiaceho programu pre PLC

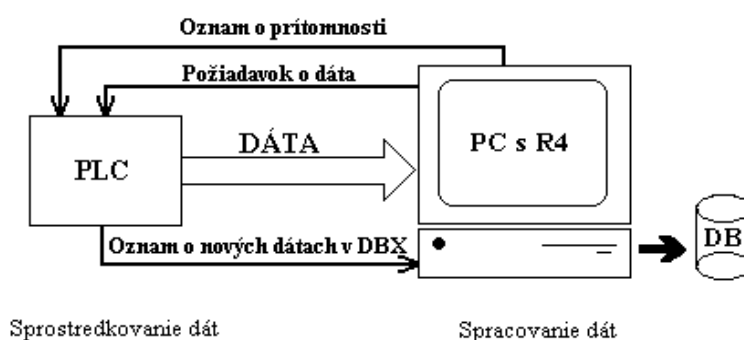
5.1.1 Dekompozícia problému

Hlavný problém je v ukladaní a archivovaní vzoriek nameraných teplôt pre neskoršiu interpretáciu. Archivácia dát v PLC nie je najlepším riešením, pretože jednak vzniká problém s nedostatkom využiteľnej pamäte a ďalší problém je spätná interpretácia nameraných dát. Riešením je celý problém rozdeliť do dvoch častí a vytvoriť tak dva kooperujúce mechanizmy, ktoré by celý problém monitorovania zvládli. Teda vytvoriť vo vývojovom prostredí MOSAIC program, ktorý by riadil činnosť PLC, a program v RELIANCE 4, ktorý by vizualizoval tepelné procesy a staral sa ako o archiváciu, tak o spätnú interpretáciu a správu nameraných dát.

Hlavnou myšlienkou programu pre obsluhu činnosti PLC bolo, aby automat vykonával len úlohu sprostredkovateľa prístupu k aktuálnym dátam na senzorech. Keďže sa jedná o monitorovací systém, ktorý nepotrebuje žiadne riadiace algoritmy pre povelovanie periférnych členov s účelom vytvárania akčných zásahov, táto funkčnosť bude pre tieto potreby postačujúca. Ako prídavná funkcia, ktorá presahuje požiadavky zadania práce, je prídanie schopnosti automatu dočasne prebrať kontrolu nad monitorovacím systémom a archivovať informácie o aktuálnom stave na senzorech za neprítomnosti hlavného archivačného programu, ktorý sa implementuje v RELIANCE 4. Otázkou je, čo s nameranými dátami ktorými disponuje PLC, ak hlavným prostriedkom pre archiváciu a správu dát je RELIANCE 4 (ďalej len R4). Ako predať nové dáta R4 tak, aby ich pridal do hlavného archívu? Ako vyriešiť problém delenia práce medzi týmito dvoma programami?

Jednou z najlepších možností je využitie OPC (*OLE for Process Control*) serveru, ktorý v sebe implementujú obidva vývojové prostriedky. Vďaka nemu je možné sprístupniť dátovú oblasť pamäte PLC, a využiť ju ako zdieľaný prostriedok pre vzájomnú komunikáciu obidvoch programov. Kooperáciu týchto programov je možné si predstaviť ako klient-server. Kde v úlohe klienta je „nosný“ program monitorovacieho systému vytvorený v R4, a v úlohe servera je PLC. Tí by si medzi sebou vymieňali dáta a kontrolovali stav jednotlivých programov (prítomnosť, požiadavok o data, oznam o nových dátach a pod.) a podľa toho by nastavovali algoritmus riešenia archivácie a delby práce. Pre lepšiu predstavu je táto koncepcia zobrazená na (obr. 5.1). Praktické znázornenie delenia práce medzi jednotlivými riadiacimi mechanizmami je znázornené na (obr. 5.2). V normálnom režime, t.j. je spustený monitorovací systém implementovaný v R4

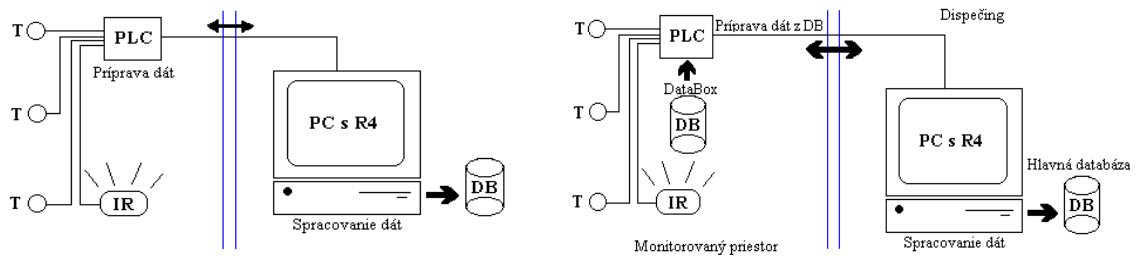
(zobrazený na obr. 5.2a)), PLC vykonáva funkciu komunikačnej jednotky s perifériami a sprostredkovateľa dát pre monitorovací systém implementovaný v R4. Ten preberá aktuálne dáta z vyhradenej pamäťovej oblasti PLC a ukladá ich do hlavnej databázy. Tá môže byť vytvorená lokálne, teda priamo v PC, alebo môže byť decentralizovaná a uložená na externom servere. V prípade, že dôjde k poruche spojenia medzi PLC (obr. 5.2c) a monitorovacím systémom, algoritmus v automate detekuje chybu a preberie riadenie a monitorovanie systému. Dáta sa budú archivovať do jeho vnútornej pamäti (DataBox) až do chvíle, kým sa znova neidentifikuje prítomnosť riadiaceho programu v R4. Ak dôjde k prerušeniu spojenia v dôsledku vypnutia monitorovacieho systému užívateľom, R4 oznámi svoju absenciu tesne pred ukončením a automat na tento podnet prejde do režimu archivácie.



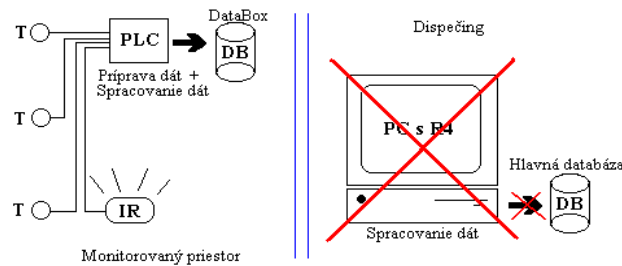
Obrázok 5.1: Konceptia klient-server medzi PLC a PC

Z dôvodu obmedzeného miesta v DataBoxe sa predpokladá, že chyba spojenia nebude dlhšia ako týždeň³. Po opravení spojenia medzi týmito mechanizmami (obr. 5.2b) sa R4 prihlási u PLC. To mu oznámi prítomnosť nových dát v lokálnej pamäti. Keďže R4 je vývojové prostredie obmedzené dátovými bodami a priamy prístup k dočasne archivovaným dátam v PLC by ich spotreboval mnoho, pristúpilo sa na (pomalejšiu) variantu požiadavku o dáta. To znamená že R4 požiada o záznam, PLC ho z DataBoxu vyberie a uloží ho na adresu prístupnú pre R4. Sprístupnenie oznámi a čaká na povel pre načítanie nasledujúceho záznamu. R4 si záznam prečíta, uloží ho do hlavnej databázy a požiada PLC o nový záznam. To sa cyklicky opakuje, až kým sa DataBox nevyprázdni. Po ukončení predávania dát sa systém vráti do normálneho stavu, ktorý je znázornený na (obr. 5.2a). Popri výmene dát prebieha na pozadí algoritmu PLC komunikácia s perifériami, a na pozadí monitorovacieho algoritmu R4 archivácia aktuálnych dát z periférií. Vyčítavanie dát z PLC je užívateľovi oznámené prostredníctvom vizualizačného okna. Bližší popis o akciách vo vizualizácii bude rozobratý nižšie.

³ Pri 512KB, 8 teplotných senzoch a perióde vzorkovania 80s.



Obrázok 5.2 a): Normálny režim práce PLC a monitorovacieho programu v R4 **Obrázok 5.2 b): Práca PLC po znovupripojení monitorovacieho programu**



Obrázok 5.2c): Režim práce PLC bez prítomnosti monitorovacieho programu

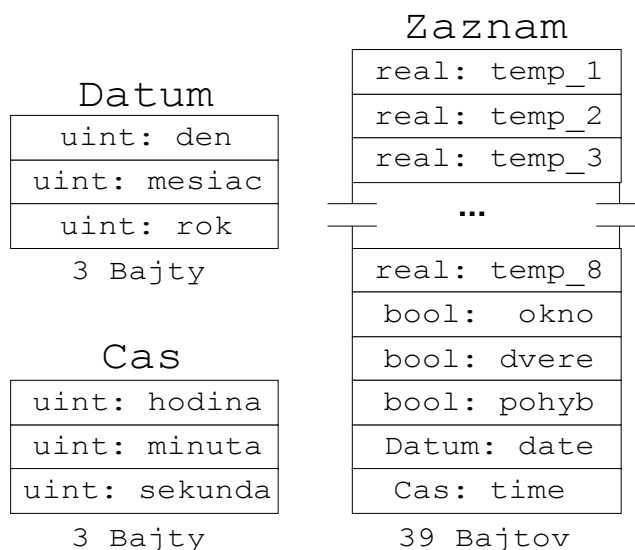
5.1.2 Funkcie využité v programe

Funkcia je podľa [6] definovaná ako najjednoduchší blok programu, ktorého hlavnou charakteristikou je, že môže dostávať parametre, môže vrátiť len jeden výsledok a po opakovanom volaní s rovnakými parametrami musí byť každý z vrátených výsledkov totožný s ostatnými. Pre priblíženie je vhodné popísať len najvýznamnejšie bloky riadiaceho algoritmu PLC a ich použitie. Pre podrobné zoznámenie s kódom viď príloha na CD. Štruktúra zápisu kódu sa drží konvencie použitej v [6].

