

**SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE  
FAKULTA CHEMICKEJ A POTRAVINÁRSKEJ  
TECHNOLÓGIE**

EVIDENČNÉ ČÍSLO: FCHPT-5415-61865

**Laboratórny výmenník tepla LTR 700 z pohľadu priemyselného  
riadenia**

**BAKALÁRSKA PRÁCA**

**2015**

**Lenka Panghyová**



**SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE  
FAKULTA CHEMICKÉJ A POTRAVINÁRSKEJ  
TECHNOLÓGIE**

**Laboratórny výmenník tepla LTR 700 z pohľadu priemyselného  
riadenia**

**BAKALÁRSKA PRÁCA**

FCHPT-5415-61865

Študijný program: automatizácia, informatizácia a manažment v chémii a potravinárstve

Číslo študijného odboru: 2621

Názov študijného odboru: 5.2.14 automatizácia, 5.2.52 priemyselné inžinierstvo

Školiace pracovisko: Ústav informatizácie, automatizácie a matematiky

Vedúci záverečnej práce/školiťel': Ing. Richard Valo, PhD.

Konzultant: Ing. Martin Kalúz

**2015**

**Lenka Panghyová**





## ZADANIE BAKALÁRSKEJ PRÁCE

Študentka: **Lenka Panghyová**  
ID študenta: **61865**  
Študijný program: **automatizácia, informatizácia a manažment v chémii a potravinárstve**  
Kombinácia študijných odborov: **5.2.14. automatizácia, 5.2.52. priemyselné inžinierstvo**  
Vedúci práce: **Ing. Richard Valo, PhD.**  
Konzultant: **Ing. Martin Kalúz**

Názov práce: **Laboratórny výmenník tepla LTR 700 z pohľadu priemyselného riadenia**

Špecifikácia zadania:

Tento výmenník tepla má dva merateľné výstupné signály:

- teplota,
- prietok vzduchu

ako aj dva vstupné signály:

- otáčky ventilátora,
- ohrev.

Výmenník bude riadený pomocou PLC od spoločnosti VIPA(CPU) a SIEMENS(I/O modul).

PLC je relatívne malý v priemysle používaný počítač na riadenie procesov v reálnom čase – riadenie strojov, alebo výrobných liniek v továrni. PLC sú odlišné od bežných počítačov nielen tým, že spracovávajú program cyklicky ale aj tým, že ich periférie sú priamo prispôsobené na pripojenie sa k technologickým procesom.

Na priblíženie a uľahčenie procesu riadenia sú k dispozícii priemyselné vizualizačné prostriedky tak na lokálnej (napr. Na LCD) ako aj vzdialenej úrovni(napr. web-stránka).

Úlohy:

1. Zoznámenie sa s procesom. (opis správania. Identifikácia)
2. Prepojenie PLC s výmenníkom. (opis použitých modulov a zapojenie)
3. Návrh riadiacich algoritmov a celkovej štruktúry režimov. (riadenie ventilátora, riadenie špirály. režim kombinujúci obe slučky)
4. Je nevyhnutné koordinovať prácu s ostatnými študentami na danom PLC.
5. Vkládanie vybraných premenných PLC aj do priemyselného routera EWON.
6. Návrh a tvorba vizualizácie na LCD.

Rozsah práce: 30

Zoznam odbornej literatúry:

1. Support, e. Use of Topics for Tag Polling. [online]. 2008. URL: [http://support.ewon.biz/download/KB/KB-0021-0-EN-\(Use of Topics for Tag Polling\).pdf](http://support.ewon.biz/download/KB/KB-0021-0-EN-(Use%20of%20Topics%20for%20Tag%20Polling).pdf).
2. Support, e. eWON General Reference Guide. [online]. 2009. URL: <http://support.ewon.biz/download/RG/RG-001-0-EN-%28General%20Reference%20Guide%29.pdf>.
3. Support, e. eWON Web Reference Guide. [online]. 2008. URL: [http://support.ewon.biz/download/RG/RG-003-0-EN-\(Web Reference Guide\).pdf](http://support.ewon.biz/download/RG/RG-003-0-EN-(Web%20Reference%20Guide).pdf).
4. GmbH, V. VIPA System 300S manual. [online]. 2009. URL: [http://www.vipa.com/uploads/tx\\_sbdownloader/HB140e\\_cpu\\_sc\\_314-6CG03\\_09-45.pdf](http://www.vipa.com/uploads/tx_sbdownloader/HB140e_cpu_sc_314-6CG03_09-45.pdf).
5. AG, S. S7-300 Module data. [online]. 2013. URL: [https://support.automation.siemens.com/WW/llisapi.dll/csfetch/8859629/s7300\\_module\\_data\\_manual\\_en-US\\_en-US.pdf](https://support.automation.siemens.com/WW/llisapi.dll/csfetch/8859629/s7300_module_data_manual_en-US_en-US.pdf).
6. Oravec, J. – Bakošová, M. PIDTOOL 2.0 – Software for Step- response-Based Identification and PID Controller Tuning. *AT&P Journal Plus* č. 2/ 2011 plus. s. 61–66. ISSN 1336-5010.

Riešenie zadania práce od: 17. 02. 2014

Dátum odovzdania práce: 24. 05. 2015

L. S.

**Lenka Panghyová**

šudentka

**prof. Ing. Miroslav Fikar, DrSc.**

vedúci pracoviska

**prof. Ing. Miroslav Fikar, DrSc.**

garant študijného programu

## **Pod'akovanie**

Týmto spôsobom by som sa chcela poďakovať svojmu vedúcemu Bakalárskej práce Ing. Richardovi Valovi, PhD, za odborné vedenie a rady , ktoré mi venoval počas mojej práce. Ďalej ďakujem Ing Martinovi Kalúzovi za sprístupnene vzdialeného systému.



# Abstrakt

Cieľom Bakalárskej práce bolo zoznámiť sa s procesom Laboratórneho výmenníka tepla LTR 700. V prvej časti sme sa zaoberali problematikou PLC. V druhej časti sme sa zaoberali prepojením PLC s laboratórnym výmenníkom tepla LTR700 cez moduly, s počítačom cez MPI a nakoniec so vzdialeným systémom cez priemyselný router eWON. Navrhli sme základné algoritmy riadenia, ktoré sme aplikovali na riadenie teploty a prietoku výmenníka tepla. Algoritmy sme prostredníctvom softvéru WinCC vizualizovali a aplikovali na počítačovú obrazovku. Posledným krokom bolo vloženie premenných využitých v algoritmoch riadenia do vzdialeného systému. Vzdialený systém študenti môžu využívať na štúdium procesov, ktoré sa môžu využívať v praxi. Pomocou vzdialeného systému sme identifikovali proces výmenníka tepla.

**Kľúčové slová:** Výmenník tepla LTR 700, PLC, vizualizácia, eWON,

# Abstract

The aim of the Bachelor thesis was to get acquainted with the process of Laboratory air heat exchanger LTR 700. In the first part we dealt with the issue of PLC. In the second part we dealt with connection of PLC with laboratory air heat exchanger through modules, with a computer through MPI and finally with a remote system through an industrial router eWON. We proposed basic control algorithms that we applied to control the temperature and flow of the air heat exchanger. We visualized algorithms by software WinCC and we applied them to a computer display. In the last step we inserted variables used in algorithms to a remote system. A remote system may be used by students to study the processes that can be used in practice. By means of a remote system we identified the process of the air heat exchanger.

**Key words:** air heat exchanger LTR 700, PLC, visualization, eWON

# Obsah

|                                                                               |    |
|-------------------------------------------------------------------------------|----|
| Zoznam skratiek a značiek .....                                               | 13 |
| Úvod.....                                                                     | 14 |
| 1 Súčasný stav riešenej problematiky .....                                    | 15 |
| 1.1 Laboratórny výmenník tepla LTR 700.....                                   | 15 |
| 1.2 PLC z hľadiska jeho využitia.....                                         | 15 |
| 1.2.1 Architektúra PLC .....                                                  | 16 |
| 1.2.2 Centrálna jednotka CPU .....                                            | 16 |
| 1.2.3 Programovanie PLC .....                                                 | 17 |
| 1.2.4 Základný popis triedy 300s od spoločnosti VIPA.....                     | 18 |
| 1.3 Simatic Step 7 .....                                                      | 20 |
| 1.3.1 Štruktúra Simaticu Step7.....                                           | 20 |
| 1.3.2 Programovací jazyk LAD .....                                            | 21 |
| 1.4 PID regulátor.....                                                        | 22 |
| 1.4.1 Označenie parametrov PID regulátora .....                               | 22 |
| 1.4.2 Bloková schéma PID regulátora v STEP 7 .....                            | 23 |
| 1.5 WinCC .....                                                               | 25 |
| 2 Cieľ práce .....                                                            | 27 |
| 3 Metodika práce a metódy skúmania .....                                      | 28 |
| 3.1 Zapojenia a moduly .....                                                  | 28 |
| 3.1.1 Vyhodnocovanie a generovanie signálov.....                              | 29 |
| 3.2 Konfigurácia .....                                                        | 29 |
| 3.3 Identifikácia procesu podľa Strejcovej metódy.....                        | 30 |
| 4 Výsledky práce .....                                                        | 32 |
| 4.1 Základné algoritmy riadenia .....                                         | 32 |
| 4.1.1 Algoritmus pre manuálny režim riadenia teploty a prietoku vzduchu ..... | 32 |

|       |                                                                          |    |
|-------|--------------------------------------------------------------------------|----|
| 4.1.2 | Algoritmus pre inverzný režim riadenia prietoku vzduchu .....            | 32 |
| 4.1.3 | Algoritmus pre relé režim riadenia teploty a prietoku vzduchu .....      | 34 |
| 4.1.4 | Algoritmus pre PID režim riadenia .....                                  | 34 |
| 4.2   | Identifikácia procesu .....                                              | 35 |
| 4.3   | Vizualizácia procesu .....                                               | 36 |
| 4.3.1 | Postup zostrojenia vizualizačných obrazoviek .....                       | 36 |
| 4.3.2 | Postup zobrazenia vizualizačných obrazoviek pre algoritmy riadenia ..... | 37 |
| 4.4   | Vzdialený prístup riadenia eWON .....                                    | 40 |
| 4.4.1 | Charakteristika Modemu eWON .....                                        | 40 |
| 4.4.2 | Vkladanie premenných z PLC do eWONu .....                                | 40 |
| 5     | Diskusia .....                                                           | 43 |
| 6     | Záver .....                                                              | 45 |
| 7     | Zoznam použitej literatúry .....                                         | 46 |
| 8     | Príloha .....                                                            | 47 |

# Zoznam skratiek a značiek

PLC- Programovateľný logický automat  
CPU- Central Processing Unit (centrálna procesorová jednotka)  
DPM- Data Protection Manager (ochrana dát)  
MPI- Multi Point Interface (rozhranie na výmenu informácií s počítačom)  
MMC- Memory Multimedia Card (pamäťová karta)  
RAM- Random Access Memory (pamäť s priamym prístupom)  
PB- profibus  
CH- chanell  
STL- statement list (popis prostredníctvom slovných príkazov)  
LAD- ladder logic (rebríková logika)  
FBD- function block diagram (blokový diagram)  
PID – proporcionálne- integračne- derivačný regulátor  
WinCC- softvér slúžiaci na vytvorenie vizuálneho zobrazenia procesu  
ADD- sčítanie  
MUL- násobenie  
DIV- delenie  
SUB- odčítanie  
SP- set point (žiadaná hodnota)  
 $e(t)$ - regulačná odchýlka v čase  $t$   
 $u(t)$ - vstupná veličina v čase  $t$   
 $y(t)$ - výstupná veličina v čase  $t$   
Z- zosilnenie  
T- časová konštanta  
 $Z_R$ - zosilnenie regulátora  
 $T_I$ - integračná konštanta  
 $T_D$ - derivačná konštanta  
OB- organizačný blok  
FB- funkčný dátový blok  
DB- dátový blok  
FC- funkcie  
SFB- systémový funkčný blok  
SFC- blok pre systémové funkcie

# Úvod

V dnešnej dobe sa stretávame s tým, že mnoho podnikov nahrádza človeka automatizačnou technikou. Dôvody nahradenia môžu byť: zvýšenie produktivity práce, zvýšenie objemu výroby, zníženie výrobných nákladov, odstránenie nežiaducich chýb vytvorené človekom, sledovanie veľkého počtu veličín a procesov. V bakalárskej práci sme sa rozhodli aplikovať programovateľný logický automat na proces laboratórneho výmenníka tepla LTR 700.

