

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE  
FAKULTA CHEMICKÉJ A POTRAVINÁRSKEJ  
TECHNOLÓGIE



KATEDRA INFORMATIZÁCIE A RIADENIA PROCESOV

INTELIGENTNÉ RIADENIE POMOCOUCI PLC SIMATIC S7  
S PODPOROU MATLABU

Dňa: 21.5.2004  
Miesto: Bratislava

Vedúci diplomovej práce: Prof. Ing. Ján Mikleš, DrSc  
Vypracoval: Bc. Juraj Vöröš

**Pod'akovanie:**

Týmto si dovoľujem poďakovať vedúcemu diplomovej práce Prof. Ing. Jánovi Miklešovi, DrSc za všestrannú pomoc, ochotu a odborné vedenie. Zároveň chcem poďakovať Ing. Lukášovi Dermíškovi za cenné rady pri zvládaní riadiaceho systému a Doc. Ing. Jánovi Dankovi, CSc pri elektroinštalácii.

## **Abstrakt**

Diplomová práca sa zaoberá návrhom prepojenia programov Matlab, WinCC a Microsoft Excel cez DDE rozhranie. Ide o rozhranie, ktoré vyvinula spoločnosť Microsoft na nadviazanie vzájomnej komunikácie medzi aplikáciami, na báze operačného systému MS Windows. Práca tiež obsahuje návrh riadenia teploty vzduchu pomocou LQ regulátora s integračnou činnosťou. Výpočet akčnej veličiny výkonu špirály pomocou tohto regulátora je uskutočnený v Matlabe a cez program Microsoft Excel a WinCC sa akčná veličina posiela do reálneho zariadenia. Reálne zariadenie, na ktoré je aplikované riadenie je laboratórny prietokový ohrievač vzduchu.

## **Abstract**

This diploma work deals with the design of connection of the programs Matlab, WinCC and Microsoft Excel by DDE interface. It's an interface, which was made by Microsoft company for making interactive communication among applications, on the base of operating system MS Windows. This work contains the control design the air temperature by LQ controller with integral action. Calculation of operating value, heating power, by this controller is made in Matlab and through Microsoft Excel and WinCC is sending to plant. Plant, on which is applying the control is laboratory fan heater.

# Obsah

<b>Úvod .....</b>	<b>6</b>
<b>1 Stav danej problematiky a opis použitých počítačových programov .....</b>	<b>7</b>
1.1 Stav danej problematiky .....	7
1.2 Matlab .....	7
1.3 Riadiaci systém SIMATIC S7- 300 .....	8
1.3.1 STEP7 .....	8
1.3.2 WinCC .....	9
1.4 Microsoft Excel .....	9
<b>2 Popis protokolu Dynamic Data Exchange.....</b>	<b>10</b>
2.1 Klient/Server princíp .....	10
2.1.1 Klient .....	10
2.1.2 Server.....	11
2.2 DDE konverzácia.....	11
2.2.1 Typy komunikačných liniek .....	11
2.2.1.1 Cold link .....	11
2.2.1.2 Hot link .....	12
2.2.1.3 Warm link .....	12
2.3 DDE terminológia.....	12
2.3.1 Služba .....	13
2.3.2 Téma .....	13
2.3.3 Prvok.....	13
<b>3 Funkcie Matlabu pre prácu s DDE a časovačom.....</b>	<b>14</b>
3.1 Matlab ako DDE server a klient .....	14
3.1.1 Matlab ako DDE server .....	14
3.1.2 Matlab ako DDE klient.....	15
3.2 Funkcie Matlabu pre prácu s časovačom.....	15
3.2.1 Vytvorenie a vymazanie objektu časovača.....	16
3.2.2 Vlastnosti objektu časovača.....	16
<b>4 Technický opis laboratórneho prietokového ohrievača vzduchu .....</b>	<b>18</b>
4.1 Laboratórne zariadenie LTR700.....	18
4.2 Ovládacie komponenty mechanickej zostavy.....	18
4.3 Ovládacie komponenty skrinky so zdrojom .....	18