Pre potreby tejto práce bolo potrebné definovať vlastné dátové štruktúry. Aby sa s dátami pracovalo čo najlepšie, boli navrhnuté 3 štruktúry (obr. 5.3). V prvom rade bola navrhnutá vlastná štruktúra dátumu a času s rovnocennými názvami Datum a Cas. Aj keď MOSAIC podporuje vlastné dátové typy DATE a TIME, pre potreby archivácie a interpretácie v R4 nie sú vhodné, pretože ten nepodporuje priamy prístup k premenným tohoto typu. Preto sa pristúpilo na variantu vlastných typov. Tieto štruktúry sú využívané pre presnú časovú identifikáciu vytvorenia záznamu. Ďalšou nadefinovanou štruktúrou je štruktúra s názvom Zaznam. Jej presná interpretácia je na (obr. 5.3). Štruktúra je využitá pre ukladanie aktuálnych dát z periférií. V režime bez prítomnosti R4 (obr. 5.2c) sa každý nový záznam ukladá ako celok do DataBoxu.

Zápis do tejto pamäte nie je taký jednoduchý ako zápis do registrov. Pri tomto úkone je potrebné prenášať bloky dát ako bytové pole bez ohľadu na dátový typ dát. Keby sa prenášali jednotlivé hodnoty od každého senzoru osobitne, mohlo by dôjsť k nekonzistencii dát, to znamená

že pri prenášaní väčších blokov dát by mohlo dôjsť k zámene dát. Výhoda ukladania celej štruktúry je v tom, že pri jej spätnom výbere je postačujúce vedieť na ktorú adresu bola archivovaná štruktúra uložená. Podľa jej dĺžky sa ľahko vyberie ako blok dát a uloží sa na adresu už preddefinovanej premennej typu `Zaznam`. Tým sa dosiahne pretypovanie bloku bytov na dátovú štruktúru bez zložitých konverzií, ku ktorej prvkom je jednoduchý prístup cez „.“ operátor. V tejto práci sa tento spôsob prístupu k položkám v značenej miere využíva.



Obrázok 5.3: Vlastné štruktúry premenných využitých v programe PLC

Z najdôležitejších funkcií v programe je samozrejme hlavný program, ktorý je vykonávaný v PLC cyklicky, tak, ako je to zobrazené na (obr. 2.2). Jeho štruktúra bola navrhovaná tak, aby pohľadom na kód bolo hneď patrné o čo sa jedná. Algoritmus hlavného programu `prgMain` je rozdelený na dva podmienené bloky. A to blok pre vykonávanie archivačného algoritmu, a blok pre vykonávanie serverového algoritmu pre R4, teda výberu krátkodobo archivovaných dát z pamäte `DataBox` a ich sprostredkovanie. Popis hlavného algoritmu programu cyklicky vykonávaného je znázornený v podkapitole 5.1.3 na (obr. 5.4). Pre celkové pochopenie algoritmu budú popísané aj pomocné rutiny, ktoré sa na výpočetnom algoritme podieľajú. Tie sa dajú rozdeliť do niekoľkých skupín:

1) *Funkcie slúžiace na získavanie dát*, teda podľa konvencie „JAVA“ označované ako „GET“funkcie. Do tejto skupiny sa dajú zaradiť funkcie `GetDatum()` a `GetCas()`.

2) *Funkcie slúžiace k archivácii*, sú funkcie, ktoré sa priamo alebo nepriamo zúčastňujú na archivácii aktuálnych záznamov do pamäte `DataBox` v PLC, pri neprítomnosti vizualizaného

programu. Patria k nim funkcie ako `Scan()`, `ZapisZaznam()` a `Archivuj()`.

3) *Funkcie slúžiace pre výber z pamäte DataBox*, sú funkcie, ktoré sú volané po ohlásení činnosti vizualizačného programu. Po registrovaní príchodu R4 sa podľa vývojového diagramu (obr. 5.4) začne vyčítavanie archivovaných záznamov v PLC. Na tejto činnosti sa podieľajú funkcie `CitajZaznam()` a `VyberDat()`.

4) *Funkcie pre kontrolu komunikácie a riadenia*, kontrolujú stav prítomnosti R4 a podľa toho nastavujú algoritmus riadenia monitoringu. Sú to funkcie `Riadenie()` a `WatchDogFcn()`.

5.1.3 Popis funkcií využitých v programe

Naznačená funkčnosť vyššie uvedených funkcií slúži len pre orientáciu v navrhnutom kóde. Presný popis bude uvedený v tomto odstavci.

1) Funkcie pre získavanie dát

`GetCas()` – funkcia načíta zo systémových registrov aktuálny stav času a vráti ho ako dátovú štruktúru `Cas` zobrazenú na (obr. 5.3). Dôvody vytvorenia týchto štruktúr a funkcií sú uvedené vyššie v odstavci 5.1.2.

`GetDatum()` – podobne ako funkcia času načítava zo systémových registrov PLC aktuálny stav a ukladá ho do dátovej štruktúry `Datum`, ktorú aj vracia.

2) Archivačné funkcie

`Scan()` – funkcia uloží hodnoty z predom pripravených obrazov aktuálnych stavov periférií uložených v špecifickej časti pamäti (obr. 5.5) a aktuálny čas a dátum do štruktúry `zaznam`, ktorá je vytvorená na predom špecifikovanom mieste, aby bolo možné tento aktuálny `zaznam` zdieľať medzi ostatnými funkciami. Správnosť aktualizácie dát oznamuje návratovou hodnotou typu `BOOL`. Funkcia sa zúčastňuje archivácie nepriamo sprostredkovaním aktuálnych dát z periférií.

`ZapisZaznam()` – funkcia preberie dáta zo zdieľanej pamäte a zapíše ich do pamäte `DataBox`, na adresu, ktorú získa tiež zo zdieľaného priestoru. Tú musí vopred nastaviť iná rutina. Funkcia ako samostatný blok nie je v hlavnom programe využívaná, ale je využitá pre účely inej archivačnej funkcie. Ako návratovú hodnotu vracia typ `BOOL` podľa úspešnosti uloženia záznamu v `DataBoxe`.

`Archivuj()` – funkcia zisti či je v `DataBoxe` voľné miesto pre uloženie ďalšieho záznamu, ak je a má k dispozícii všetky aktuálne dáta tak inkrementuje ukazateľ na posledne obsadenú adresu v `DataBoxe` a uloží aktuálny záznam z periférií pomocou funkcie `ZapisZaznam()`. Podľa úspešnosti tejto funkcie a podľa príznakov o nových dátach sa vracia typ `BOOL` hovoriaci o úspešnosti archivácie.

3) Funkcie pre výber dát z DataBoxu

`CitajZaznam()` – je inverziou k funkcii pre zápis záznamu. Funkcia prečíta z posledne čítanej adresy (ktorú nastavuje iný, „vyšší“ mechanizmus) záznam a uloží ho na vopred definované miesto v pamäti, ku ktorému má prístup R4. O úspešnosti operácie rozhoduje počet prenesených bytov. Ak je odpovedajúci požiadavku, funkcia vráti typ `BOOL` ktorý potvrdzuje správnosť priebehu (`true`), ak nie tak vráti `BOOL` s informáciou o chybe (`false`).

`VyberDat()` – funkcia, ktorá implementuje serverový mechanizmus obsluhy. Zisťuje prítomnosť dát v `DataBoxe` a požiadavok na nich. Ak toto existuje, tak pomocou vyššie uvedenej funkcie pre čítanie záznamu sprístupní archivovaný záznam, oznámi prítomnosť vyčítaného záznamu nastavením príznakového bitu a posunie ukazateľ na nasledujúci najstarší záznam v `DataBoxe`. Úspešnosť akcie je podmienená výsledkom čítacej funkcie, resp. zostávajúcim počtom záznamov v `DataBoxe`. Funkcia teda vracia typ `BOOL`.

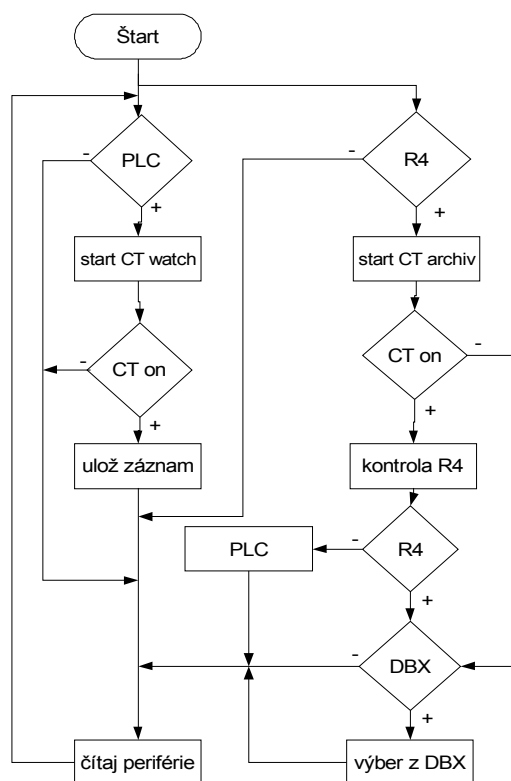
POZN.: Ak sa už začne vyčítavať `DataBox`, je potrebné aby táto operácia bola atomická, inak sa prepísaním ukazateľov na posledné (čítané a zapisované) adresy v `DataBoxe` znemožní opätovný prístup k zostávajúcim dátam. Tie môžu byť prepísané novými dátami. Atomicita je aspoň čiastočne zabezpečená v prostredí R4.