Laboratórny výmenník tepla LTR 700 sa skladá z dvoch častí. Prvá časť reprezentuje trubicu, v ktorej je umiestená špirála a ventilátor. Druhá časť reprezentuje skrinku, v ktorej zabezpečujeme prepojenie trubice s PLC. Pomocou laboratórneho výmenníka tepla sme riadili dve výstupné veličiny. Pomocou otáčok ventilátora sme riadili prietok vzduchu. Pomocou výkonu ohrevu špirály sme riadili teplotu vzduchu.

V prvej časti bakalárskej práce sme sa zaoberali popisom Laboratórneho výmenníka tepla LTR 700. Popisovali sme teóriu vzniku a využitia PLC v automatizačnej technike. Zaoberali sme sa využitím programovacieho jazyka pri navrhovaní algoritmov riadenia a vizualizačného softvéru.

V metodike bakalárskej práce sme sa zaoberali prepojením PLC s laboratórnym výmenníkom tepla cez moduly, s počítačom cez MPI a web stránky cez Ethernetový modul. Vo výsledkoch bakalárskej práce sme navrhovali základné algoritmy, ktorými sme riadili výstupné veličiny výmenníka tepla. Po vytvorení algoritmov riadenia sme ich zobrazili pomocou vizualizácie na obrazovku. Posledným krokom pri úspešnej realizácii bakalárskej bolo vloženie premenných využitých v algoritmoch riadenia do vzdialeného systému. Vo vzdialenom systéme sme vytvorili len manuálny režim riadenia. Prostredníctvom manuálneho riadenia sme boli schopní proces laboratórneho výmenníka tepla LTR 700 identifikovať. Pomocou premenných, ktoré sú vložené vo vzdialenom systéme, študenti môžu vytvoriť ďalšie algoritmy na riadenie laboratórneho výmenníka tepla.

Na základe realizovaných pokusov na laboratórnym výmenníku tepla sa môžu navrhovať výmenníky pre praktické využitie v reálnych systémoch. Najčastejšie využitie výmenníkov tepla môže byť udržiavanie teploty v miestnostiach, v laboratóriách. V prípade laboratórneho výmenníka tepla poskytnutého školou sme robili experimenty na základe zvyšovania prietoku a zvyšovania teploty vzduchu pretekajúceho cez zariadenie. Ďalšou možnosťou využitia navrhnutého procesu môže byť, zvyšovanie prietoku vzduchu, v dôsledku ktorého sa zníži teplota vzduchu pretekajúceho cez výmenník tepla. Tento proces má aplikáciu pri klimatizačných zariadeniach.

# 1 Súčasný stav riešenej problematiky

## 1.1 Laboratórny výmenník tepla LTR 700

Laboratórny výmenník tepla LTR 700 je zariadenie určené na riešenie problematiky riadenia procesov. Samotné zariadenie je rozdelené na dve časti. Prvá časť reprezentuje zariadenie výmenníka tepla ako trubicu. Druhá časť zobrazuje schému zariadenia a jeho zapojenie. Pomocou laboratórneho výmenníka možno zaznamenávať dve výstupné veličiny, a to prietok a teplotu vzduchu. Získaný interval hodnôt teploty sme získali prostredníctvom výkonu ohrevu špirály. Interval hodnôt prietoku vzduchu sme získali pomocou otáčok ventilátora. Špirála a ventilátor sú zabudované v trubici laboratórneho výmenníka tepla. Hodnoty prietoku a teploty sme zaznamenávali pomocou senzorov, ktoré sú pripojené k trubici výmenníka tepla. Tlakovým senzorom sme zaznamenávali prietok vzduchu. Teplotu vzduchu sme zaznamenávali senzorom teploty. Na obr.1 sme znázornili zariadenie laboratórneho výmenníka tepla LTR 700.



Obr. 1. Laboratórny výmenník tepla LTR 700

## 1.2 PLC z hľadiska jeho využitia

Prvé programovateľné logické automaty vznikli na začiatku 60-tych rokov 20. Storočia. Ich hlavnou úlohou bolo nahradiť reléovú a bezkontaktnú logiku. V 80-tych rokoch boli programovateľné automaty považované za efektívnejšie riadiace prostriedky na zabezpečenie riadenia strojov, technologických zariadení a výrobných liniek v tej dobe používaných. Tým, že

sa nahradila reléová a neskôr bezkontaktná logika vytvoril sa jazyk , ktorý sa podobal jazyku logických schém, booleovským rovniciam, reléovým schémam a assembleru. Vytvorením jednoduchšieho jazyka sa zabezpečilo, že logické automaty boli efektívnejšie než technika, ktorá fungovala prostredníctvom logického riadenia. Programovateľné logické automaty dokázali logické rovnice naprogramovať, zatiaľ čo reléová a bezkontaktná logika pracovala na základe toho, že logické rovnice fungovali na princípe fyzického prepojenia logických členov.

Výhody programovateľných logických automatov:

- Flexibilita – možnosť vytvárania programov presne na určitý prístroj
- Pripojenie modulov
- Rýchlo programovateľné úlohy
- Hospodárnosť – nízke náklady na prevádzku, lacnejšie oproti inej automatizačnej technike
- Jednoduché programovanie
- Množstvo produktov od rôznych výrobcov zaoberajúcimi sa PLC
- Odolnosť voči poruchám
- Vstupnou – výstupnou jednotkou sú schopné spracovať unifikovaný signál
- Jednoduchá údržba v prevádzke [1]

### **1.2.1 Architektúra PLC**

Základom programovateľného logického automatu tvorí centrálna procesorová jednotka CPU, ktorá môže obsahovať vstupy a výstupy pre analógový alebo digitálny signál. Výhodou programovateľného logického automatu je možnosť pripájať analógové, digitálne, komunikačné moduly. Pomocou komunikačných modulov je možné zabezpečiť komunikáciu s inými systémami ako je napríklad vizualizácia alebo vzdialenie riadenie technologických procesov.

### **1.2.2 Centrálna jednotka CPU**

Centrálna jednotka realizuje súbor inštrukcií, zaisťuje komunikačné funkcie s vlastnými ale i vzdialenými modulmi, s nadriadeným systémom a s programovacím prístrojom. Pamäťový priestor je rozdelený do častí. Prvá časť slúži na uloženie užívateľského programu, dátových blokov. Obsah sa zadáva v edičnom režime a počas vykonávania programu sa obsah obvykle nemení. Druhú časť reprezentuje zápisník v ktorom sú uložené užívateľské registre, čítače, obrazy vstupov a výstupov, komunikačné, časové a iné systémové premenné. Obsah operačnej časti sa dynamicky mení pôsobením užívateľského a systémového programu [2].

Modul CPU sa skladá z častí:

- Pamäť EEPROM pre operačný systém
- Pamäťové registre
- Pamäť RAM
- Programovacie rozhranie
- Štandardizované rozhranie s protokolom pre sériovú komunikáciu s nižšou a vyššou riadiacou úrovňou
- Môže byť doplnením bitovým procesorom pre rýchle riešenie logických rovníc
- Procesorom, ktorý obsahuje 8,16, 32 bitov

## **1.2.3 Programovanie PLC**

### **1.2.3.1 Užívateľský program , cyklická slučka**

Program PLC predstavuje postupnosť inštrukcií a príkazov jazyka využitých pri programovaní PLC. Základný režim programovateľného logického automatu je cyklické vykonávanie programu v programovej slučke. Oproti inej automatizačnej technike programátor PLC sa nemusí starať o to, aby po konci programu vrátil jeho vykonávanie na začiatok. Túto funkciu zaistí systémový program. V prípade ak program dlhodobo zotrvá v programovej slučke hlási systém prekročenie doby cyklu.

Vždy po vykonaní poslednej inštrukcie užívateľského programu je riadenie odovzdané systémovému programu, ktorý vykoná slučku. V slučke sa aktualizujú najskôr hodnoty vstupov a výstupov. Hodnoty uložené v pamäti ako obrazy výstupov prepíše do registra výstupných periférnych modulov. Hodnoty vstupných modulov kopíruje do pamäťových obrazov vstupov. Aktualizuje časové údaje pre časovače, systémové registre, ošetrí komunikáciu. Po otočke cyklu sa program predá do užívateľského programu.

Program, ktorý je vytvorený v programovateľnom logickom automate sa vykonáva v cykle. Pri spustení programu sú najskôr vykonané aktualizácie systémové a časové premenné, naplánovaná aktivácia programu pre ďalší cyklus. Získajú sa aktuálne hodnoty fyzických vstupov, ktoré sú počas cyklu uložené ako obrazy vstupov. Realizuje sa program a na konci cyklu sú obrazy výstupov premenené na aktuálne hodnoty výstupov.

Dôležité je spomenúť, že PLC nepracuje s aktuálnymi hodnotami vstupov a výstupov, ale sú uložené ako obrazy vstupov a výstupov v zápisníkovej pamäti [2].

## 1.2.4 Základný popis triedy 300s od spoločnosti VIPA

Na úspešné riešenie bakalárskej práce sme vybrali PLC 300s od spoločnosti VIPA. Hlavným predpokladom pre výber tohto typu PLC bola veľká kapacita pamäte. Táto kapacita umožnila pripojiť na to isté PLC viacero prístrojov ako hydraulická sústava uDAQ28/3H a tepelne – optická sústava uDAQ28/LT.

Systém 300s je modulárny riadiaci systém určený pre veľké a na rýchlosť náročné aplikácie. Jadrom radu CPU SPEED7 je špecializovaný čip PLC7000 s multiprocesorovou architektúrou, ktorý zabezpečuje extrémne krátke časy spracovania programu. Všetky procesory sú štandardne vybavené rozhraním MP2I a Ethernet-TCP komunikačným rozhraním pre programovanie a vizualizáciu. Užívateľské programy sa vytvárajú v jazyku Step®7 firmy Siemens [5].