<b>5 Konfiguračný a programový software STEP7.....</b>	<b>20</b>
5.1 Vytvorenie projektu.....	20
5.2 Konfigurácia siete.....	20
5.3 Konfigurácia hardwaru .....	20
5.4 Kalibrácia vstupných a výstupných signálov .....	22
5.4.1 Výkon špirály .....	22
5.4.2 Otáčky ventilátora.....	23
5.4.3 Teplota vzduchu .....	23
5.4.4 Prietok vzduchu .....	24
5.5 Programovanie blokov a konfigurácia premenných.....	25
<b>6 Vizualizačný software WinCC .....</b>	<b>28</b>
6.1 Práca s tagmi.....	28
6.1.1 Tagy spájajúce WinCC a Step7 .....	28
6.1.2 Tagy spájajúce WinCC a Microsoft Excel .....	28
6.2 Vytvorenie vizualizačnej obrazovky .....	32
6.3 Tvorba archívov.....	33
<b>7 Komunikácia Matlab – Microsoft Excel.....</b>	<b>35</b>
<b>8 Návrh LQ regulátora pre riadenie ohrievača vzduchu.....</b>	<b>36</b>
8.1 Polynomický LQ regulátor s integračnou činnosťou.....	38
8.2 Riadenie ohrievača vzduchu navrhnutým regulátorom .....	40
<b>Záver .....</b>	<b>43</b>
<b>Literatúra .....</b>	<b>44</b>
<b>Príloha A.....</b>	<b>45</b>
<b>Príloha B.....</b>	<b>48</b>

# Úvod

V súčasnej dobe sa na trhu objavuje veľké množstvo riadiacich systémov. Mnohé z nich majú vlastné aplikácie na vytváranie regulátorov, avšak pokiaľ si užívateľ chce navrhnúť špecifický regulátor v inom programe, musí nadviazať spojenie medzi jednotlivými produktmi. Jednou z možností je nadviazanie vzájomnej komunikácie týchto produktov pomocou protokolu DDE, ktorý vyvinula spoločnosť Microsoft na báze operačného systému MS Windows. Pri komunikácii dvoch produktov je jeden z nich server a druhý klient.

Cieľom tejto práce bolo vytvoriť komunikáciu medzi produktmi Simatic S7, Microsoft Excel a Matlab cez DDE rozhranie. Simatic S7 mal slúžiť na napojenie a vizualizáciu laboratórneho prietokového ohrievača vzduchu, Matlab na výpočet akčných zásahov pomocou LQ regulátora a Microsoft Excel ako DDE server pre oba produkty.

Komunikáciu medzi Matlabom a Simatic S7 cez OPC server som z dôvodu nedostatku licencií a času po dohode so školiteľom z diplomovej práce vypustil.

Po stručnom úvode sú v prvej kapitole stručne opísané produkty, na ktoré aplikujem komunikáciu a stav danej problematiky. Druhá kapitola sa venuje podrobnému opisu Dynamic Data Exchange (DDE) a jeho funkcií, posielaných správ medzi procesmi využívajúcimi túto technológiu. Tretia kapitola opisuje funkcie poskytované Matlabom na komunikáciu s DDE serverom, DDE server v produkte Matlab a funkcie Matlabu pre prácu s časovačom. Štvrtá kapitola sa zaoberá opisom laboratórneho prietokového ohrievača vzduchu a jeho napájania k skrinke so zdrojom. Piata kapitola opisuje vytvorenie projektu v programe STEP7 a jeho pripojenie k riadenému objektu. V šiestej kapitole je popísané vytvorenie vizualizačných obrazoviek a práca s tagmi. V siedmej kapitole je zhrnutá komunikácia medzi Matlabom a programom Microsoft Excel. Ôsma kapitola sa zaoberá identifikáciou systému, návrhom LQ regulátora s integračnou činnosťou a riadením systému.