4) Kontrolné funkcie

`Riadenie()` – funkcia, ktorá kontroluje príznaky riadenia (štruktúra rozdelenia pamäte je na (obr. 5.5)) a ako návratovú hodnotu vracia typ `BOOL`, ktorý hovorí o tom, či má kontrolu nad monitorovaním priestoru PLC, alebo R4.

`WatchDogFcn()` – funkcia ktorá v určitých časových intervaloch kontroluje či bol nastavený príznak komunikácie (nastavuje ho R4). Ak počas intervalu nečinnosti tejto funkcie nastavený bol, je všetko v poriadku, funkcia tento príznak znovu vypne a vráti výsledok `true` typu `BOOL`. Ak tento príznak behom jej nečinnosti nastavený nebol, tak vráti výsledok `false`. Jej výsledok je potom v hlavnom programe použitý pre rozhodovanie nastavenia algoritmu riadenia. Tento spôsob kontroly je tzv. `watchdog` (strážny pes). Je popísaný v [10].

5.1.4 Vývojový diagram hlavného algoritmu PLC



Obrázok 5.4: Vývojový diagram hlavného programu

Cyklus programu by mal byť jasný z vývojového diagramu. Pre lepšie pochopenie je vhodný popis jednotlivých blokov.

PLC – označuje stav monitorovacieho systému, kedy preberá riadenie a archivovanie dát do svojej kompetencie práve PLC. Stav je znázornený na (obr. 5.2c).

R4 – označuje stav systému, kedy je zistená alebo ohlásená prítomnosť vizualizačného programu vytvoreného v R4. Tento stav je znázornený na (obr. 5.2a).

CT archiv – je označenie čítača, ktorý riadi periódu odoberania záznamov z okolitých periférií. Tento čítač je aktívny len v prípade vyššie popísaného režimu PLC, tak ako je to patrné z vývojového diagramu.

CT watch – je označenie čítača takzvaného „watchdog“ (strážny pes). Ten slúži v hlavnom programe na určenie intervalu kontroly, s akým sa bude zisťovať prítomnosť vizualizačného programu.

CT on – označuje že uplynula doba za ktorú mal čítač nastaviť svoj výstup na logickú hodnotu *true*. Ten je použitý v podmienkovom bloku a slúži k povoleniu určitej funkcie po uplynutí

časového intervalu, na ktorý je čítač nastavený.

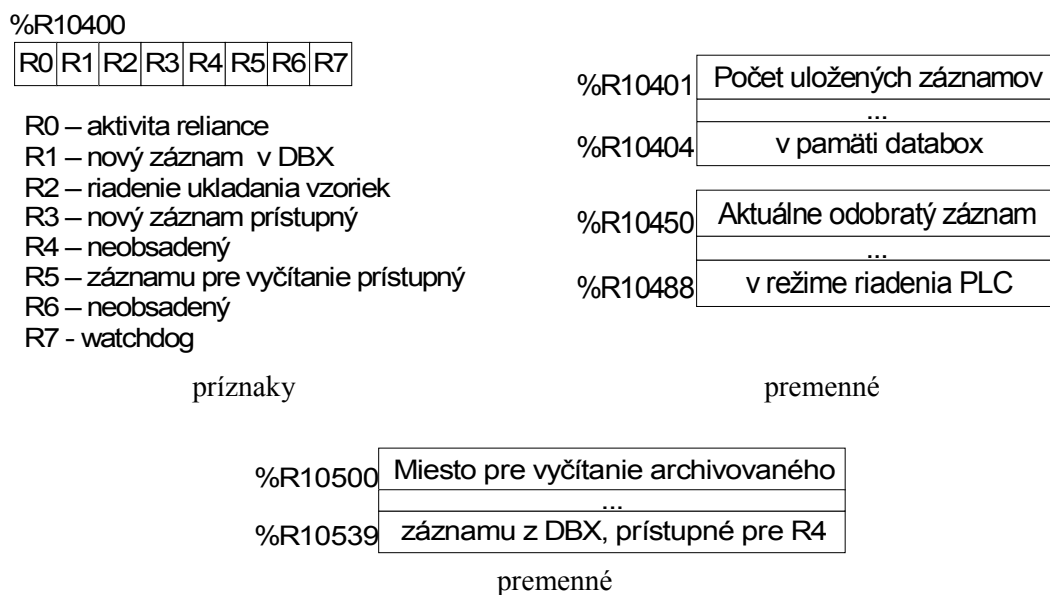
DBX – označuje lokálnu pamäť DataBox, do ktorej sú ukladané záznamy o hodnotách na periférnych snímačoch teploty. Tento stav automatu, kedy preberá riadenie je na (obr. 5.2c).

kontrola R4 – je výkonná rutina, ktorou PLC kontroluje príznakový bit o vzájomnej komunikácii. V podstate ide o jednoduchý princíp kontroly aktivity, ktorý je využitelný aj v reálnom živote⁴. PLC v určitom časovom intervale kontroluje nastavenie príznaku aktivity R4. Ak je aktívny, tak ho nastaví na hodnotu *false* a očakáva sa, že do nasledujúcej kontroly bude vizualizačným programom pre nastavený na hodnotu *true*. Ak sa toto neudeje, PLC to zaznamená ako poruchu a preberá kontrolu nad systémom.

výber z DBX – myslí sa tým prečítanie najstaršieho archivovaného záznamu z prípadu na (obr. 5.2c) a uloženie na predom určenú adresu ktorá je prístupná pre R4.

ulož záznam – uloží aktuálny stav periférií ako záznam do DBX, ktorý bude neskôr vybratý.

čítaj periférie – je tým myslené spustenie rutiny pre komunikáciu s perifériami, a postupné zozbieranie dát ok každej z nich. Dáta sa uložia na špecifické miesto v pamäti PLC, z ktorého bude tento aktuálny stav periférií prístupný aj pre R4.



Obrázok 5.5: Rozdelenie pamäťového priestoru pre spoločnú komunikáciu medzi R4 a PLC

Pozn.: Znak % označuje absolútnu adresu od začiatku pamäťového priestoru registra R v bytoch.

⁴ Pri vedení vlaku musí strojvodca do určitého časového intervalu stlačiť „tlačidlo bdelosti“. V opačnom prípade vlak zastaví.

5.2 Návrh riadiaceho programu v RELIANCE

5.2.1 Dekompozícia problému

Oproti vytvoreniu riadiaceho algoritmu pre PLC, kde nie je potreba veľmi dbať na grafickú stránku veci (skôr na čitateľnosť a pochopiteľnosť kódu), sa požiadavky pre vytvorenie vizualizácie postavili do inej pozície. Keďže sa vytvára užívateľské prostredie, je potreba klásť dôraz na jednoduchosť a prehľadnosť. Okrem týchto kritérií je ďalším problémom riešenie užívateľsky latentných riadiacich algoritmov, akým je hlavný algoritmus archivácie dát, problém komunikácie a výmeny dát medzi PLC a R4 a ich interpretovanie v užívateľsky zrozumiteľnej podobe, teda prostredníctvom grafu. Keďže sa pracuje práve v prostredí akým je R4, a ten je určený práve pre riešenie týchto problémov, je v tomto prostredí dostatok mechanizmov pre jednoduchú implementáciu, ktoré uľahčia riešenie mnohých komplexných problémov. No tieto nástroje sú všeobecné a pre ich použitie je nutná rozsiahla konfigurácia. Všetky užívateľské algoritmy implementované v R4 sa prevádzajú pomocou skriptov. Je to primitívne programovanie a pre zložité algoritmy (prehľadavanie databáze a práca s ňou, tvorba grafov) je obmedzené, takže je nutné sa držať v rámci možností R4.

Myšlienka s archivovaním dát bola priblížená v podkapitole 5.1.1. V podstate ide o implementáciu archivácie práve v tomto systéme. Pre čo najefektívnejšie využitie celého systému a zmeranie čo najväčšieho počtu vzoriek dát z okolitých periférií sa archivácia rozdelí medzi dva riadiace algoritmy. Obidva algoritmy budú medzi sebou komunikovať a v závislosti na správaní každého prispôbovať svoju činnosť. Za hlavný je považovaný algoritmus implementovaný v R4, ktorý spracováva už konečné výsledky. Tieto sa zhromažďujú v jeho databázi s možnosťou ich neskoršieho výberu.