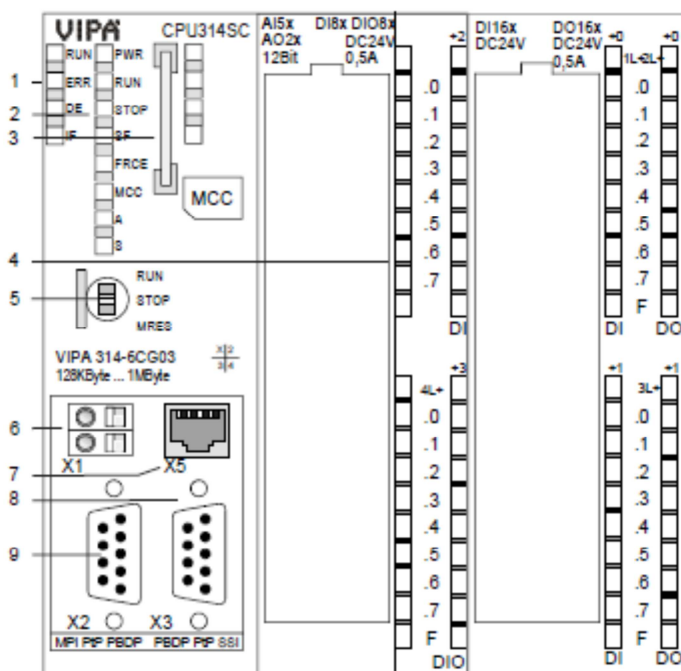
Vlastnosti PLC 300s od spoločnosti VIPA:

- programovanie s WinPLC7 f. VIPA alebo Step®7 f. Siemens a kompatibilita s modulmi S7-300 od firmy Siemens
- integrovaná work/load memory – prevádzka bez dodatočnej MMC karty
- flexibilné rozšírenie pamäte použitím Memory Configuration Card
- integrovaný akumulátor pre zálohovanú RAM
- podpora štandardnej MMC (Multi Media Card) pre zálohu programu a dát
- paralelná zbernica SPEED-Bus umožňujúca použitie High-Speed signalových modulov a komunikačných procesorov
- integrovaný ethernet, Profibus-DP a MPI port
- diagnostická WEB stránka CPU
- hodiny reálneho času
- možná zmiešaná prevádzka modulov VIPA a Siemens v tom istom racku
- až 32modulov v jednom racku k CPU
- Integrovaný Ethernet PG/OP interface

#### 1.2.4.1 CPU 314 SC/DPM

Náše CPU, ktoré sme využili pri programovaní laboratórneho výmenníka tepla sa skladá z častí, ktoré sú zobrazené na obr.2. K vybranému CPU sme mali pripojené vstupné- výstupné analógové moduly, ktoré spracovávali unifikovaný signál vo forme prúdu elektrického signálu v jednotkách (mA). Digitálne moduly, ktoré spracovávali unifikovaný signál vo forme napätia elektrického signálu v jednotkách (V). K CPU sme mali možnosť pripájať ďalšie moduly.

**CPU 314SC/DPM**  
314-6CG03



### Obr. 2 CPU 314SC/DPM

Popis schémy centrálnej procesnej jednotky CPU:

1. LEDs
2. LEDS CPU časti, zobrazenie stavu CPU
3. Priestor pre pamäťovú kartu MMC
4. LEDS I/O časti
5. Prepínací mód tlačidiel CPU
6. Slot pre integrované napájanie DC 24V

7. Ethernetové rozhranie pre PG/OP kanál
8. PB- DP/PtP rozhranie
9. MPI/PtP/PB rozhranie

## 1.3 Simatic Step 7

Simatic Step 7 je software, ktorý umožňuje tvorbu programov hardwarovú konfiguráciu, prenos údajov z PLC alebo do PLC a monitorovanie programu.

Step 7 podporuje tri programovacie jazyky:

- **STL** (Statement list) - programovanie pomocou príkazov
- **LAD** (Ladder Logic) – programovanie pomocou rebríkovej logiky
- **FBD** (Function Block Diagram) - programovanie pomocou blokovej schémy

### 1.3.1 Štruktúra Simaticu Step7

Výhodou Step 7 je možnosť rozdelenia do blokov. Rozdelenie blokov v SIMATIC STEP7:

- Organizačné bloky (OB) – predstavujú rozhranie medzi operačným systémom CPU a programom. Prostredníctvom tohto bloku je možné vyvolať funkčné bloky a funkcie
- Dátové bloky (DB) - bloky, ktoré slúžia na uloženie dát použitých na riadenie samotného procesu. Obsahujú všetky premenné, ktoré sa využívajú v programe
- Funkčné bloky (FB) - blok, kde sa vytvára program, obsahujú pamäť kam sa môžu ukladať vnútorné premenné. Pri vyvolávaní funkčného bloku musíme poznať číslo dátového bloku, ktorý sa automaticky otvorí
- Funkcie (FC) -obsahujú najčastejšie používané funkcie, nemajú pamäť
- Systémové funkčné bloky (SFB) a Systémové funkcie (SFC) - sú integrované priamo v CPU jednotke a umožňujú výrobcovi riešenia rôznych úloh [4].

## 1.3.2 Programovací jazyk LAD

Programovací jazyk LAD je možno použiť:

- editovanie a ladenie programu
- editovanie a monitorovanie programu

### 1.3.2.1 Bitové operácie

Bitové operácie pracujú s binárnymi číslami 1 a 0. 1 predstavuje stav zapnutý, zatiaľ čo 0 predstavuje stav vypnutý

- Priradenie
- Logický súčin (AND)
- Logický súčet (OR)
- SET výstup
- RESET výstup

### 1.3.2.2 Aritmetické operácie:

- Sčítanie (ADD\_) - aritmetická operácia, ktorá slúži na sčítanie dvoch vstupných hodnôt a výsledná hodnota sa uloží na inú adresu
- Násobenie (MUL\_) - aritmetická operácia, ktorá slúži na vynásobenie dvoch vstupných hodnôt a výsledná hodnota sa uloží na adresu
- Delenie (DIV\_) - aritmetická operácia, ktorá slúži na delenie dvoch vstupných hodnôt a výsledná hodnota sa uloží na adresu
- Odčítanie (SUB\_) - aritmetická operácia na sčítanie dvoch vstupných hodnôt a výstupná hodnota sa uloží na konkrétnu adresu

MOVE – umožňuje priradenie hodnoty do premennej. Ak je vstup EN aktívny, skopíruje sa hodnota zo vstupu na adresu pre výstup.

CONVERTER- slúži na porovnanie hodnôt, závisí od typu premennej.

ROUND – zaokrúhľenie premennej

## 1.4 PID regulátor

PID (Proporcionálne – Integračný - Derivačný) regulátor je najčastejšie využívaný v praxi. Úlohou regulátora je zabezpečiť plynulý chod riadenia procesu bez toho, aby doňho ručne zasahoval operátor výroby či linky. V procese by regulátor mal automaticky meniť akčný zásah tak, aby výstupná hodnota sa ustálila na hodnote žiadanej veličiny. Pri regulácii môže vzniknúť aj trvalá regulačná odchýlka, ktorá je definovaná ako rozdiel medzi skutočnou a požadovanou hodnotou výstupnej regulovanej veličiny.

Matematicky možno PID regulátor vyjadriť vzťahom:

$$G_R(s) = Z_R(1 + \frac{1}{T_I s} + T_D s) \quad (1)$$

### 1.4.1 Označenie parametrov PID regulátora

- Proporcionálna zložka
- Integračná zložka
- Derivačná zložka

#### 1.4.1.1 Proporcionálna zložka

Pri proporcionálnej zložke je akčný zásah úmerný regulačnej odchýlke.

$$u(t) = Z_R e(t) \quad (2)$$

V prípade zväčšovania zložky zosilnenia sa môže hodnota regulačnej odchýlky znižovať.

#### 1.4.1.2 Integračná zložka

Pri integrálnej zložke akčný zásah je úmerný dobe, pri ktorej existuje regulačná odchýlka.

$$u(t) = \frac{1}{T_I} \int_0^t e(t) dt \quad (3)$$

Na odstránenie regulačnej odchýlky bolo treba zaviesť integrátor - akčný zásah sa bude meniť dovtedy, kým nebude regulačná odchýlka nulová.

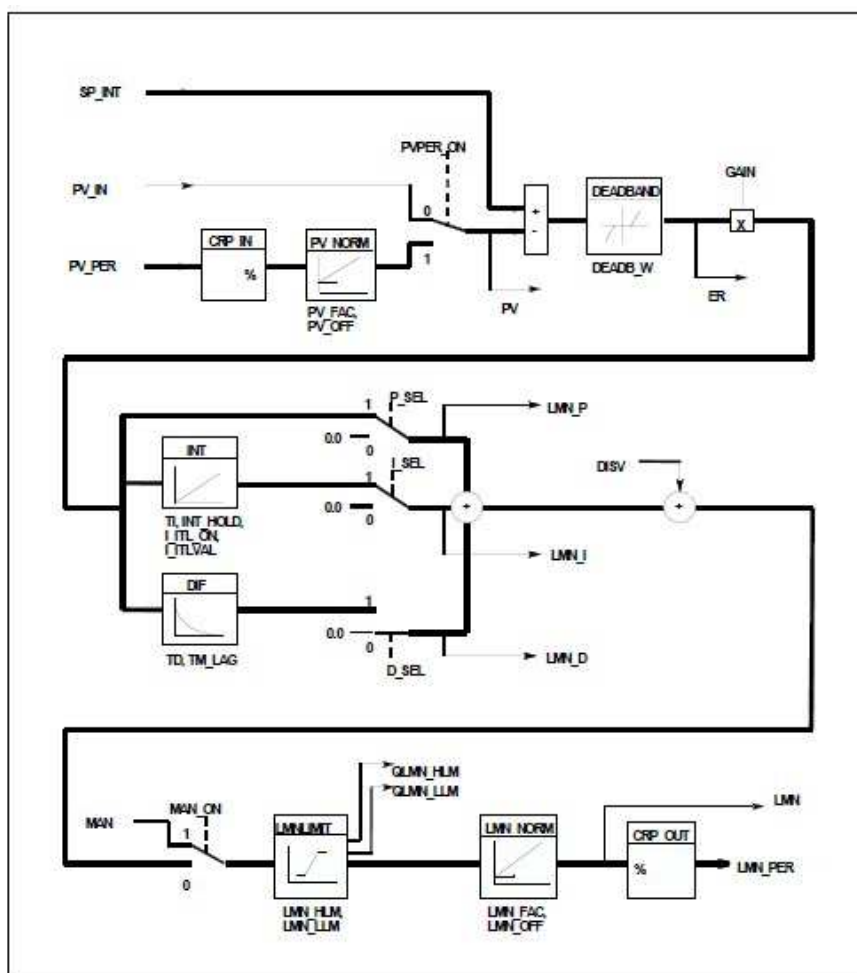
### 1.4.1.3 Derivačná zložka

Pri derivačnej odchýlke je akčný zásah priamo úmerný rýchlosti zmeny regulačnej odchýlky.

$$u(t) = T_D \frac{de(t)}{dt} \quad (4)$$

Využitie spočíva v tom, že v predstihu kompenzuje zmenu regulovanej veličiny, čím spôsobí, že sa utlmia kmity v regulačnom obvode [6].