# **1 Stav danej problematiky a opis použitých počítačových programov**

## **1.1 Stav danej problematiky**

V súčasnosti sa v priemysle využíva veľké množstvo riadiacich systémov. Mnohé z nich majú vlastné aplikácie na vytváranie regulátorov, prípadne preddefinované určité typy regulátorov, ktoré možno použiť po deklarovaní ich parametrov. Návrh vlastných regulátorov v aplikácii riadiaceho systému môže byť zložitý a často nedokáže splniť všetky požiadavky. Preto je snaha spájať rôzne druhy programov napríklad cez DDE alebo OPC. Táto technológia sa využíva všade tam, kde je potrebné zabezpečiť komunikáciu medzi viacerými produktmi. Väčšina produktov ju už obsahuje alebo je možné dokúpiť balíky na vytvorenie tejto komunikácie.

## **1.2 Matlab**

Matlab je vysokovýkonný integrovaný prostriedok pre technické výpočty. Je charakterizovaný integráciou výpočtov, vizualizáciou a programovaním v jednoduchom užívateľskom prostredí, kde problémy a riešenia sú vyjadrené bežnými matematickými zápismi. Zahŕňa nasledovné oblasti:

- Matematické výpočty
- Vývojové prostriedky pre tvorbu algoritmov
- Modelovanie a simulácia
- Vizualizácia a analýza dát
- Vysoko výkonná 2D a 3D grafika
- Aplikačné prostriedky pre vytváranie grafického užívateľského prostredia



Matlab je interaktívny systém, ktorého základné dátové prvky nemusia mať vopred definované veľkosti. Integráciou potrebných vlastností veľmi dobrého užívateľského programového prostriedku, Matlab umožňuje riešenie širokého spektra problémov. Dovoľuje tiež vytvárať program v neinteraktívnych programovacích jazykoch C alebo Fortran. Základnú časť Matlabu dopĺňajú špeciálne problémovo orientované knižnice inak nazývané toolboxy. Bohatá časť toolboxov umožňuje pokryť množstvo problémov, ako napríklad Optimalizácia, Fuzzy systémy, riadenie v reálnom čase a mnohé ďalšie.

### 1.3 Riadiaci systém SIMATIC S7- 300

Celý riadiaci systém pozostáva z dvoch hlavných častí, ktoré spolu tvoria kompletný celok

- **Pracovná stanica SIMATIC S7- 300** je univerzálny PLC schopný pôsobiť v širokom spektre aplikácií automatizačného inžinierstva s dôrazom na produkciu, ekonomické a praktické riešenia v rôznych odvetviach priemyslu. Pracovná stanica obsahuje:
  - *Napájací modul PS 307 2A*
  - *CPU 313C* je centrálna procesorová jednotka
  - *vstupno – výstupné I/O moduly*
- Programátorský a vizualizačný software **Step7** a **WinCC**

#### 1.3.1 STEP7

STEP7 je program od firmy SIEMENS, ktorý sa používa na realizáciu relatívne rozsiahlych a komplexných aplikácií, kde sa vyžaduje pre programovanie, funkcie alebo komunikačné moduly použitie jazykov vyššej úrovne a jazykov grafických návrhov. STEP7 je zlučiteľný s doplnkovým programovým balíkom Engineering tools. Slúži na naprogramovanie riadiaceho algoritmu, ktorý je následne prenesený cez rozhranie ( MPI, Profibus...) do PLC.

### **1.3.2 WinCC**

Program WinCC slúži na vytvorenie vizualizačných okien, pomocou ktorých je možné bežiaci proces monitorovať a aktívne do neho zasahovať. Všetky dôležité dáta čerpá z programu STEP7, s ktorým je prepojený prostredníctvom tagov ( dátové kanály). Počítač, na ktorom je nainštalovaný WinCC musí byť pripojený k PLC, v ktorom je riadiaci program cez rozhrania uvedené v kapitole 1.3.1.

### **1.4 Microsoft Excel**

Aplikácia Microsoft Excel je obsiahnutá v kancelárskom balíku Microsoft Office. Slúži na vytváranie a správu tabuliek, na riešenie rôznych funkčných závislostí. Využíva sa v takmer vo všetkých druhoch obchodných spoločností a vo všetkých odvetviach priemyslu.