Databázy implementované v R4 majú svoju špecifickú štruktúru, a pri jej zmene by nastali problémy so spätnou interpretáciou. Preto sa pre tieto účely ponechala pôvodná štruktúra. Pridávanie aktuálnych dát do databázy (ďalej len DB) sa bude riadiť časovým skriptom. To znamená, že aktuálne dáta ktoré sa práve nachádzajú na obrazovke vizualizačného okna sa v jednom časovom okamihu zapíšu do DB a to sa bude realizovať cyklicky. Dáta sú ukladané do databázového súboru uloženého lokálne na PC v ktorom je R4 inštalovaný. Pre lepšiu orientáciu v nameraných dátach sa archivačný súbor vytvára cyklicky a to s periódou 1x za mesiac.

Problém komunikácie v sebe zahŕňa aj riešenie požiadavkov na PLC a oznamovanie priebehu spracovania podľa (obr. 5.1). Ako najprijateľnejšie je riešenie komunikácie cez nastavovanie príznakov. Príznakové bity majú svoje definované miesto v pamäti PLC a v rámci monitorovacieho systému má každý svoj špecifický význam. Prístupové práva k týmto

bitom sú nastavené v rámci logiky komunikácie medzi systémami (obr. 5.2). Ich kontrolovaním a nastavovaním pri vykonávaní algoritmu na oboch stranách sa dosiahne že riadiace mechanizmy môžu nastavovať svoju činnosť podľa správania sa toho druhého. Teda ako už bolo spomínané, po spustení R4 sa má monitorovací program zaregistrovať u PLC. To prebehne nastavením príznaku o prítomnosti zo strany R4. Po zaregistrovaní tohoto príznaku na strane PLC sa spustí mechanizmus výberu dát z lokálnej pamäti a nastaví sa príznakový bit oznamu nových dát pripravených na odber. Tento bit zároveň slúži ako príznak či je záznam už uložený v hlavnej databáze alebo nie. R4 má právo tento bit zrušiť ak bol pripravený záznam uložený. To je podnet pre PLC na vyčítanie ďalšieho záznamu z lokálnej pamäte DataBox. Cyklus sa opakuje. Popri tom je potrebné zabezpečiť atomicitu tohoto cyklu výberu dát, pretože pri náhodnom alebo úmyslenom prerušení vyčítavania je DataBox automaticky prepísaný novými dátami.

POZN.: Pre kooperáciu týchto mechanizmov musí byť presne špecifikovaná časť pamäte v PLC ktorá posluží ako komunikačný kanál. Toto rozdelenie je na (obr. 5.5). Pre vytvorenie tejto pamäte je využitý register *R*, označovaný ako remanentná zóna. Pomocou záložného zdroja ostáva zachovaný aj po výpadku prúdu. Jeho hodnoty sa nezmenia ani po prechode automatu do režimu *HALT*.

5.2.2 Popis hlavných skriptov

Naproti funkciám, ktoré boli použité pri riešení riadiaceho algoritmu PLC, sa v tomto prípade volil systém procedúr. Poväčšine sú vo vizualizačnom programe využité skripty, ktoré niečo zobrazujú, alebo nastavujú hodnoty premenných pre zobrazenie, poprípade zobrazujú vizualizačné okná. Pri týchto akciách nie je dôležitá návratová hodnota o úspešnosti operácie. Pri prípadnej chybe vykonávania skriptu nedôjde k zrúteniu riadenia systému. Maximálne môže dôjsť k nezobrazeniu požadovaných výsledkov a táto chyba sa hlási užívateľovi pomocou chybového hlásenia. To ale nemá také fatálne následky, aké by mohli nastať pri neošetrení poruchy programu PLC. Preto sa pri programovaní týchto skriptov nebral až taký ohľad na spätnú kontrolu správnosti výsledku. Pre predstavu o typoch a použití budú uvedené len skupiny skriptov a ich rozdelenie podľa funkčnosti ktorú v programe vykonávajú. Presný popis je uvedený priamo vo vývojovom prostredí R4 a je označený ako popis skriptu.

Skripty využité v tomto vizualizačnom programe sa dajú rozdeliť do troch skupín.

- 1) *Skripty pre nastavovanie hodnôt premenných* – tieto nastavujú premenné pre zobrazenie, poprípade nastavujú príznakové bity PLC, podľa ktorých sa rozhoduje o vetve vykonávaného algoritmu.
- 2) *Skripty pre spúšťanie a ukončovanie okien* – slúžia k zobrazeniu rôznych dialógových okien

a v prípadoch, kedy sa nedá jedným tlačidlom v programe vykonať viacero akcií naráz. Sú to prípady už spomínaného čiastočného zabezpečenia atomicity výberu dát z PLC. V tomto prípade je pri pokuse o ukončenie projektu v prípade že sa vykonáva vyčítavanie dát zobrazené varovné okno a až po jeho potvrdení je zobrazený ukončovací dialóg. Takéto zobrazenie sekvencie okien by pomocou akcií⁵ v R4 nebolo možné realizovať pre rôzne vstupné podmienky a „jedným klikom myši“.

3) *Skripty pre archivovanie dát* – sú skripty ktoré slúžia na ukladanie či už aktuálnych dát, alebo k prenosu a uloženiu dočasne archivovaných dát z PLC do hlavnej databázy. V tomto prípade sa jedná o časovo riadené skripty, spúšťané periodicky podľa vstupných podmienok. Vstupnými podmienkami sa rozumejú príznakové bity, nastavované riadiacim algoritmom PLC, alebo skriptami z R4, ktoré sú popísané vyššie.

Všetky tieto skripty sú spúšťané podnetom od užívateľa, časovo, alebo udalosťou, akou je napríklad zachytenie spádovej, resp. nábežnej hrany určitej premennej. Jedným z možných prípadov je pridanie dočasne archivovaných dát z PLC do hlavnej databázy. Vtedy sa nastaví príznakový bit indikujúci dáta v DBX na hodnotu 0. Zaregistruje sa spádová hrana a stav sa oznámi užívateľovi ako systémové hlásenie pomocou skriptu reagujúceho na túto udalosť.

5.3 Vizualizácia

5.3.1 Popis vizualizačných okien

Z dôvodu obmedzeného rozsahu tohoto dokumentu nebude robený detailný popis vizualizačného okna a jeho funkčnosti. Pre tieto účely slúži nápoveda vyvolaná po spustení vizualizácie vytvorená ako súčasť tejto práce. Pohľad interiéru miestnosti sa volil pôdorysový, aby bola umožnená čo najľahšia orientácia v priestore. Grafické spracovanie zodpovedá 2D zobrazeniu, z dôvodu jednoduchosti a obmedzenej ponuke objektov pre modelovanie v R4. Pre tvorbu 3D pohľadov by bolo nutné vymodelovať detail priestoru v niektorom z modelovacích nástrojov (napr. 3D MAX STUDIO), ktoré jednak neboli k dispozícii a časová náročnosť zvládnutia tvorby modelu je neúmeraná zadaniu práce (jej náplňou bolo zoznámenie sa s nástrojmi R4 a MOSAIC).

V projekte je vytvorených viacero okien, ktorých funkčnosť je obmedzená z dôvodu rozsahu hardvéru pre tento minitorovací systém. Je obmedzený len na jednu miestnosť. Ostatné okná vo vizualizácii boli vytvorené ako demonštrácia možností projektu vytvoreného v R4. Pre účely merania je v tomto projekte využitá miestnosť č.1. Tá sa najviac približuje skutočným

⁵ V R4 neexistujú poslucháči ako to býva v iných programovacích jazykoch, ale dajú sa nadefinovať tzv. akcie. Je to jednoduchšie ako systém poslucháčov, ale funkčnosť je obmedzená len na predom definované úkony.

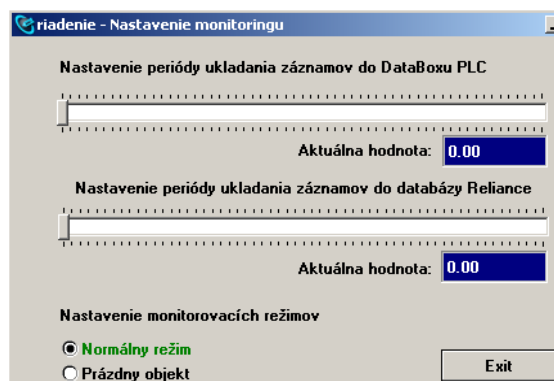
podmienkam inštalácie meracích senzorov. Tie sú inštalované podľa úvah z odstavca 4.1.1. Celé grafické prostredie bolo navrhnuté tak, aby sa ovládalo jednoducho a pokiaľ možno intuitívne pre užívateľa PC. Obrazovka programu je tvorená pôdorysom aktuálnej miestnosti, jednoduchým menu v dolnej, alebo pravej časti obrazovky a tlačidlom pre návrat do hlavného okna. Ostatné ovládacie prvky sa dajú spúšťať z tzv. „scroll-menu“ po kliknutí na pravé tlačidlo myši v pôdoryse miestnosti, alebo na danú zobrazovaciu komponentu. Pre lepšiu predstavu je detail jednej miestnosti znázornený na (obr. 5.7b). V projekte je spracovaná aj tzv. „bublínková nápoveda“, ktorá pomáha pri prvotnej práci s programom. Tá sa zobrazí pri zastavení kurzoru myši na komponente programu. Keďže hlavným cieľom tohoto programu je názorne prezentovať teplebné procesy, ťažiskovými komponentami sú zobrazovacie jednotky, v ktorých sa nachádza aktuálne zmeraná teplota v tej-ktorej miestnosti. Každému zobrazovaciemu prvku je priradená premenná, ktorej hodnotu interpretuje. Tieto premenné sú previazané s premennými v PLC, kde sa nachádzajú obrazy hodnôt z periférií. Tieto hodnoty sú ukladané prostredníctvom skriptu v R4 do databázy. Kliknutím na zobrazovaciu komponentu je možné tieto hodnoty rýchlo z databázy vybrať a interpretovať ich ako časový priebeh. Umiestnenie každého displeja v pôdoryse miestnosti odpovedá jeho skutočnému umiestneniu. Aj keď takáto voľba nie je práve esteticky najoptimálnejšia, slúži k zjednodušeniu orientácie sa v programe.