### 1.4.2 Bloková schéma PID regulátora v STEP 7



Obr. 3 Schéma PID regulátora

**Tab. 1 Opis vstupných veličín PID regulátora**

| OZNAČENIE | POPIS                                                       |
|-----------|-------------------------------------------------------------|
| MAN_ON    | Zapnutie zložky pre zadávanie manuálnej hodnoty             |
| P_SEL     | Zapnutie zosilnenia regulátora                              |
| I_SEL     | Zapnutie integračnej časovej konštanty regulátora           |
| D_SEL     | Zapnutie derivačnej časovej konštanty regulátora            |
| TI        | Hodnota integračnej časovej konštanty                       |
| TD        | Hodnota derivačnej časovej konštanty                        |
| SP_INT    | Vstup žiadanej veličiny zadáva sa hodnota žiadanej veličiny |
| PV_INT    | Meraná veličina                                             |
| GAIN      | Zosilnenie regulátora                                       |
| LMN_HLM   | Horná hranica obmedzenia akčnej veličiny                    |
| LMN_LLM   | Dolná hranica obmedzenia akčnej veličiny                    |
| DISV      | Premenná porúch                                             |

**Tab .2 Opis výstupných veličín PID regulátora**

| OZNAČENIE | POPIS                                                                   |
|-----------|-------------------------------------------------------------------------|
| LMN       | Hodnota akčnej veličiny vystupujúcej z regulátora- Nastaviteľná hodnota |
| LMN_P     | Výstup z P zložky- proporčný koeficient                                 |

|          |                                          |
|----------|------------------------------------------|
| LMN_I    | Výstup z I zložky- integračný koeficient |
| LMN_D    | Výstup z D zložky- derivačný koeficient  |
| ER       | Regulačná odchýlka                       |
| QLMN_HLM | Horný limit nastaviteľnej hodnoty        |
| QLMN_LLM | Dolný limit nastaviteľnej hodnoty        |

## 1.5 WinCC

WinCC je softvér, ktorý slúži na vytvorenie vizualizačného programu, ktorý umožňuje znázornený proces kontrolovať, ale aj doňho počas jeho fungovania zasahovať.

Najvýznamnejšie editory WinCC využívané v praxi:

1. WinCC Explorer – predstavuje centrálny koordinačný nástroj projektu. Spravuje a poskytuje prístup ku všetkým komponentom WinCC. Poskytuje prehľad o nahratom projekte a o súboroch, ktoré sú v projekte použité.
2. WinCC Graphics Designer – predstavuje grafický systém pre tvorbu užívateľského rozhrania. Využíva grafické objekty umiestnené v paletách nástrojov, ktorými sa vytvárajú procesné obrazovky.
3. Editor menu a toolbarov - špecificky vytvorené menu a toolbary je možné integrovať do užívateľského rozhrania runtime aplikácie. Tieto nové ovládacie prvky môžu byť pevné alebo voľne premiestniteľné ako štandardné Windows objekty.
4. WinCC Alarm Logging Editor - tento konfiguračný nástroj slúži na konfiguráciu alarmov a správ, pre zaznamenávanie a logovanie udalostí.
5. WinCC Tag Logging Editor- slúži na konfiguráciu archívneho systému, ktorý umožňuje archiváciu procesných meraní, ich kompresiu a správu jednotlivých archívnych databáz.
6. WinCC Report Designer - slúži na konfiguráciu dokumentov a ich prepojenie s runtime dátami. Reporty je možné neskôr vytvárať na základe šablón v runtime aplikácii [7].

WinCC je prepojený s riadiacim systémom Step 7 prostredníctvom tagov, z ktorých čerpá dáta. Tag sa dá predstaviť ako spojovník medzi programom v PLC a samotným vizualizačným programom. Na jednej strane je tag napojený na pamäťovú adresu premennej, na druhej strane sa dáta sprístupňujú užívateľovi.

Rozlišujú sa 2 typy tagov: interný a externý.

Interný tag slúži na uloženie vnútorných premenných

Externý tag slúži na komunikáciu s PLC

## 2 Ciel' práce

Cieľom bakalárskej práce bolo vytvorenie základných algoritmov riadenia. Tieto algoritmy sme aplikovali na riadenie teploty a prietoku vzduchu laboratórneho výmenníka tepla LTR 700. Po úspešnom vytvorení algoritmov bolo ďalšou úlohou tieto algoritmy vizualizovať a vložiť do vzdialeného systému. Konečným výsledkom bakalárskej práce boli identifikované prenosy pre teplotu a prietok vzduchu laboratórneho výmenníka tepla LTR700.

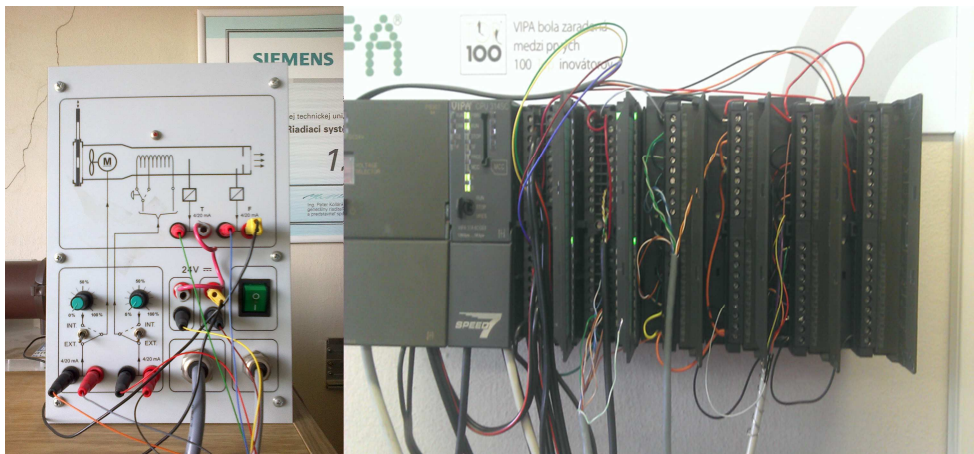
# 3 Metodika práce a metody skúmania

## 3.1 Zapojenia a moduly

Na obr.4 sme znázornili zapojenie PLC s laboratórnym výmenníkom tepla LTR 700 prostredníctvom zapojených káblov. Z hľadiska káblov snímače teploty sme prepojili pomocou zeleného káblu a snímače prietoku pomocou modrého káblu. S PLC sme tieto káble prepojili pomocou analógového modulu, ktorý bol súčasťou CPU pre výstupné veličiny. Snímač ohrevu špirály sme prepojili pomocou červeného kábla. Snímač otáčok ventilátora sme prepojili pomocou oranžového kábla. S PLC sme tieto káble prepojili pomocou analógového modulu pre vstupné veličiny.

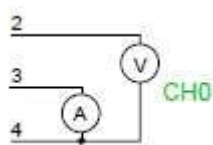
Pomocou MPI portu sme zabezpečili prepojenie PLC s počítačom. Prostredníctvom prepojenia cez MPI port sme do PLC vložili vytvorený program na riadenie teploty a prietoku vzduchu.

Pomocou priemyselného routra eWON sme zabezpečili komunikáciu PLC so vzdialeným systémom cez vytvorenú webovú stránku.



**Obr. 4 Zapojenia a moduly**

Každý modul je rozdelený do skupín, označených písmenom CH. Ich označenie bolo CH0-CH3. Na každú skupinu bolo možné pripojiť len jeden vstupný alebo výstupný signál. Spôsob zapojenia na modul sme zobrazili na obr.5. Na druhý pin sme mali možnosť pripojiť kábel pre napäťový signál. Na tretí pin sme pripojili kábel pre prúdový signál. Štvrtý pin nám slúžil na uzemnenie.



Obr. 5 Zapojenie

### 3.1.1 Vyhodnocovanie a generovanie signálov

Rozsah prúdového unifikovaného signálu bol v rozsahu 0-20mA a 4-20 mA. Dôvod prečo sme zaviedli oba rozsahy spočíval v tom, že nám hodnoty teploty ukazovali záporné hodnoty. Aby sme sa tomu vyhli zaviedli sme pre teplotu unifikovaný signál 0-20 mA.

Vstupy I/O modulov sme využili na ovládanie akčných členov ohrevu špirály a otáčky ventilátora. Výstupy I/O modulov sme využili na ovládanie snímačov teploty a prietoku vzduchu.

## 3.2 Konfigurácia

Prvým krokom pri vytvorení programu bolo zabezpečiť konfiguráciu hardvéru. Úlohami bolo vložiť konkrétne CPU, ktoré sme využívali pri tvorbe programu, zapojené vstupné a výstupné moduly pripojené k CPU. Na začiatku sme museli otvoriť program HW konfigurácie. HW konfiguráciu sme otvorili v programe Simatic Manager. V novovytvorenom projekte v rade SIMATIC 300 Station sme do kolónky hardvérovej konfigurácie vložili kód compatibility, ktorý bol uvedený v manuáli *CPU 314C-2DP (6ES7 314-6CG03-0AB0 V2.62)*. Keďže bolo nutné použiť ethernetový kábel, bolo potrebné vložiť modul CP343\_1 –Simatic Net. Moduly sme vkladali zo zoznamu dostupných modulov poskytnutých programom. Z toho zoznamu sme vybrali moduly, ktoré sme využili na prepojenie výmenníka tepla s PLC

HW konfiguráciu sme zobrazili na obr.6. Zobrazenie HW konfigurácie sme rozdelili na dve časti. Horná časť obr.6 zobrazila zariadenia použité pri vytvorení programu a ich prepojenie. Dolná časť obr.6 zobrazovala zloženie zariadenia využíteho pri tvorbe programu. Druhé miesto obsadilo využitie CPU, tretie miesto sme neobsadili, z toho dôvodu, že na PLC sa môže pripojiť iné zariadenie. Štvrtý až siedmi riadok predstavoval využitie vstupné a výstupné moduly. Ôsmy riadok obsadil modul pre vzdialený systém.

|        |                      |
|--------|----------------------|
| (0) UR |                      |
| 1      |                      |
| 2      | <b>CPU314 C-2 DP</b> |
| X2     | DP                   |
| 2.2    | DI24/DO16            |
| 2.3    | AI5/AO2              |
| 2.4    | Count                |
| 2.5    | Position             |
| 3      |                      |
| 4      | AI4/AO2x8/8Bit       |
| 5      | AI4/AO2x8/8Bit       |
| 6      | AI4/AO2x8/8Bit       |
| 7      | AO4x12Bit            |
| 8      | CP 343-1             |
| 9      |                      |
| 10     |                      |
| 11     |                      |

| (0) UR |                      |                            |             |          |           |           |  |
|--------|----------------------|----------------------------|-------------|----------|-----------|-----------|--|
| Slot   | Module               | Order number               | Fl...       | M...     | I address | Q address |  |
| 1      |                      |                            |             |          |           |           |  |
| 2      | <b>CPU314 C-2 DP</b> | <b>6ES7 314-6CG03-0AB0</b> | <b>V2.6</b> | <b>2</b> |           |           |  |
| X2     | DP                   |                            |             |          | 1023*     |           |  |
| 2.2    | DI24/DO16            |                            |             |          | 124...126 | 124...126 |  |
| 2.3    | AI5/AO2              |                            |             |          | 752...767 | 752...755 |  |
| 2.4    | Count                |                            |             |          | 768...783 | 768...783 |  |
| 2.5    | Position             |                            |             |          | 784...799 | 784...799 |  |
| 3      |                      |                            |             |          |           |           |  |
| 4      | AI4/AO2x8/8Bit       | 6ES7 334-0CE01-0AA0        |             |          | 272...279 | 272...275 |  |
| 5      | AI4/AO2x8/8Bit       | 6ES7 334-0CE01-0AA0        |             |          | 280...287 | 276...279 |  |
| 6      | AI4/AO2x8/8Bit       | 6ES7 334-0CE01-0AA0        |             |          | 288...295 | 288...291 |  |
| 7      | AO4x12Bit            | 6ES7 332-5HD01-0AB0        |             |          |           | 304...311 |  |
| 8      | CP 343-1             | 6GK7 343-1EX11-0XE0        | V2.0        | 3        | 256...271 | 256...271 |  |
| 9      |                      |                            |             |          |           |           |  |
| 10     |                      |                            |             |          |           |           |  |
| 11     |                      |                            |             |          |           |           |  |

Obr. 6 HW konfigurácia

### 3.3 Identifikácia procesu podľa Strejcovej metódy

Strejcova metóda identifikácie procesu sa využíva pri vyhodnotení odmeranej odozvy systému na skokovú zmenu vstupnej veličiny.