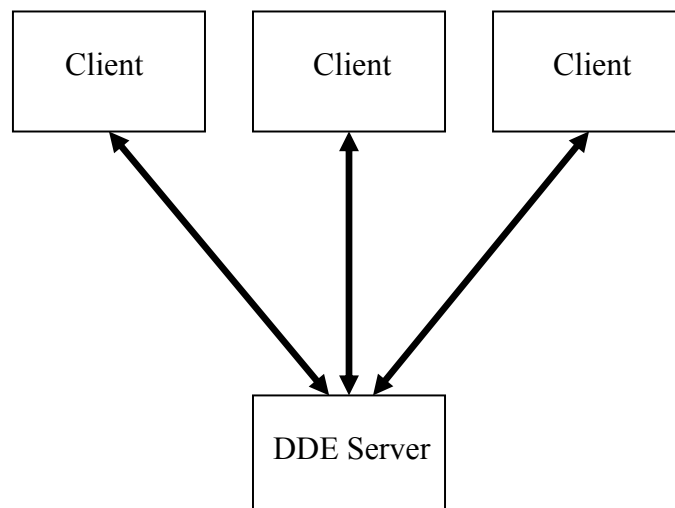
## 2 Popis protokolu Dynamic Data Exchange

Dynamic Data Exchange ( DDE ) je protokol, ktorý vyvinula spoločnosť Microsoft pre vzájomnú komunikáciu aplikácií na báze operačného systému MS Windows. Umožňuje komunikáciu v reálnom čase, či už medzi aplikáciami na jednom počítači, alebo medzi aplikáciami na jednom počítači, alebo v počítačovej sieti. Protokol je vytvorený na báze Server – Klient. DDE používa na výmenu dát medzi aplikáciami zdieľanú pamäť. Aplikácie môžu využívať DDE na jednorázový prenos údajov, ale aj na kontinuálny prenos a aktualizáciu dát.

### 2.1 Klient/Server princíp

#### 2.1.1 Klient

Klient je aplikácia, ktorá inicializuje DDE konverzáciu a riadi výmenu dát. Klient riadi logický komunikačný kanál, otvára ho, posíla požiadavky a opäť ho zatvára. Klient môže otvoriť viac ako jedno spojenie so Serverom a môže pristupovať viacerými premennými na spojenie. ( obr. 1 )



Obr.1 Možné spojenia medzi klientom a serverom

### 2.1.2 Server

Server je aplikácia, ktorá odpovedá na žiadosti o informácie od klienta. Server je pasívny partner v konverzácii a odpovedá iba žiadostiam o informácie od aktívneho partnera t.j. klienta.

## 2.2 DDE konverzácia

Aplikácie komunikujú medzi sebou po nadviazaní DDE konverzácie. Typický postup krokov v DDE konverzácii je nasledovný:

- DDE klient spustí DDE konverzáciu otvorením spojenia so Serverom. DDE Server potvrdí otvorenie DDE spojenia.
- Výmena dát s aplikáciami.
- Klient alebo Server ukončí DDE komunikáciu.

### 2.2.1 Typy komunikačných liniek

Sú tri rozdielne typy komunikačných liniek: **cold link**, **warm link** a **hot link**.

#### 2.2.1.1 Cold link

Jednorázový príjem dát je spôsobený jednorázovou požiadavkou od klienta alebo jednorázovým prenosom dát od klienta k Serveru. Cold link konverzácia sa používa na čítanie premenných. V prípade dynamického vymieňania dát v automatickom systéme, klient môže požiadať rovnaký premenný čas a čas opäť zostať aktuálny. V tomto osobitnom prípade, kde závisí od frekvencie žiadostí o informácie, nie je záruka, že všetky zmeny premennej budú zachytené. Na druhej strane, väčšina žiadostí o informácie by zostali neúčinné, ak sa dáta zmenili pomaly. Toto je dôvod, prečo sú DDE výkonnejšie typy konverzácie pre dynamické vymieňanie dát.

- **Príjem:** Klient pošle požiadavku o informácie Serveru. Ak je Server v pozícii poskytnúť tieto dáta, pošle ich klientovi. Ak nie, pošle negatívne potvrdenie. Žiadosti klienta o informácie môžu byť opakované tak často ako je potrebné, až kým klient alebo Server uzavrie DDE spojenie.
- **Odoslanie:** Klient jednorázovo posíla dáta Serveru.