Okrem merania teplôt a ich ukladania do databázy je v programe zahrnutá aj funkcia počítania štatistiky (jej okno je na obr. 5.6a). Štatistika je počítaná vždy v časovom intervale jedného dňa, podľa zadaného dátumu. Viacej informácií je možné získať z nápovedy k programu. K ďalším zaujímavým funkciám patrí funkcia riadenia ukladania systémových hlásení a nastavovanie intervalu ukladania vzoriek do databázy v R4, rovnako aj interval ukladania vzoriek do DBX v režime autonómneho riadenia PLC. Okno pre toto nastavenie je na (obr. 5.6b) a podrobný popis už v spomínanej nápovede k programu.

5.3.2 Ukážky vizualizácií



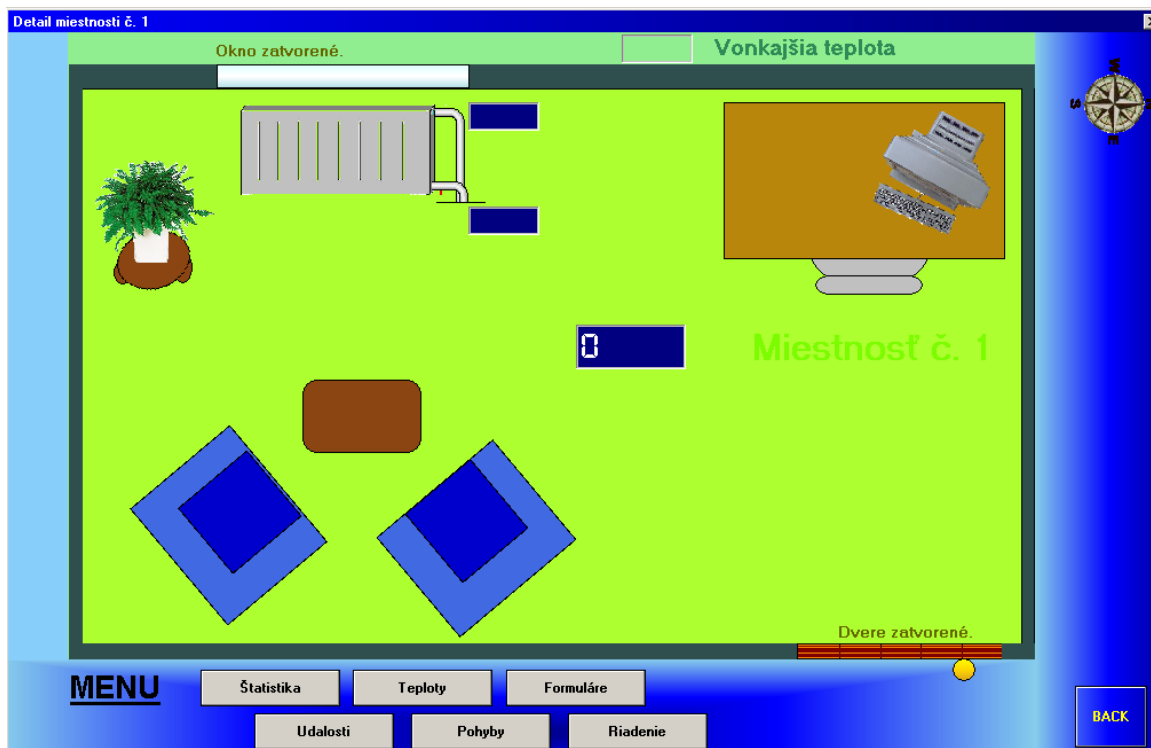
Obrázok 5.6a): Okno pre zobrazenie štatistiky vo vizualizačnom programe R4



Obrázok 5.6b): Okno pre nastavenia vo vizualizačnom programe R4



Obrázok 5.7a): Hlavné okno vizualizačného programu



Obrázok 5.7b): Okno detailu miestnosti so spomínanými indikačnými prvkami

POZN.: Program je v štádiu vývoja, takže okná sa môžu v líšiť v niektorých detailoch oproti verzii uvedenej v tomto dokumente. Obrázky slúžia len pre ilustráciu.

6. Kapitola

6.1 Problémové sekcie

Problém č.1: Časové oneskorenie pri aktualizácii premenných

Diskutabilný bol návrh systému tak, aby sa vytvorila medzi PC a PLC zdieľaná pamäť, cez ktorú sa bude komunikovať. Nakoniec sa pristúpilo na systém nastavovania príznakov v pamäti PLC, tak ako to bolo popísané v odstavci 5.1.1. Problémom pri takejto komunikácii je čas od nastavenia hodnoty na strane R4 za ktorý sa prejaví zmena hodnoty v PLC, a naopak. Tento čas predstavuje (na simulátore PLC) rádovo stovky ms. Pri cyklickom vyčítavaní dát, po znovupripojení R4, pomocou „while“ smyčky dochádzalo k tomu, že sa do prebehnutia ďalšej otočky nestihla prejavíť hodnota o potvrdení prijatia dát zo strany Reliance (príznak R5 z obr. 5.5 sa nenastavil na hodnotu 0) , ktorá bola prenasťavovaná práve v smyčke. Tento príznak bol zároveň aj podmienkou pre pridanie záznamu do databázy. Toto oneskorenie spôsobilo že bolo do hlavnej databázy pridaných niekoľko rovnakých záznamov (smyčka prebehla za čas prenasťavenia príznaku niekoľkokrát), čo spôsobovalo problémy pri neskoršom výbere dát pre zobrazenie grafu.

Riešenie:

Problém sa vyriešil najjednoduchším spôsobom, a to tak, že sa odstránila „while“ smyčka, a skript sa nastavil ako cyklicky spúšťaný v určitých časových intervaloch. Interval bol nastavený na 250ms. Za ten čas sa príznak prenasťaví, a do databázy sa pridá už nová hodnota záznamu. Toto riešenie je však na úkor rýchlosti prenesenia dát z DBX do hlavnej databázy v R4. Predpokladá sa, že ak sa jedná o monitorovací systém, tak rýchlosť vyčítavania dát nehrá dôležitú úlohu. Pretože ak sa už systém R4 spustí, tak sa predpokladá že sa spúšťa na dlhšiu dobu ako je maximálna doba vyčítania celého DBX, čo predstavuje cca. 1 hodinu. Okrem toho vyčítavanie dát nijako neovplyvňuje činnosť celého monitorovacieho systému. Aktuálne dáta sa ukladajú do hlavnej databázy a popri tom sú ukladané aj archívne dáta z PLC. V programe je popri vyčítavaní možné vykonávať všetky bežné činnosti.

Problém č. 2: Databáza v Reliance 4

Spočíva v tom, že pri spomínanom režime monitorovania v R4 a súčasnom vyberaní dát z PLC sa do databázy ukladajú aktuálne a archívne záznamy zároveň. Tým pádom je sa v

databáze prekladajú vety archívneho a aktuálneho záznamu, čím vzniká nezoradená štruktúra archívneho súboru, keďže ten je ukladaný sériovo a jeden takýto súbor obsahuje dáta nazbierané v definovanom časovom intervale. Z pohľadu databázy by toto nemalo spôsobovať žiaden problém, pretože pri výbere si záznamy zoradí. Môže však nastať prípad, že sa dáta vzorkujú do PLC práve na prelome tohoto časového intervalu. Po spustení R4 monitorovací skript začne automaticky ukladať aktuálne dáta do nového databázového súboru, ktorý bude odpovedať nasledujúcemu časovému intervalu ktorý bude pokrytý práve týmto súborom. Okrem toho sa spustí oznam o uložení dát v PLC a začne sa vyčítavanie archivovaných záznamov z pamäte PLC. Tieto záznamy sú neskoršieho dáta, ale ukladajú sa nových tabuliek. Po ukončení vyčítania nastane to, že R4 si tieto novo pridané záznamy pretriedi a po zistení že nepatria do práve aktuálneho súboru pre ukládanie, ich presunie do osobitného súboru a označí ako „DEL“ (vymazané), s ktorými už databáza nepracuje. Pri spätnom pokuse zobrazit' graf v určitom časovom rozsahu odpovedajúcom vymazaným záznamom v grafe vznikajú nespojitosti. Tieto záznamy je možné zobrazit', ale len po manuálnom prepísaní názvu súboru a jeho „vnútení“ databáze.

Riešenie:

Problém sa neodstránil úplne, pretože riadenie práce databázy je pre užívateľa vývojového prostredia latentné. Eliminovať sa však v značnej miere tým, že sa nastavil interval vytvárania archivačných súborov na rozsah jedného mesiaca. Je to priateľné z hľadiska veľkosti vytvoreného súboru a z hľadiska možnosti vzniku tejto situácie. Tá v tomto prípade môže nastať maximálne 1-krát za mesiac, a to na jeho prelome za predpokladu, že nebude aktívny monitorovací systém R4, ale dáta sa budú ukladať do vnútornej pamäte PLC.