Dynamicke vlastnosti identifikovaného systému aproximujeme pomocou náhradného prenosu v tvare:

$$G(s) = \frac{z}{(T_s + 1)^n} e^{-Ds} \quad (5)$$

Postup identifikácie:

1. Z prechodovej charakteristiky určiť hodnoty zosilnenia, čas nábehu  $t_n$ , čas priet'ahu  $t_u$
2. Určiť podiel

$$f(s) = \frac{tu}{tn} \quad (6)$$

3. Z tab.3 vybrať rád systému tak, aby platila rovnica

$$f(n) \leq fs < f(n + 1) \quad (7)$$

4. Určiť dopravné oneskorenie ako rozdiel medzi skutočným a fiktívnym časom nábehu  $t_u$

$$D = [fs - f(n)]t_n \quad (8)$$

5. Určiť časovú konštantu pomocou hodnôt z riadku funkcie v tab.3 .pre príslušný rád.

$$T = g(n)t_n \quad [6] \quad (9)$$

**Tab.3 Tabuľka pre strejcovu metódu identifikácie**

| N    | 1     | 2     | 3     | 4     | 5     | 6     |
|------|-------|-------|-------|-------|-------|-------|
| f(n) | 0,000 | 0,104 | 0,218 | 0,319 | 0,410 | 0,493 |
| g(n) | 1,000 | 0,368 | 0,271 | 0,224 | 0,195 | 0,161 |

Prameň: Bakošová, M.,- Fikar, M. 2008: Riadenie procesov, Bratislava, Slovenská technická univerzita, 2008, ISBN 978-80-227-3763-0

# 4 Výsledky práce

## 4.1 Základné algoritmy riadenia

Úlohou bakalárskej práce bolo zabezpečiť reguláciu nameraných výstupných veličín teploty a prietoku vzduchu. Na úspešné riešenie tejto úlohy sme zostrojili základné algoritmy riadenia. Zaoberali sme sa algoritmami ako manuálny, inverzný, PID, relé a nakoniec kombinačný algoritmus pre riadenie teploty a prietoku vzduchu. Hlavný program sme zapisovali v OB1,. Hlavný dátový blok sme označili DB3. Dôvodom bolo, že na tom istom PLC pracovalo viacero osôb na inej úlohe. Dátové bloky DB1,DB8,DB2 sme využívali pre PID regulátory.

Pre každý typ algoritmu riadenia sme v dátovom bloku vytvorili premenné v tvare BOOL, ktoré označovali konkrétny typ algoritmu. V prípade, ak sme nastavili hodnotu 1 režim bol aktívny, ak hodnotu 0 režim bol neaktívny. Pomocou SET a RESET výstupov sme zabezpečili, aby v tom istom okamihu nebolo zapnutých viac režimov. Pri vytváraní algoritmov sme využívali vnútorné premenné do ktorých sme priebežne zapisovali údaje. Vytvorené algoritmy sme vložili do prílohy B.

### 4.1.1 Algoritmus pre manuálny režim riadenia teploty a prietoku vzduchu

V manuálnom režime sme zadávali ľubovoľné hodnoty akčného zásahu, ktoré pomocou vytvoreného programu určilo ustálenú hodnotu výstupnej veličiny. Získané hodnoty sme spracovali s programom Matlab a získali sme graf pre ustálené hodnoty systému. Prostredníctvom programu pre manuálny režim sme určili maximálne a minimálne hodnoty vstupných a výstupných veličín.

### 4.1.2 Algoritmus pre inverzný režim riadenia prietoku vzduchu

Hlavným dôvodom vytvorenia algoritmu pre inverzný režim bolo získať ustálené hodnoty vstupnej veličiny pri zadaní intervalu hodnôt výstupnej veličiny. Pri riešení algoritmu inverzného

režimu sme vychádzali z manuálneho režimu riadenia prietoku vzduchu. V manuálnom režime sme vytvorili graf ustálených hodnôt funkcie:

$$y = f(x) \quad (10)$$

x- otáčky ventilátora

y- prietok vzduchu

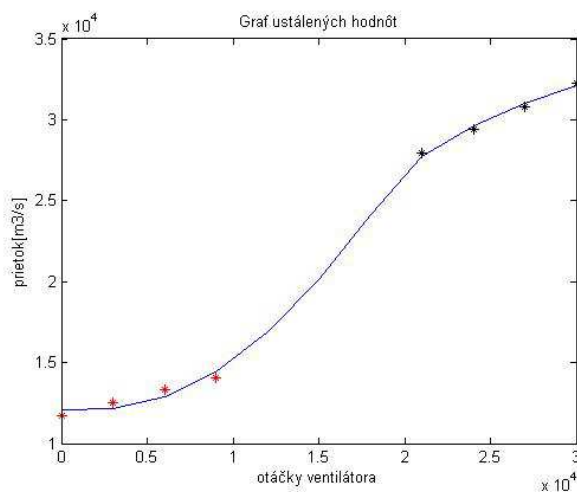
Postup zostrojenia spočíval v tom, že sme si zostrojenú krivku rozdelili do troch základných častí, z ktorých sme vytvorili ich inverznú funkciu. Dôvodom tohto postupu, bolo získať čo najmenšiu chybu, ktorá by nám ovplyvnila priebeh algoritmu. Získané rovnice priamky sme zlinearizovali a ich údaje sme vložili do riadiaceho algoritmu.

Na obr. 7 sme zobrazili graf ustálených hodnôt z algoritmu manuálneho režimu. Rovnicu (11) sme získali inverziou prvej časti krivky zobrazenej pomocou červených bodiek na obr.7. Rovnicu (12) sme získali inverznou funkciou strednej časti krivky na obr.7. Rovnicu (13) sme získali inverznou funkciou pre poslednú časť znázornenú čiernymi bodkami na obr.7

$$y = 3,8422x - 46488 \quad (11)$$

$$y = 0,90188x - 4024,5 \quad (12)$$

$$y = 2,0772x - 36636 \quad (13)$$



**Obr. 7 Zobrazenie ustálených hodnôt**

### 4.1.3 Algoritmus pre relé režim riadenia teploty a prietoku vzduchu

Relé režim pracoval na princípe dvojpohového regulátora. Matematicky ho bolo možno vyjadriť nasledujúcou rovnicou:

$$u = \begin{cases} u_{max} & \text{ak platí } e > 0 \\ u_{min} & \text{ak platí } e < 0 \end{cases} \quad (14)$$

Zo vzťahu bolo zrejmé, že regulátor v prípade kladnej relatívnej odchýlky generoval maximálne možný akčný zásah. V prípade zápornej relatívnej odchýlky generoval minimálne možný akčný zásah. Z dôvodu odstránenia nežiaducej skokovej zmeny sme boli nútení zaviesť tzv. hysteréziu. Hysterézia predstavuje rozhranie medzi tým kedy výstup mení svoju hodnotu z maximálnej hodnoty akčnej veličiny na hodnotu 0, a keď prechádza výstupná veličina z hodnoty 0 na hodnotu maximálnej hodnoty akčného zásahu.

### 4.1.4 Algoritmus pre PID režim riadenia

V práci sme sa zaberali troma PID regulátormi. PID regulátor na reguláciu teploty, prietoku a kombináciu týchto dvoch veličín pri konštantnej teplote. Dôležité je spomenúť, že úlohou bakalárskej práce bolo navrhnuť algoritmus pre PID regulátor. Keďže matematický model pre výmenník tepla nebol v zadaní práce, neboli následne určené hodnoty zosilnenia, časovej a derivačnej konštanty. Získanie týchto hodnôt pre regulátor bolo podmienkou pre ustálenie systému. V STEP7 sme vytvorili blok, kde sme výpočtom podľa vzťahov určili hodnoty konštant pre vstupné a výstupné veličiny. Tieto hodnoty sme do programu vložili tak aby PID regulátor fungoval.

Vzorový výpočet hodnôt vstupných veličín regulátora:

Akčný zásah ventilátora bol v hodnote 0 až 30000. V rozsahu od 0 po 100.

$$LMN = \frac{LMN_{PER} * 100}{27648} \quad (15)$$

$$LMN_{MIN} = \frac{0 * 100}{27648} = 0 \quad (16)$$

$$LMN_{MAX} = \frac{30000 * 100}{27648} = 108,5069 \quad (17)$$

Riešenie sústavy rovníc: cez rozsah akčného zásahu

$$100a + b = 108,5069 \quad (18)$$

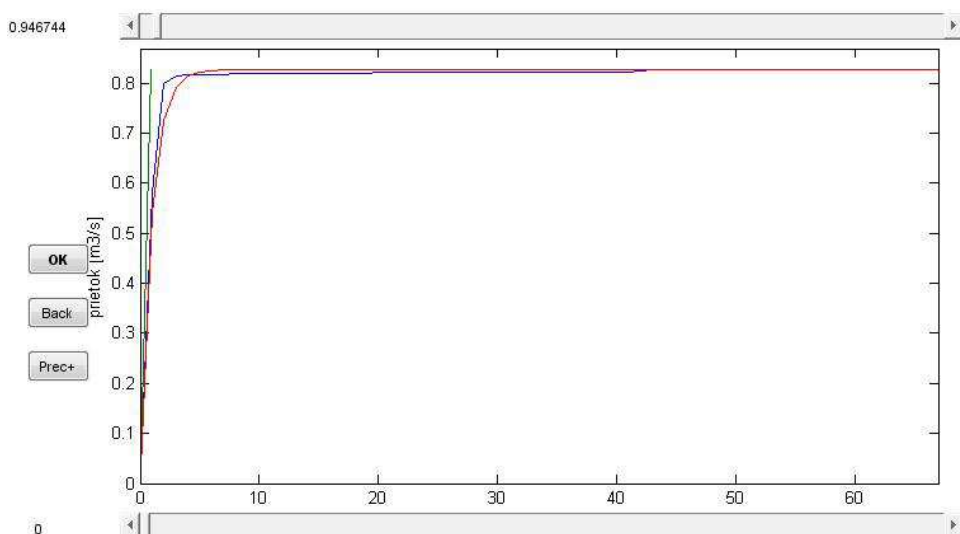
$$0a + b = 0 \quad (19)$$

$$LMN_{FAC} = a = 1,085069 \quad (20)$$

$$LMN_{OFF} = b = 0 \quad (21)$$

## 4.2 Identifikácia procesu

Proces sme identifikovali pomocou údajov, ktoré sme získali prostredníctvom vzdialeného systému pre manuálne riadenie. Vytvorili sme prechodové funkcie, ktoré predstavovali matematický opis systému pri jednotkovej skokovej zmene vstupnej veličiny pri nulových začiatkových podmienkach. Proces sme identifikovali Strejcovou metódou pomocou programu Piddesgn. Postup identifikácie pomocou Strejcovej metódy sme opísali v kapitole 3.3.