### 2.2.1.2 Hot link

Priebežný príjem dát je spôsobený jednorázovou požiadavkou. Pri tomto type konverzácie, klient posíla jednorázovo žiadosť o dáta Serveru. To znamená, že Server má za úlohu oznámiť klientovi zmenu hodnoty jednotlivých prvkov. Keď Server potvrdí túto požiadavku, pošle všetky zmeny hodnôt prvkov klientovi bez toho aby klient musel poslať ďalšiu požiadavku Serveru. Zmeny sú posielané dovedy, kým klient neukončí dátové spojenie. Tento typ konverzácie je vhodný pre žiadanie zmeny dát a používa sa, keď si klient želá aktualizovať premennú. Akonáhle sú prítomné aktualizované dáta, sú poslané cez DDE linku. Toto zaručí, že aj nepravidelne meniace sa dáta sú spracované v Serveri.

### 2.2.1.3 Warm link

Priebežný príjem odkazov (ako opak k dátam) je spôsobený jednorázovou požiadavkou od klienta. Tento typ konverzácie je kombináciou cold a hot linky. Na rozdiel od hot linky, klient neprijíma aktualizované dáta bez požiadania. Tento typ konverzácie sa používa, keď si klient želá byť informovaný o aktualizovanej premennej, ale chce riadiť dátový príjem.

## 2.3 DDE terminológia

Aplikácie komunikujú medzi sebou po nadviazaní DDE konverzácie. Pri nadviazaní DDE komunikácie, musí klient poslať dva parametre: *Názov služby* a *Názov prvku* ( *Item name* ). Keď serverová aplikácia prijme požiadavku na konverzáciu, ktorá obsahuje podporovanú tému, potvrdí danú požiadavku a vytvorí DDE konverzáciu. Kombinácia

*Názov služby* a *Názov témy* jednoznačne identifikujú konverzáciu. Ani jeden z týchto dvoch parametrov sa nesmie počas trvania konverzácie meniť. Jedna služba môže obsahovať viac tém. Počas DDE konverzácie si Server a Klient vymieňajú dáta súvisiace s prvkom. Prvok je odkaz na dáta, ktoré sú významné pri konverzácii pre obe aplikácie. Hociktorá z aplikácií môže prvok zmeniť.

### **2.3.1 Služba**

Každá aplikácia, ktorá môže byť DDE serverom, má jednoznačný *Názov služby*. Zvyčajne je to názov spúšťačieho súboru aplikácie bez prípony.

### **2.3.2 Téma**

Téma definuje subjekt DDE konverzácie.

### **2.3.3 Prvok**

Každá téma obsahuje jeden alebo viac prvkov. Prvok identifikuje dáta posielené počas DDE konverzácie.