7. Kapitola

7.1 Návrhy na vylepšenie

Jedným z vylepšení by bolo umiestnenie hlavnej databázy na externý server SQL, aby bolo možné pristupovať nie len k aktuálnym dátam, ale aj k archivovaným tabuľkám okrem klienta R4, aj cez webové rozhranie prostredníctvom serveru implementovaného v systémoch FOXTROT. Okrem toho, systém MOSAIC obsahuje podprogram zvaný GRAPHMAKER, ktorý dokáže zobrazit' aktuálne dáta ako graf, ale nie je možné tento nástroj použiť online z webového servera implementovaného v tomto systéme. Preto by bolo vhodné v rámci možností prístupu z webu vytvorenie java-appletu, ktorý by s SQL komunikoval a zo získaných dát by bol schopný vykresliť graf pre zvolený čas a miestnosť.

Prínosom z hľadiska rýchlosti programu by bolo aj zlepšenie a zrýchlenie vyčítavania dát z internej pamäte PLC. Ako už bolo spomínané, predpokladá sa dlhodobý monitoring regulovaných priestorov, ale v ojedinelých prípadoch je potrebné odpojenie PC od systému na dlhší čas. Po následnom naviazaní komunikácie PC a automatu trvá výber uložených záznamov dlhšie, čo môže občas spôsobiť čakanie na výsledky merania prevádzaného práve v čase nečinnosti PC s R4.

Ďalším prínosom by bolo v dnešnej dobe vylepšenie grafického dizajnu vizualizačného okna. Užívateľom náročným na 3D a prepracovanú grafiku nemusí vyhovovať (prehľadnejší) 2D pôdorys priestoru.

Nakoniec, ak sa odbočí úplne od zmyslu tejto práce (vytvorenie systému, ktorý zmerá teploty a vykreslí graf), tak pri použití inteligentných INELS prvkov by bola škoda nevyužiť ich funkčnosť naplno. Okrem merania teploty dokážu spínať až 16A, čo by sa mohlo využiť na ovládanie nízkonapäťových zariadení a tak rozšíriť program na malý riadiaci systém inteligentného domčeka. Tým by sa z malého monitorovacieho systému dal vytvoriť silný nástroj pre riadenie chodu domácnosti.

7.2 Záver

Práca nepokrýva v celom rozsahu možnosti systémov Tecomat. Pre tieto účely boli systémy využité len na zlomok možností ich využitia a naprogramovania pre daný účel. Práca slúži ako demonštračný príklad toho, ako jednoducho ide dodať týmto systémom funkčnosť podľa zadaného problému. Celý tento dokument, ktorý problém podľa zadania rieši je rozdelený do ucelených častí a snaží sa ho riešiť a priblížiť. V prvej kapitole je rozobratá teória okolo vytápaných miestností. Kôli rozsahu dokumentu bol naznačený len hrubý postup identifikácie systému, ktorý vychádza z rozsiahlej teórie termodynamiky. Ako príklad implementácie tepelnej regulácie v praxi je uvádzaný dvojstavový regulátor a jeho správanie sa na systémoch, aby bolo možné porovnať zmerané a predpokladané výsledky.

Zmyslom ďalšej časti dokumentu (kapitoly 2 a 3) je priblížiť čitateľovi problematiku PLC a hlavne predstaviť použitý materiál pre vytváranie meracej sady. Sú popísané najdôležitejšie vlastnosti použitého PLC, prvkov INELS 2, a použitého softvéru. Podrobný popis a predstavovanie týchto systémov by bolo zbytočné, pretože k tomu slúžia katalógové listy, tutoriály a návody, ktoré sú k tomu určené. Nosnou časťou práce by mali byť kapitoly 4, 5, 6, kde je vytvorená dekompozícia problému a jeho riešenie. Na základe analýzy sú zvolené veličiny, ktorých sledovanie vyzerá byť účelné. Na jej základe je potom riešené usporiadanie celého systému do reálneho sveta. Kapitola 5 je venovaná analýze problému z programátorského hľadiska. Je rozoberaný problém implementácie programu PLC, tak ako aj implementácia riadiacich skriptov pre vizualizačný program. Okrem toho je riešené usporiadanie vizualizačných okien a funkčnosť ktorá je jednotlivým oknám priradená. Rieši problémy s archiváciou a následnou prezentáciou nameraných dát. Ako je spomínané v kapitole 6, toto riešenie nemusí byť práve najoptimálnejšie, ale postačujúce v rámci zadania a možností softvérových obmedzení.

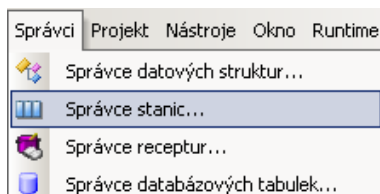
Čo sa týka sprístupnenia dát z internetu, tak predpoklad z úvodu práce, že dáta by mohli byť prístupné aj z vonku, sa nepodarilo splniť. Jednak z dôvodu školskej licencie R4, ktorá v sebe nezahŕňa plné sprístupnenie kontrolného servera (vylúčil sa tak prístup k dátam prostredníctvom R4), a z dôvodu chýbajúceho MMC slotu v PLC (parametre modelu neodpovedajú popisu v [4]). Merané dáta sú teda prístupné len z lokálneho PC, čo pre názornú grafickú prezentáciu postačuje.

Výsledok realizácie už nie je nutné popisovať, pretože sa nachádza na priloženom CD a v prílohe časť C ako výsledok merania v podobe grafov a krátkeho popisu nameraných hodnôt pomocou realizovanej meracej sústavy.

Prílohy

A .Pripojenie PLC k RELIANCE

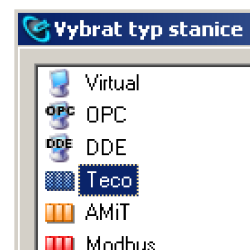
Krok 1



Krok 2



Krok 3



Krok 4

Základní | Databox | Informace

Název
NazovStaniceVRamciProjektu

Alias
[]

Různé
Adresa: 1 | IP adresa: vlastna IP
Zařízení/Model: není definováno

Délka registrů

Registr X	8 192	Registr Y	8 192
Registr S	6 144	Registr R	40 956
Registr M (Databox)	524 288		

Krok 5

Základní | Informace

Název
Mosaic

Alias
[]

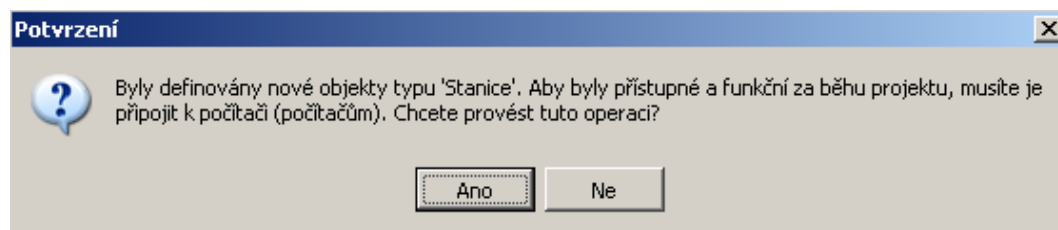
Přenos dat
 Přímý Síťový MEM soubor

Základní | Upřesnit | Driver | Čas | Ostatní

Typ komunikačního kanálu
Mosaic

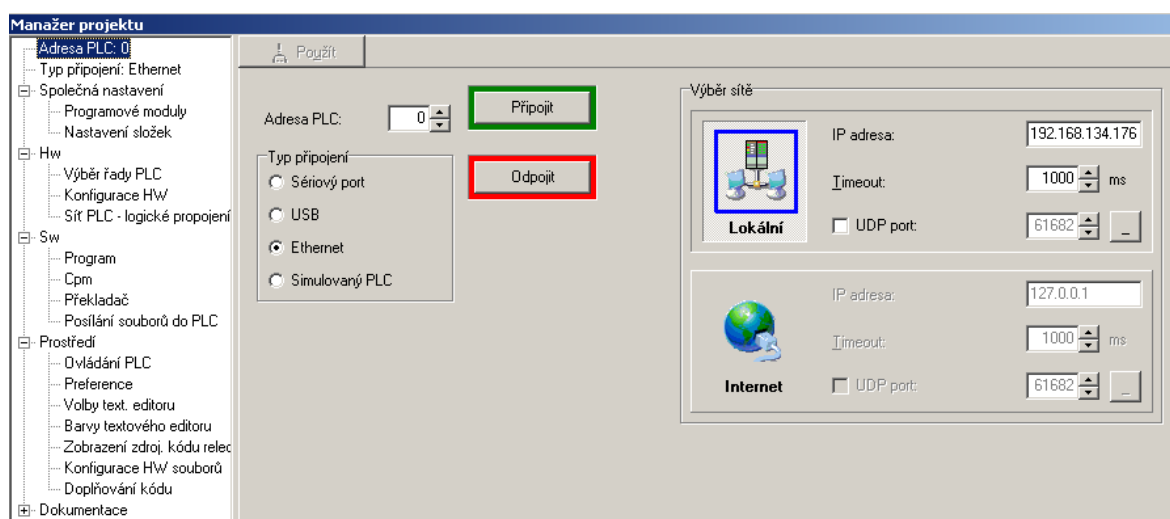
Adresa: 0 | Interval testu komunikace (s): 30
Timeout komunikace (ms): 2000