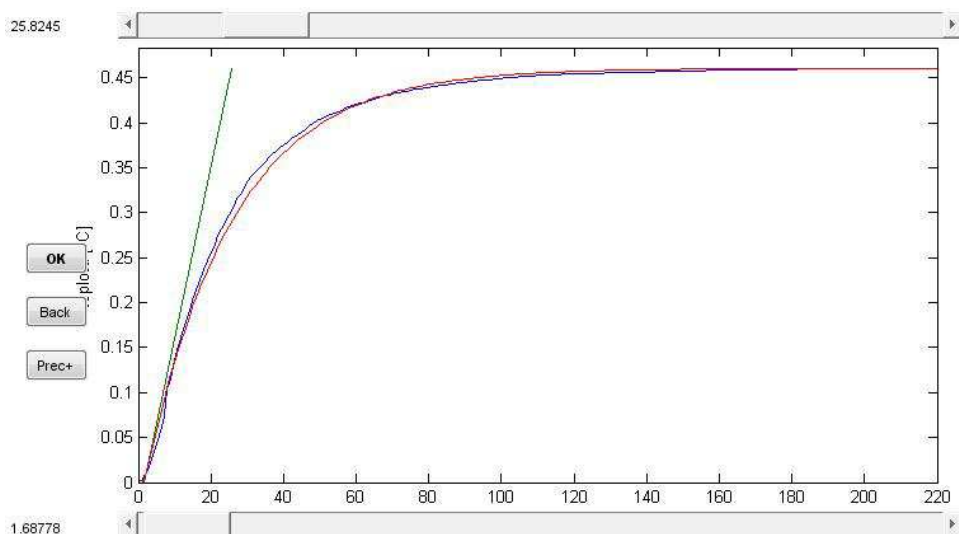


**Obr. 8 Identifikácie prietok vzduchu**

Matematické vyjadrenie prenosu :

$$G(s) = \frac{0,82702}{0,946747s+1} e^{-0s} \quad (22)$$

Chyba identifikácie mala hodnotu 0,00019487



**Obr. 9** Identifikácia teplota vzduchu

Matematické vyjadrenie prenosu:

$$G(s) = \frac{0,46067}{24,1367s+1} e^{-1,6878s} \quad (23)$$

Chyba identifikácie mala hodnotu 0,00016593

## 4.3 Vizualizácia procesu

### 4.3.1 Postup zostrojenia vizualizačných obrazoviek

Prvým krokom k dosiahnutiu úspešnej vizualizácie bolo vloženie základných adries premenných do *TAG MANAGMENT*. Do *TAG MANAGMENT* sme vložili základné algoritmy režimov riadenia, ktoré sme chceli na obrazovke vizualizovať. Vytvorili sme samotnú skupinu tagov pre PID regulátory.

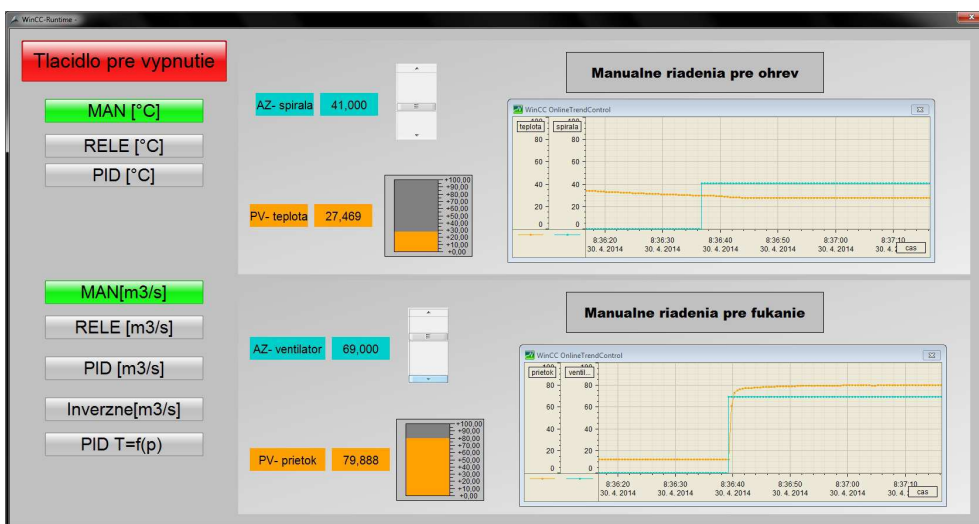
Ako druhý krok k úspešnej vizualizácie bolo zostrojenie hlavnej obrazovky. Nastavili sme jedno tlačidlo, ktoré označovalo či proces bol vypnutý alebo zapnutý. Na ľavej strane sme vytvorili tlačidlá, ktoré nám poukazovali na konkrétne algoritmy režimov riadenia. V prípade ak režimy svietili na zeleno, boli aktívne. Pravú časť obrazovky reprezentovali zostrojené grafy,

ktoré sme naviazali na algoritmy režimov riadenia. V programe sme zabezpečili prepočet hodnôt teploty a prietoku na reálne hodnoty. Pre teplotu na °C a pre prietok na m<sup>3</sup>/s.

## 4.3.2 Postup zobrazenia vizualizačných obrazoviek pre algoritmy riadenia

### 4.3.2.1 Manuálny algoritmus pre riadenie teploty a prietoku

Prostredníctvom tlačidiel MAN sme nastavili systém na prepnutie do manuálneho režimu pre riadenie teploty a prietoku vzduchu. Akčný zásah sme naviazali na slider. Sliderom sme ľubovoľne mohli meniť hodnoty akčných zásahov. Výstupné veličiny sme naviazali na bar, ktorým sme mali možnosť zistiť hodnotu výstupu pri aktuálnej hodnote akčného zásahu. Údaje sme zobrazili v grafe na obr. 10. Modrá krivka poukazovala na akčný zásah, oranžová krivka na výstupné veličiny.



Obr. 10 Manuálny algoritmus

### 4.3.2.2 Relé algoritmus pre riadenie teploty a prietoku vzduchu

Na obr.11. sme zobrazili relé režim pre riadenie teploty a prietoku. Funkčnosť režimu sme zabezpečili stlačením tlačidla RELE. Adresy hysterézie a set pointu sme naviazali pomocou tagov na slidery. Výstupné veličiny sme naviazali na bar. Výsledok sme znázornili na grafoch v

obr.11. V oboch prípadoch modrá krivka reprezentovala akčné zásahy. Žltá krivka poukazovala na hodnoty set pointu a oranžová krivka poukazovala na hodnoty výstupných veličín.



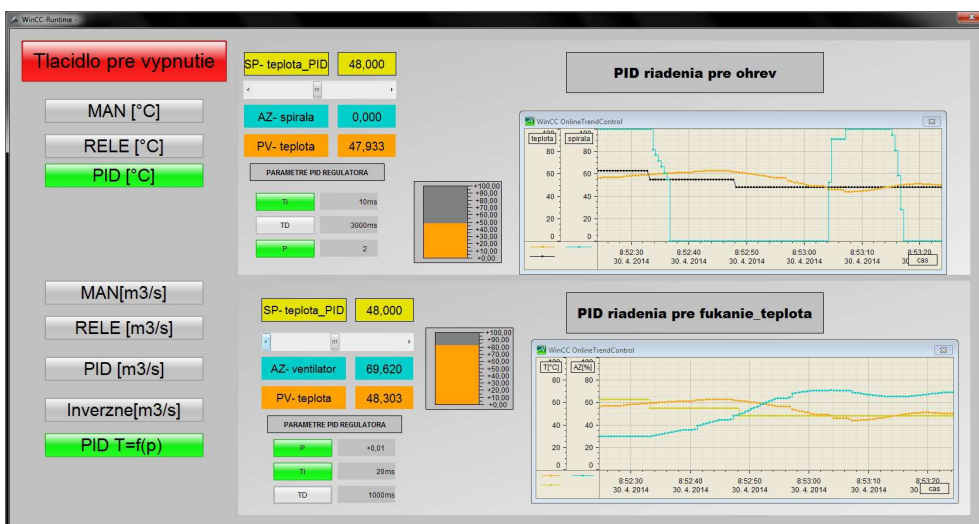
Obr. 11 Relé algoritmus

#### 4.3.2.3 PID algoritmus pre riadenia teploty a prietoku vzduchu

Na obr.12 a obr.13 sme zobrazili PID regulátor na reguláciu teploty a prietoku vzduchu . Na slider sme naviazali adresu set pointu, ktorého hodnotu sme ľubovoľne menili. Pre zložky PID regulátora sme vytvorili spínače, ktoré v prípade zapnutia svietili nazeleno. Vedľa spínačov sme vytvorili kolónky, ktoré sme naviazali na adresy zložiek PID regulátora. Kolónkami sme mohli meniť hodnoty zložiek . Na bar sme naviazali adresu výstupných veličín. Na obr. 10 znázorňujeme PID regulátor na reguláciu prietoku pri konštantnej teplote. Tento PID regulátor porovnávame s regulátorom pre reguláciu teploty. V oboch prípadoch sme vytvorili slider , ktorým môžeme meniť hodnotu set pointu. Táto hodnota bola pre oba procesy naviazaná na seba a navzájom majú neustále rovnakú hodnotu.



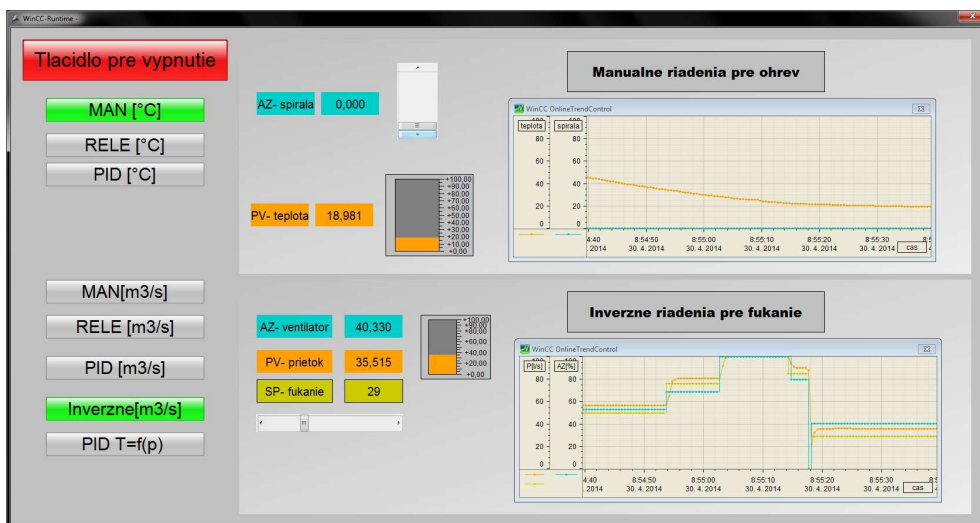
Obr. 12 Algoritmus PID



Obr. 13 Algoritmus PID

#### 4.3.2.4 Inverzný algoritmus pre riadenie prietoku

Na obr. 12 sme zobrazili vizualizáciu pre algoritmus ustálených hodnôt vstupných veličín. Akčný zásah sme naviazali prostredníctvom tagu na slider. Výstupnú veličinu sme naviazali na bar, ktorý nám v stĺpci ukazoval ustálenú hodnotu. Grafické znázornenie sme vytvorili pomocou trendu.