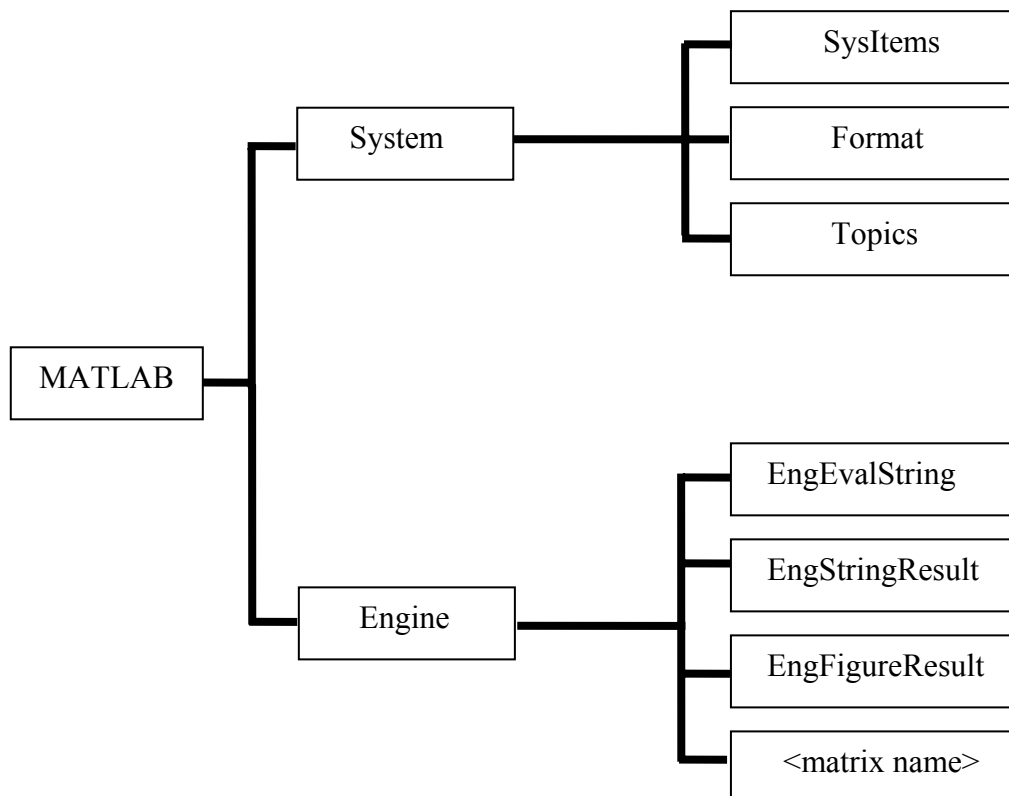
### 3 Funkcie Matlabu pre prácu s DDE a časovačom

#### 3.1 Matlab ako DDE server a klient

Matlab poskytuje funkcie, ktoré umožňujú pristupovať k dátam iných aplikácií cez DDE a taktiež odpovedať iným aplikáciám na ich požiadavky.

##### 3.1.1 Matlab ako DDE server

Klientská aplikácia môže pristupovať k DDE serveru Matlabu po zadaní Názvu služby „matlab“. Matlab má v sebe zabudované dve témy. Systémová má názov „system“ a dátová má názov „engine“. Hierarchia DDE servera v Matlabe je znázornená na obr. 2. Význam jednotlivých funkcií je zhrnutý v tab. 1.



Obr. 2 Hierarchia DDE servera v Matlabe

Prvok	Opis
<b>SysItems</b>	Zoznam prvkov podporovaných témou „system“.
<b>Format</b>	Zoznam formátov podporovaných serverom.
<b>Topics</b>	Zoznam tém podporovaných serverom.
<b>EngEvalString</b>	Príkaz posielať Matlabu na vykonanie.
<b>EngStringResult</b>	Výsledok vykonaného príkazu.
<b>EngFigureResult</b>	Grafický výsledok vykonaného príkazu.
<b>&lt;matrix name&gt;</b>	Premena, ktorej dáta klient požaduje.

Tab. 1 Stručný popis prvkov DDE servera Matlabu

### 3.1.2 Matlab ako DDE klient

Matlab poskytuje používateľovi pomocou definovaných príkazov možnosť pristupovať k ľubovoľnému DDE serveru v systéme. Jednotlivé funkcie a ich význam je uvedený v tab. 2.

Funkcia	Opis
<b>ddeinit</b>	Inicializácia komunikácie so serverom
<b>ddeterm</b>	Ukončenie komunikácie so serverom
<b>ddeadv</b>	Žiadosť o nadviazanie upozorňovacej slučky so serverom
<b>ddeunadv</b>	Ukončenie upozorňovacej slučky so serverom
<b>ddereq</b>	Žiadosť o dáta zo serveru
<b>ddepoke</b>	Poslanie dát serveru
<b>ddeexec</b>	Žiadosť o vykonanie príkazu serveru

Tab. 2 DDe funkcie Matlabu

## 3.2 Funkcie Matlabu pre prácu s časovačom



Matlab poskytuje objekt časovača, ktorý je možné použiť na plánovanie vykonania rôznych úloh. Spustenie načasovanej úlohy neznamená jej vykonanie ale jej zaradenie do radu úloh čakajúcich na vykonanie. Postup pre použitie časovača je nasledovný:

- Vytvorenie objektu časovača
- Nastavenie parametrov časovača
- Spustenie časovača

Na konci je treba objekt časovača zastaviť a vymazať.