Skupina projektů
[]

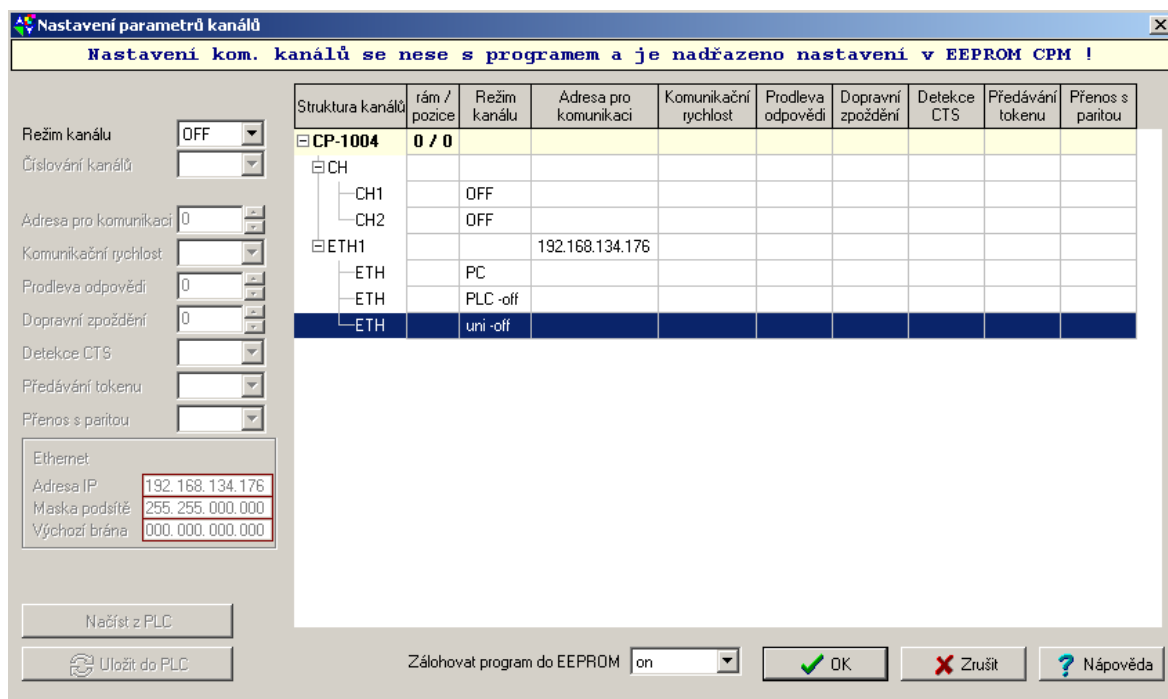


POZN: Pri takomto nastavení je nutné, aby bolo spustené aj vývojové prostredie MOSAIC, ktoré bude slúžiť ako sprostredkovateľ prístupu k pamäťovému priestoru PLC. V prípade že bude vizualizácia spustená skôr ako MOSAIC, spustí si ho automaticky sama.

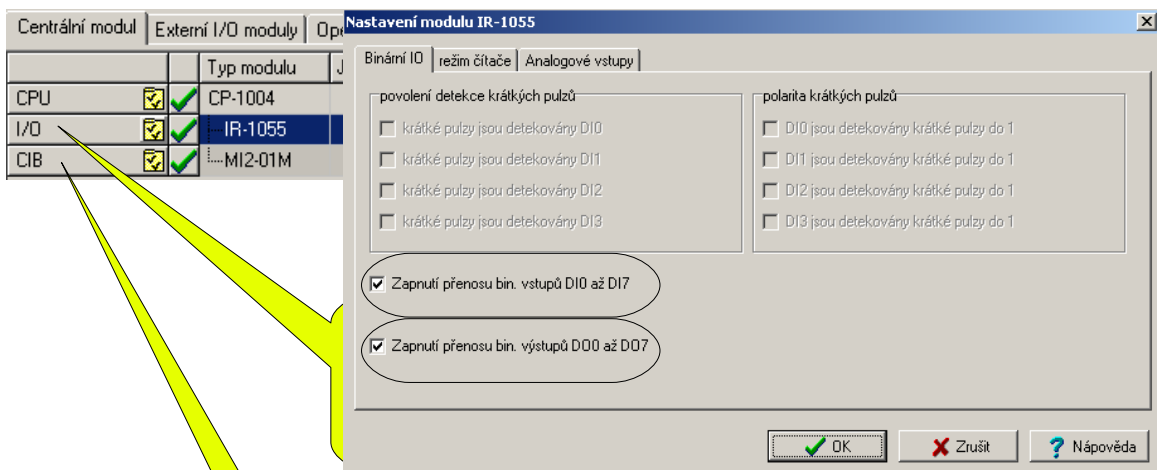
B .Pripojenie PLC k MOSAIC



V tomto prípade pre potreby vývoja programu a prepojenia PLC s PC bolo použité nastavenie uvedené na obrázku. V prípade iného nastavenia siete je možné použitie nastavenia IP pre špecifický prípad.

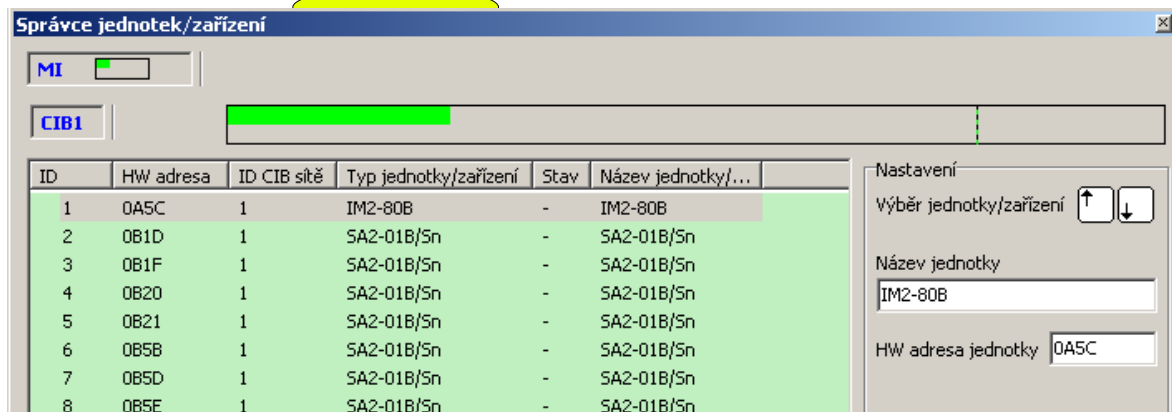


Je potrebné vypnutie nepoužitých komunikačných kanálov, pomocou nastavenia ich režimu na hodnotu OFF. V prípade potreby dodatočných informácií o podrobnejšom nastavovaní je vhodné prezrieť si príručku [9].