Obrázok 14 Inverzný algoritmus

## 4.4 Vzdialený prístup riadenia eWON

### 4.4.1 Charakteristika Modemu eWON

Modemy eWON sú určené k vzdialenej správe PLC a ďalších zariadení, ktoré sú pripojené prostredníctvom sériového alebo ethernetového portu. Pomocou modemu je možné vzdialene programovať PLC, konfigurovať operátorský panel a pomocou kamery sledovať proces. Modem je schopný archivovať dáta z PLC do vstavanej pamäte a vizualizovať dáta z PLC v internetovom prehliadači počítača [8].

### 4.4.2 Vkládanie premenných z PLC do eWONu

Premenné, ktoré sme vkladali do eWONu sme stiahli z dátových blokov DB1, DB2, DB3 a DB8. DB2 predstavoval dátový blok pre PID regulátor na riadenie prietoku. DB8 na riadenie prietoku pri konštantnej teplote. DB1 dátový blok PID regulátora pre riadenie teploty.

Postup vkladania spočíval v tom, že sme si na internetovom prehliadači nastavili stránku eWONu. Prešli sme do kolónky MAIN Menu a TAG setup- Create new tags. V Tag name sme si pomenovali premennú. Dôležité bolo pridať pred samotný názov he\_, pretože na eWON boli pripojené viaceré procesy, v dôsledku čoho eWON neprijal rovnaké názvy premenných. Nastavili sme, aby premenná bola naviazaná na laboratórny výmenník tepla (fúkar). Na obr. 15 sme znázornili ako sme vkladali premennú prietoku vzduchu. Server name predstavuje typ PLC

v našom prípade to bolo S73&400, ktorý predstavuje Siemens 300s alebo 400s. Adresa premennej DB3F38, kde DB predstavovala z akého dátového bloku sme premennú stiahli. F typ premennej, 38 na akej adrese sa premenná nachádzala. V tab.4 sme znázornili všetky premenné, ktoré sme vložili do eWONU. Prvý stĺpec predstavoval názvy premenných, ktoré sme využívali vo vzdialenom systéme. Druhý stĺpec predstavoval premenné. Tretí stĺpec predstavoval adresu premenných využitých V PLC. Štvrtý stĺpec predstavoval adresy premenných využitých v eWONE. Piaty stĺpec znázorňoval aký typ premenných sme využívali v PLC aj v eWONE. Šiesty stĺpec poukazoval na typ využitého PLC.

Obr.15 Znáznornenie vkladanie premennej do EWONU

Tab. 3 Označenie všetkých premenných vložených do EWONU

| Premenná v ewone    | Premenná      | Adresa v PLC | Adresa v ewone | Typ            | Server name |
|---------------------|---------------|--------------|----------------|----------------|-------------|
| he_fan              | Ventilator.   | DB3.DBD36    | DB3F36         | Floating-point | S73&400     |
| he_heating          | Spirala       | DB3.DBD40    | DB3F40         | Floating point | S73&400     |
| He_MAN_fukanie      | Man- fukanie  | DB3.DBX0.6   | DB3B0#6        | Bool           | S73&400     |
| He_MAN_ohrev        | Man-ohrev     | DB3.DBX0.7   | DB3B0#7        | Bool           | S73&400     |
| he_air_temp         | Teplota       | DB3.DBD32    | DB3F32         | Floating-point | S73&400     |
| he_air_flow         | Prietok       | DB3.DBD38    | DB3F38         | Floating-point | S73&400     |
| he_SP- ohrev        | SP- ohrev     | DB3.DBW10    | DB3S10         | Integer        | S73&400     |
| he_SP-fukanie       | SP- fukanie   | DB3.DBW8     | DB3S8          | Integer        | S73&400     |
| he_Rele-fukanie     | Rele- fukanie | DB3.DBX6.4   | DB3B6#4        | Bool           | S73&400     |
| he_Rele ohrev       | Rele ohrev    | DB3.DBX6.5   | DB3B6#5        | Bool           | S73&400     |
| he_H-fukanie        | H- fukanie    | DB3.DBW12    | DB3S12         | Integer        | S73&400     |
| he_H-ohrev          | H-ohrev       | DB3.DBW14    | DB3S14         | Integer        | S73&400     |
| he_Inv-fukanie      | Inv- fukanie  | DB3.DBX6.0   | DB3B6#0        | Bool           | S73&400     |
| he_PID_ohrev        | PID- ohrev    | DB3.DBX6.3   | DB3B6#3        | Bool           | S73&400     |
| he_PID-fukanie      | PID- fukanie  | DB3.DBX6.2   | DB3B6#2        | Bool           | S73&400     |
| he_PV- PID          | PV- PID       | DB3.DBD24    | DB3F24         | Floating point | S73&400     |
| he_PID_T=f(prietok) | PID- p=f(T)   | DB3.DBX58.0  | DB3B58#0       | Bool           | S73&400     |

|                         |                  |            |         |                |         |
|-------------------------|------------------|------------|---------|----------------|---------|
| <b>he_P_ohrev</b>       | Gain- Spirala    | DB1.DBD20  | DB1F20  | Floating point | S73&400 |
| <b>he_TI_1</b>          | TI               | DB1.DBS24  | DB1S24  | integer        | S73&400 |
| <b>he_TD_1</b>          | TD               | DB1.DBS28  | DB1S28  | integer        | S73&400 |
| <b>He_P-sel_ohrev</b>   | P-sel            | DB1.DBX0.3 | DB1B0#3 | bool           | S73&400 |
| <b>he_I_SEL_ohrev</b>   | I-sel            | DB1.DBX0.4 | DB1B0#4 | bool           | S73&400 |
| <b>he_D_SEL_fukanie</b> | D-sel            | DB1.DBX0.7 | DB1B0#7 | bool           | S73&400 |
| <b>he_P_fukanie</b>     | Gain- ventilator | DB2.DBD20  | DB2F20  | Floating point | S73&400 |
| <b>he_TI_fukanie</b>    | TI               | DB2.DBS24  | DB2S24  | integer        | S73&400 |
| <b>he_TD_2</b>          | TD               | DB2.DBS28  | DB2S28  | integer        | S73&400 |
| <b>he_P-sel_fukanie</b> | P-sel            | DB2.DBX0.3 | DB2B0#3 | bool           | S73&400 |
| <b>he_I_SEL_fukanie</b> | I-sel            | DB2.DBX0.4 | DB2B0#4 | bool           | S73&400 |
| <b>he_D_SEL_fukanie</b> | D-sel            | DB2.DBX0.7 | DB2B0#7 | bool           | S73&400 |
| <b>he_P_f(p)</b>        | Gain- kombin     | DB8.DBD20  | DB8F20  | Floating point | S73&400 |
| <b>he_TI_2</b>          | TI               | DB8.DBS24  | DB8S24  | integer        | S73&400 |
| <b>he_TD_3</b>          | TD               | DB8.DBS28  | DB8S28  | integer        | S73&400 |
| <b>he_P_SEL_f(P)</b>    | P-sel            | DB8.DBX0.3 | DB8B0#3 | bool           | S73&400 |
| <b>he_I-sel_f(P)</b>    | I-sel            | DB8.DBX0.4 | DB8B0#4 | bool           | S73&400 |
| <b>he_D_SEL_f(P)</b>    | D-sel            | DB8.DBX0.7 | DB8B0#7 | bool           | S73&400 |

Hlavný význam vzdialeného systému bolo sledovanie a kontrolovanie procesu mimo laboratórium, v ktorom je fyzicky prístroj pripojený na počítač a na PLC. Prostredníctvom vložených premenných existuje možnosť vytvoriť ďalšie algoritmy na riadenie teploty a prietoku vzduchu laboratórneho výmenníka tepla. Pomocou vzdialeného systému bolo možné stiahnuť údaje na identifikáciu procesu. Vo vzdialenom systéme sme určili časový interval v ktorom má systém údaje spracovať a vytvoriť súbor s týmito hodnotami.

## 5 Diskusia

V bakalárskej práci sme sa zaoberali problematikou najpoužívateľnejšej automatizačnej techniky v praxi -PLC. Zaoberali sme PLC 300s od spoločnosti VIPA, ktorú nám poskytla škola na riešenie problematiky riadenia procesov. Prvý dôvod prečo sme uprednostnili PLC 300s je že PLC 200s poskytnutý školou mohol mať maximálne dva pripojené moduly. Na zabezpečenie riadenia laboratórneho výmenníka tepla sme potrebovali dva dostupné I/O moduly a na vzdialenie riadenie sme potrebovali komunikačný modul. Ďalší dôvod prečo sme si nemohli zvoliť PLC 200 s bolo, že mal pripojený len jeden komunikačný modul na ktorý už bol pripojený proces elektrického pohonu. Z týchto dôvodov sme uprednostnili PLC 300s od spoločnosti VIPA pred PLC 200s od spoločnosti Siemens.

V práci sme sa zaoberali štyrmi úlohami. Úlohy spočívali v navrhnutí algoritmov na riadenie laboratórneho výmenníka tepla LTR 700 v softvéry SIMATIC STEP 7 od spoločnosti Siemens. Pomocou navrhnutých algoritmov riadenia sme vytvorili ich vizuálny pohľad na problematiku riadenia pomocou softvéru WinCC od spoločnosti Siemens. Prostredníctvom týchto dvoch čiastkových cieľov sme vytvorené premenné vložili do vzdialeného systému. Pomocou ktorého sme získali identifikované prenosy.

Z práce je zrejmé, že najjednoduchší algoritmus, ktorý sme vytvorili predstavoval manuálny algoritmus. Algoritmus zabezpečoval ustálenosť systému pri zadaní intervalu hodnôt akčných zásahov. Z vytvorených grafov prostredníctvom vizualizácie bolo zrejmé, že systém sme dokázali uradiť. Z tohto dôvodu sme mohli konštatovať, že systém bol ustálený.

Relé algoritmus bol zložitejší v tom zmysle, že sme museli pridať hysteréziu, ktorá nám odstránila nežiaducu skokovú zmenu. V prípade riadia prietoku výmenníka tepla boli časté skokové zmeny vstupnej veličiny, ktoré mali príčinu, že sa systém nedokázal priblížiť k žiadanej hodnote výstupnej veličiny. V prípade riadenia teploty neboli skokové zmeny akčnej veličiny časté, čo zapríčinilo, že výstupná veličina sa približovala žiadanej hodnote výstupnej veličiny. Z týchto dôvodov neexistuje možnosť ustálenia systému.

Pri vytvorení PID algoritmu sme proces nedokázali ustáliť. Hlavným dôvodom bolo, že sme nevytvorili matematický model. Zložky PID regulátora sme mohli získať prostredníctvom získaného identifikovaného procesu a vytvoreného matematického modelu. Vybranou vhodnou metódou by sme mohli regulovať PID regulátor laboratórneho výmenníka tak aby bol ustálený, mal čo najmenšiu relatívnu odchýlku, nekmitavosť systému. Z dôvodu nezískania hodnôt PID regulátora sme nedokázali zabezpečiť ustálenosť systému.