### 3.2.1 Vytvorenie a vymazanie objektu časovača

Časovač vytvoríme pomocou funkcie „**timer**“. Po ukončení práce s časovačom ho vymažeme z pamäte pomocou funkcie „**delete**“.

### 3.2.2 Vlastnosti objektu časovača

Vlastnosti časovača možno rozdeliť do štyroch skupín:

- **Informačné parametre** – popisujú súčasný stav časovača. Tieto parametre užívateľ nemôže meniť. Uvedené sú v tab. 3.

Parameter	Opis
<b>AveragePeriod</b>	Priemerná perióda vykonávania obslužnej funkcie
<b>InstantPeriod</b>	Posledná hodnota Periódy vykonania obslužnej funkcie
<b>Running</b>	Súčasný stav časovača („on“ alebo „off“)
<b>TaskExecuted</b>	Počet vykonaní obslužnej funkcie os spustenia

Tab. 3 Informačné parametre objektu časovača v Matlabe

- **Parametre správania** – udávajú časové správanie sa časovača. Uvedené sú v tab. 4.

Parameter	Opis
<b>BusyMode</b>	Udáva spôsob ďalšieho spracovania obslužnej funkcie, ak posledné spracovanie nebolo ukončené
<b>ExecutionMode</b>	Spôsob zorad'ovania obslužnej funkcie
<b>Period</b>	Periódá medzi dvomi vykonaniami v sekundách
<b>StartDelay</b>	Časový interval pred prvým vykonaním v sekundách
<b>TasksToExecute</b>	Zostávajúci počet vykonaní obslužnej funkcie

Tab. 4 Parametre správania sa objektu časovača v Matlabe

- **Parametre obslužných funkcií** – špecifikujú funkcie, ktoré sa majú vykonať pri danej udalosti. Uvedené sú v tab. 5.

Funkcia	Opis
<b>ErrorFcn</b>	Funkcia vykonaná v prípade chyby
<b>StartFcn</b>	Funkcia vykonaná pri štarte časovača
<b>StopFcn</b>	Funkcia vykonaná pri zastavení časovača
<b>TimerFcn</b>	Funkcia vykonaná časovačom v každom kroku

Tab. 5 Parametre obslužných funkcií objektu časovača v Matlabe

- **Parametre definované užívateľom** – umožňujú pridať vlastné dáta. Uvedené sú v tab. 6.

Funkcia	Opis
<b>Name</b>	Textový reťazec identifikujúci časovač
<b>Tag</b>	Užívateľom definovaný popis časovača
<b>UserData</b>	Užívateľom definované dáta

Tab. 6 Parametre definované užívateľom objektu časovača v matlabe

DDE rozhranie Matlabu a práca s časovačmi je podrobne popísaná v [1].

## **4 Technický opis laboratórneho prietokového ohrievača vzduchu**

### **4.1 Laboratórne zariadenie LTR700**

Laboratórne zariadenie LTR700 zabezpečuje vzduch o žiadanom prietoku a teplote. LTR700 sa skladá z dvoch základných častí:

- z mechanickej zostavy (trubice), ktorá obsahuje ventilátor a ohrevnú špirálu, diferenciálny tlakový snímač a snímač teploty
- zo skrinky so zdrojom, ktorá má dve fixné prepojenia na mechanickú zostavu

### **4.2 Ovládacie komponenty mechanickej zostavy**

Skoková funkcia prietoku vzduchu sa inicializuje pomocou regulačnej klapky. Klapka je priamo prepojená s pákou, ktorá je pomocou magnetu udržiavaná vo vertikálnej pozícii. V tejto polohe je minimálny prietok vzduchu, pohyb páky do horizontálnej polohy vedie k otvoreniu klapky, čo vedie k maximálnemu prietoku. Táto páka sa ovláda manuálne.