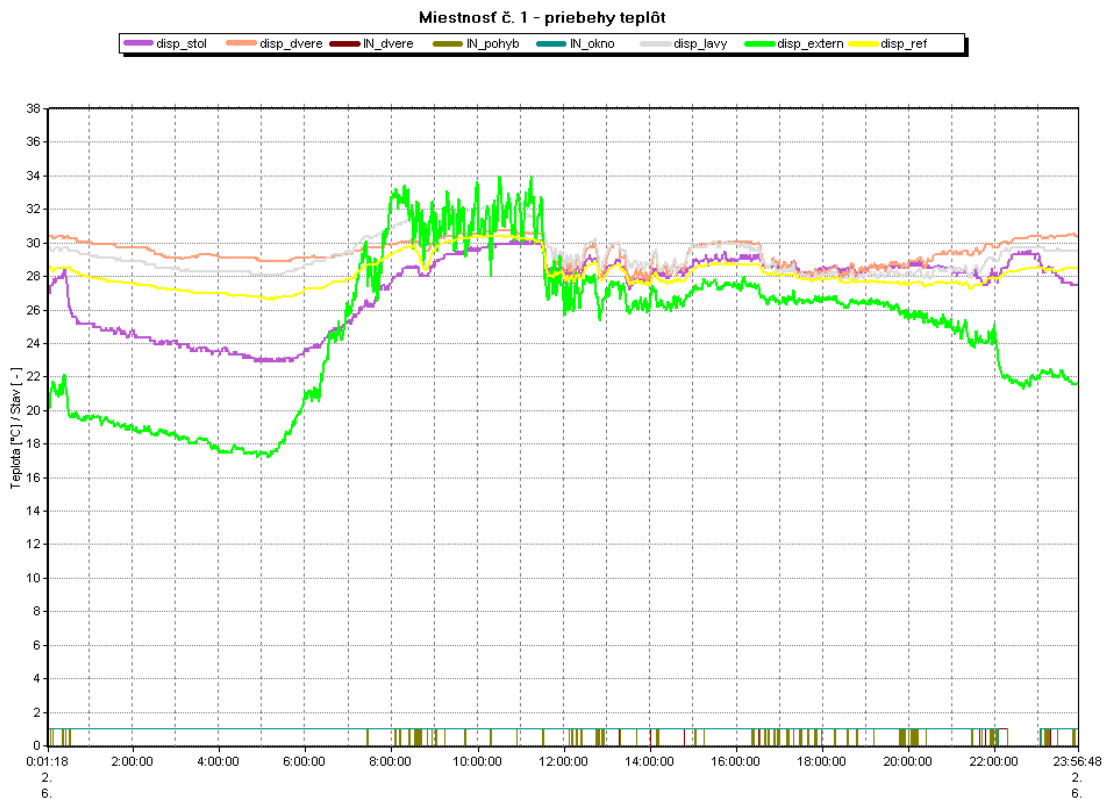
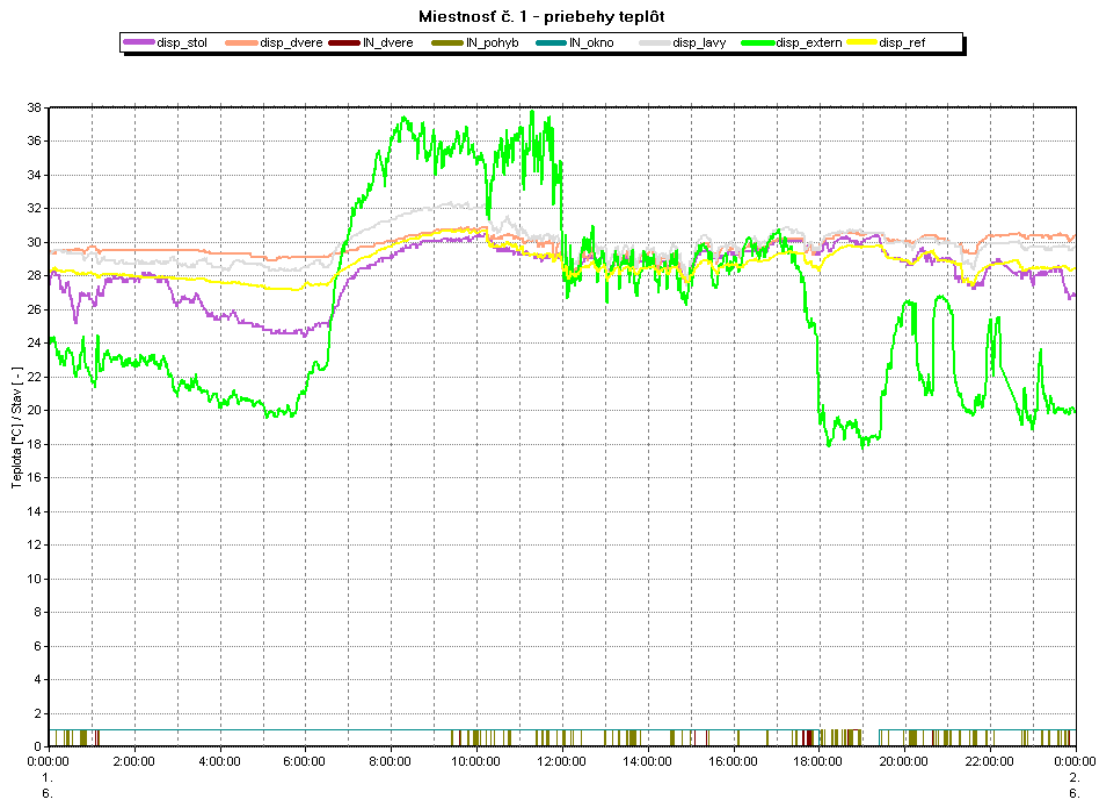
Pre možnosť odskúšania komunikácie, ako aj pre potreby projektu je potrebné povoliť vstupy a výstupy PLC (výstupy v projekte nevyužitú, je možné ich vypnúť po odskúšaní).

NASTAVENIE CIB



Jednotky je možné si pomenovať pre lepšiu prehľad v projekte. V tomto prípade zostávajú jednotky nepomenované, pretože konfigurácia a uloženie fyzického systému sa budú meniť.

C .Výsledky merania



Typické priebehy zmerané v dňoch 01. a 02.06.2008

01.06.2008:

Ako je patrné z nameraných priebehov, jednalo sa o slnečný deň. Najväčšie skoky v tepelnej charakteristike externého senzoru sú zaznamenané v ranných hodinách a v časovom úseku od 18 – 19 hodiny. Rýchle zmeny v ranných hodinách sú spôsobené priamym slnečným žiarením na teplotný senzor, z čoho je možné vydedukovať že je umiestnený na východnej strane na parapete (tmavý podklad). Tepelné špičky sú dôsledkom slnečného žiarenia a absorpcie slnečnej energie. To spôsobí, že sa senzor zahrieva. Malé výkyvy v teplote môžu byť spôsobené zatienením priameho slnečného svetla mrakmi, respektíve pôsobením vetra na senzor, čo spôsobí rýchle ochladenie senzora, pretože teplota okolitého vzduchu je podstate nižšia. Teplotná charakteristika v popoludňajších hodinách je pomerne stála, bez rapidných výkyvov. Pokles teploty je zaznamenaný až v spomínanom čase od 17-19 hodiny. Je to dôsledok dažďa (do cca. 19:30). Následné zmeny teploty v časovom rozsahu od 20 až 24 boli spôsobené prehánkami.

Teplota vo vnútri miestnosti je pomerne stála. Ak sa berie do úvahy referenčná teplota, maximálny tepelný rozdiel v rozsahu jedného dňa je cca. 4 °C. V čase spomínaného dažďa bolo zatvorené okno, čo spôsobilo nárast referenčnej teploty o cca. 2 °C. Po následnom otvorení okna je badateľný pokles teploty.

02.06.2008:

Z grafu je badateľná skutočnosť, že sa znova jedná o slnečný deň, s tým rozdielom, že oproti predchádzajúcemu dátumu nebol zaznamenaný dažď ani prehánky. Tepelné špičky v ranných hodinách majú už vyššie vysvetlený dôvod. V miestnosti otvorené okno počas celého dňa, referenčná teplota pomerne stála. Zaujímavé je sledovať senzor nachádzajúci sa blízko otvoreného okna. Ten v podstatne vyššej miere kopíruje externú teplotu pri otvorenom okne, než referenčný senzor, umiestnený v strede miestnosti. Zaujímavé tiež je, že v nočných hodinách, kedy nie je zaznamenaný pohyb v miestnosti (čas spánku) sa vnútorné teploty menia pomalšie, a majú stálejšiu hodnotu, než v čase pracovného dňa. Tým je potvrdený predpoklad zmien teplôt pri zvýšenom pohybe osôb a vyvetrávaní otvorením dverí, rozobratý v 4 kapitole.

Referencie

- [1] Vobecký, J. - Záhlava, V.: Elektronika: Součástky a obvody, principy a příklady. Třetí, rozšířené vydání. Grada Publishing, a.s., Praha 2005.
- [2] Bayer, J. - Hanzálek, Z. - Šusta, R.: Logické systémy pro řízení. První vydání. Vydavatelství ČVUT, Praha 2000.
- [3] Franklin, G.F. - Powell, J.D. - Emami-Naeini, A.: Feedback control of dynamic systems. Fifth edition. Pearson Prentice Hall, New Jersey 2006.
- [4] Teco, a.s. : Programovatelné automaty TECOMAT FOXTROT. Čtvrté vydání – únor 2008. PDF verze.
- [5] Elko EP, s.r.o.: Katalóg produktů INELS. PDF verze 2006.
- [6] Teco, a.s.: Programování PLC podle normy IEC 61 131-3 v prostředí Mosaic. Desáté vydání – listopad 2007. PDF verze.
- [7] GEOVAP, s.r.o. - Držková, M.: Dokumentace k Reliance 3. PDF verze 2003.
- [8] Teco, a.s.: Příručka programátora PLC TECOMAT. Třinácté vydání – září 2007. PDF verze.
- [9] Teco, a.s.: Začínáme v prostředí MOSAIC. Sedmé vydání – únor 2007. PDF verze.
- [10] Bayer, J. - Šebek, Z. - Píša, P.: Počítače pro řízení. Přednášky. Vydavatelství ČVUT, Praha 2002.
- [11] Baláš, J.: Klientské rozhraní aplikací SCADA. Diplomová práce, ČVUT Praha, září 2006.
- [12] ČVUT, FS.: Vytápění, větrání, instalace – odborný časopis. Praha, listopad 2000.

www.tecomat.cz

www.edumat.cz

www.reliance.cz

<http://ocw.mit.edu>

<http://en.wikipedia.org>

www.tzb-info.cz

Pod'akovanie

Moja veľká vďaka patrí hlavne prof. Ing. Michaelovi Šebekovi, DrSc., vedúcemu katedry riadenia na FEL ČVUT, ktorý uvoľnil finančné prostriedky pre realizáciu tejto práce. Ďalej chcem poďakovať vedúcemu tejto práce, pánu Ing. Martinovi Hlinovskému, Ph.D., ktorý presadil návrh na zakúpenie hardvéru pre realizáciu tohoto projektu a pomáhal pri formálnej stránke veci tohoto dokumentu. Veľká vďaka patrí aj pánu Ing. Ladislavovi Šmejkalovi, CSc., ktorý sa postaral o zabezpečenie objednávky systémov a pomohol pri nejasnostiach v zadaní práce.

Okrem toho patrí vďaka všetkým kolegom z kolejí Strahov, ktorí pomohli kriticky zhodnotiť a otestovať navrhnutý vizualizačný program, poprípade prispeli do realizácie práce požičianím náradia alebo pomohli pri úskaliach s vytváraním ilustrácií.