Dôležitou úlohou bakalárskej práce bolo vytvorenie vzdialeného systému prostredníctvom eWONu. Našou úlohou nebolo vytvorenie webovej stránky ale len vloženie adres využitých pri riadení výstupných veličín procesu. Vo vzdialenom systéme sme vytvorili len manuálny algoritmus riadenia pre výstupné veličiny. Pomocou vzdialeného systému študenti majú možnosť na stránke oddelenia automatizácie, informatizácie sledovať priebeh procesu laboratórneho výmenníka tepla.

Poslednou úlohou bakalárskej práce bolo identifikovať proces pre dve vstupné veličiny. Grafické znázornenia a matematické vyjadrenia procesu sme zobrazili v kapitole 4.2. Údaje prostredníctvom, ktorých sme identifikovali proces sme stiahli zo vzdialeného systému. Z identifikácie sme získali systémy 1. rádu, pričom sme museli využiť v programe piddesign funkciu controll tuning na zabezpečenie čo najnižšej chyby merania. Táto identifikácia by sa mohla dať využiť na návrh PID regulátora v prípade zostrojenia matematického modelu.

## 6 Záver

Cieľom bakalárskej práce bolo pochopiť základné poznatky využitia automatizačnej techniky v praxi. Vybrali sme si zo zariadení poskytnutých školou laboratórny výmenník tepla LTR 700.

V prvej časti bakalárskej práce sme sa zaoberali teoretickým základom najpoužívateľnejšej automatizačnej techniky - programovateľný logický automat. Zaoberali sme sa jeho vývojom , architektúrou, zložením a základným popisom jednotlivých častí programovateľného automatu.

V metodike práce sme sa zaoberali prepojením PLC s laboratórnym výmenníkom tepla a s počítačom. Zdefinovali sme metódu, ktorou sme identifikovali proces laboratórneho výmenníka tepla.

Vo výsledkoch práce sme popísali algoritmy vytvorené pre laboratórny výmenník tepla, graficky sme zobrazili algoritmy pomocou vizualizácie. Posledným krokom úspešného riešenia bakalárskej práce bolo vloženie premenných do vzdialeného systému . Pomocou vzdialeného systému pre manuálny režim sme identifikovali proces riadenia výmenníka tepla..

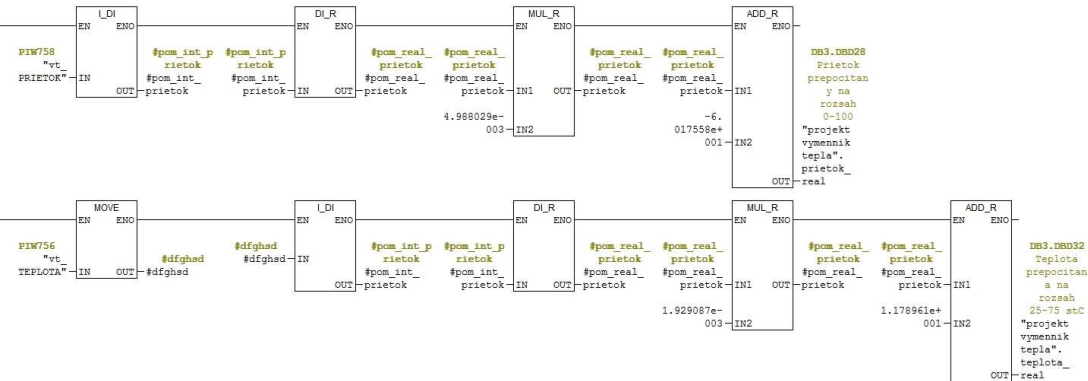
## 7 Zoznam použitej literatúry

1. Zezulka, F.- Bradáč, Z. – Fiedler, P. –Kučera, P.- Štohl, R.2003. Programovateľné automaty,[online].2003, [cit. 2015.04.20], s. 7-13 Dostupné na internete : <[http://www.vaeprosyst.cz/Dokumentace/Programovatelne\\_automaty/Programovatelne\\_automaty-Skripta\\_FEKT\\_VUT\\_Brno.pdf](http://www.vaeprosyst.cz/Dokumentace/Programovatelne_automaty/Programovatelne_automaty-Skripta_FEKT_VUT_Brno.pdf)>
2. Kovař J., Prokopova Z., Šmejkal L., Programování PLC,[online] Investice do rozvoje vzdělání, reg. č. CZ 1.07/1.1.08/01.0016, [cit. 2015.04.22], s. 2-5, Dostupné na internete: <[http://www.spszl.cz/soubory/plc/programovani\\_plc.pdf](http://www.spszl.cz/soubory/plc/programovani_plc.pdf)>
3. Mraško, L. 2010. PLC a ich programovanie-1. Čo je to PLC?, Elektrotechnika,2010, roc. 3, č 4,[cit. 2015.04.22], s. 1,
4. Vaneková, K. 2004. Priemyselný riadiaci systém SIMATIC, Bakalársky projekt , Bratislava, STU, Katedra informatizácie a riadenia procesov, 2004,[cit. 2015.04.23], s.9,
5. Control System, manual, System 300s, [online],[cit. 2015.04.24] Dostupné na internete: <<http://support.controlsystem.sk/vipa/system-300s>>
6. Bakošová, M.,- Fikar, M. 2008: Riadenie procesov, Bratislava, Slovenská technická univerzita, 2008. [cit 2015.04.24], ISBN 978-80-227-3763-0,,s.148-149, s. 117-120
7. SIEMENS, Simatic WinCC procesná vizualizácia [online], [cit. 2015.04.27] Dostupné na internete: <[www.cee.siemens.com/web/sk/sk/produkty\\_sluzby/industry/automatizacne\\_systemy/priemyselne\\_auto\\_sys/simatic\\_wincc](http://www.cee.siemens.com/web/sk/sk/produkty_sluzby/industry/automatizacne_systemy/priemyselne_auto_sys/simatic_wincc)>
8. Foxon s.r.o. 2010 Průmyslové modemy eWON [online],Automa, 2010,[cit. 2015.04.30], s. 22 Dostupné na internete: <<http://www.odbornecasopisy.cz/res/pdf/41658.pdf>>

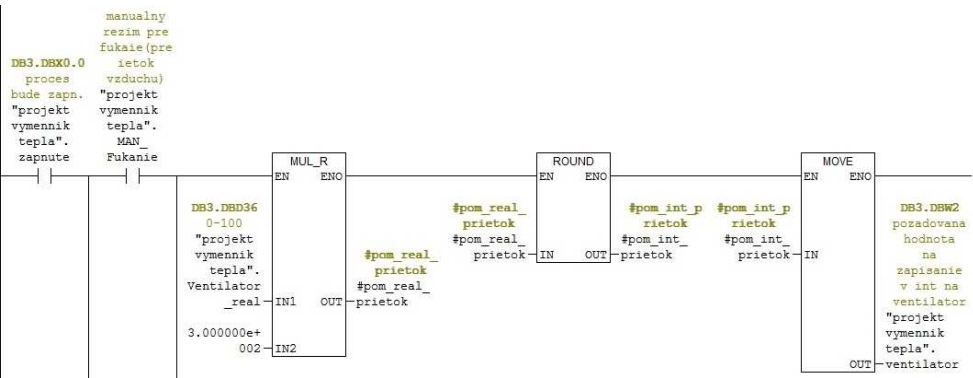
# 8 Príloha

Príloha A: Priložené CD s programom vytvorenom v Step 7

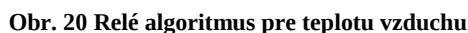
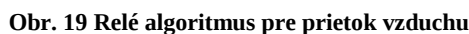
Príloha B. Algoritmy na riadenie teploty a prietoku vzduchu v Step 7

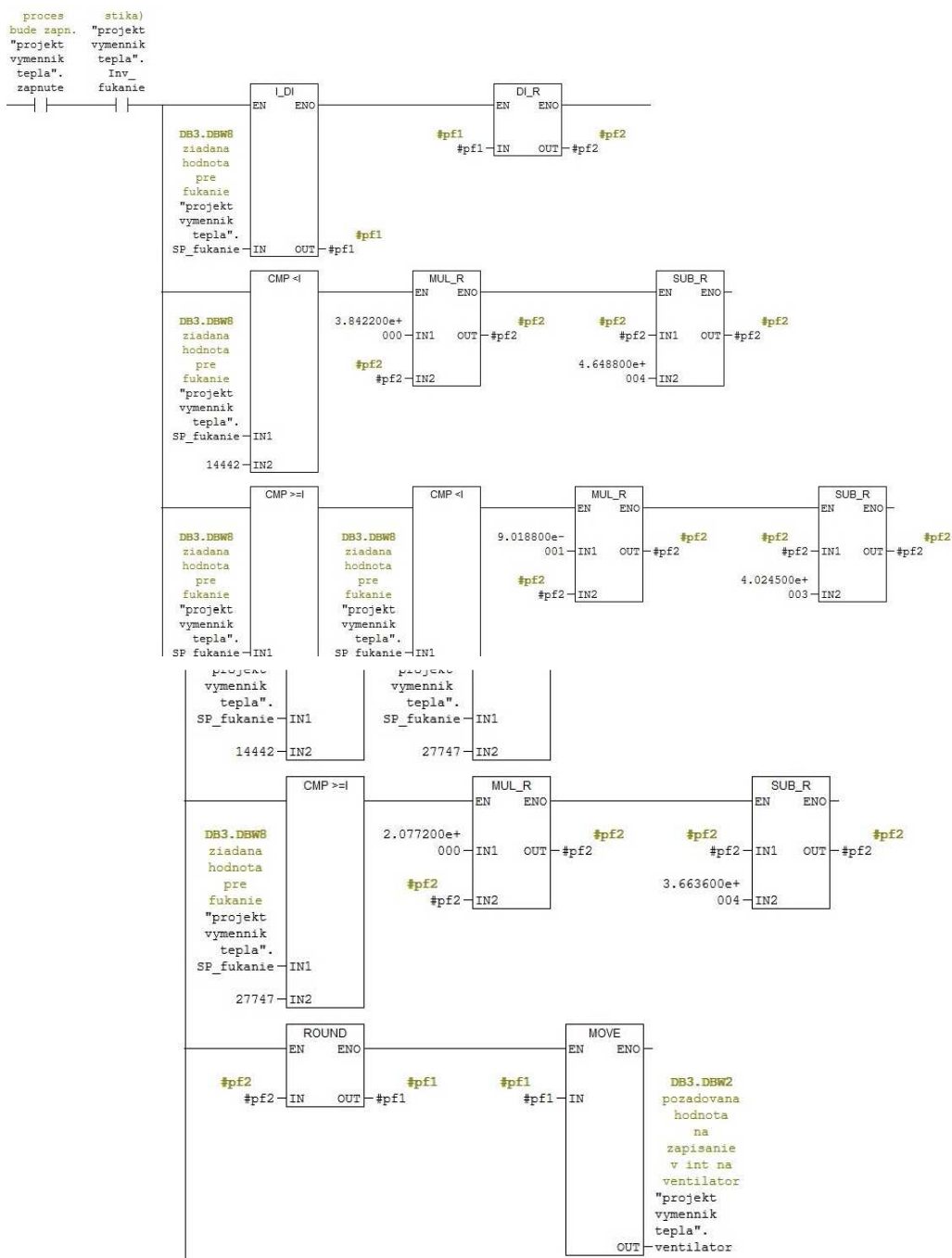


Obr.16 Prepočet hodnôt z integrov na real



Obr. 17 Manuálny algoritmus pre prietok vzduchu





Obr. 21 Inverný algoritmus pre prietok vzduchu