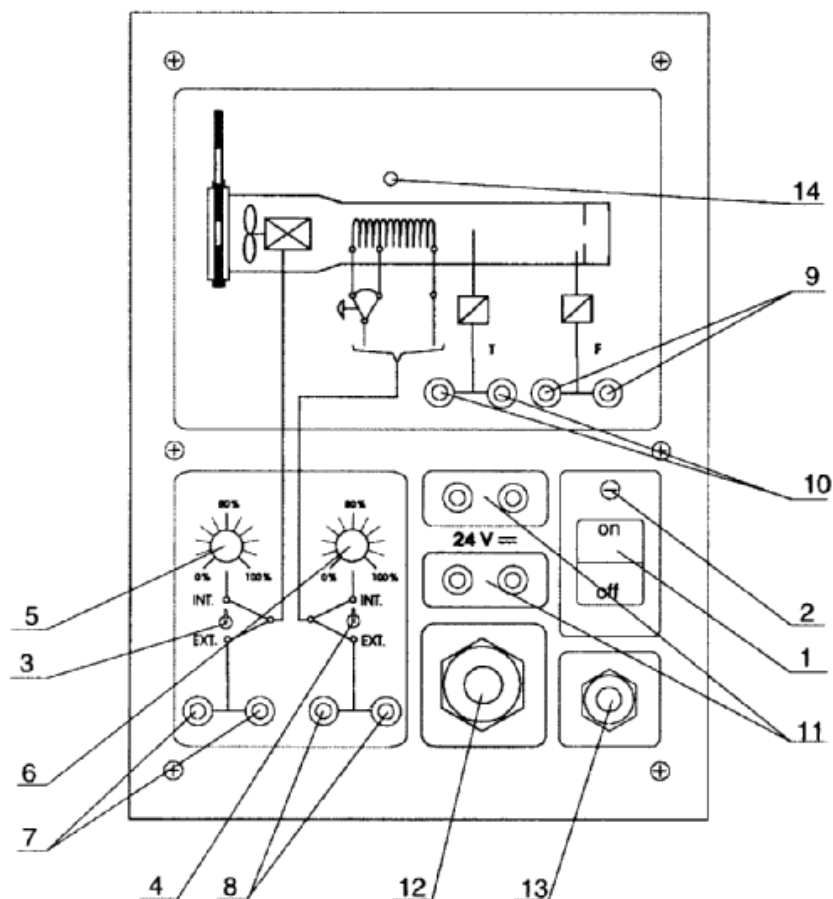
Skoková funkcia ohrevu sa dosiahne pomocou vypínača s polohami „0“ a „1“, ktorý sa nachádza na pripojenej skrinke. Minimálny ohrev je pri polohe „0“ a maximálny pri polohe „1“.

### **4.3 Ovládacie komponenty skrinky so zdrojom**

Vstupná jednotka prepojenia na sieť sa nachádza na zadnom paneli skrinky, ktorá má dve poistky v podobe sklenných trubíc ako aj vstupný filter. Na obr. 3 sú komponenty nachádzajúce sa na prednom paneli skrinky. Jednotlivé označenia na obrázku znamenajú:

1. vypínač
2. svetelná kontrolka signalizujúca zapnutie zariadenia
3. prepínač medzi interným alebo externým riadením otáčok ventilátora

4. prepínač medzi interným alebo externým riadením výkonu špirály
5. potenciometer na nastavenie otáčok ventilátora pri internom riadení
6. potenciometer na nastavenie výkonu špirály pri internom riadení
7. vstup na externé riadenie otáčok ventilátora
8. vstup na externé riadenie výkonu špirály
9. meraný výstup zo senzora prietoku vzduchu
10. meraný výstup zo senzora teploty vzduchu
11. Dva elektricky izolované napäťové výstupy
12. vývod pre dátový kábel meraných signálov a riadiaceho signálu otáčok
13. vývod pre riadiaci signál ohrevu
14. svetelná kontrolka indikujúca činnosť špirály



Obr. 3 Predný panel skrinky so zdrojom

Oba vstupy a výstupy z ohrievača pracujú s prúdovým signálom v rozmedzí 4-20mA. Kompletná dokumentácia k zariadeniu je uvedená v [4].

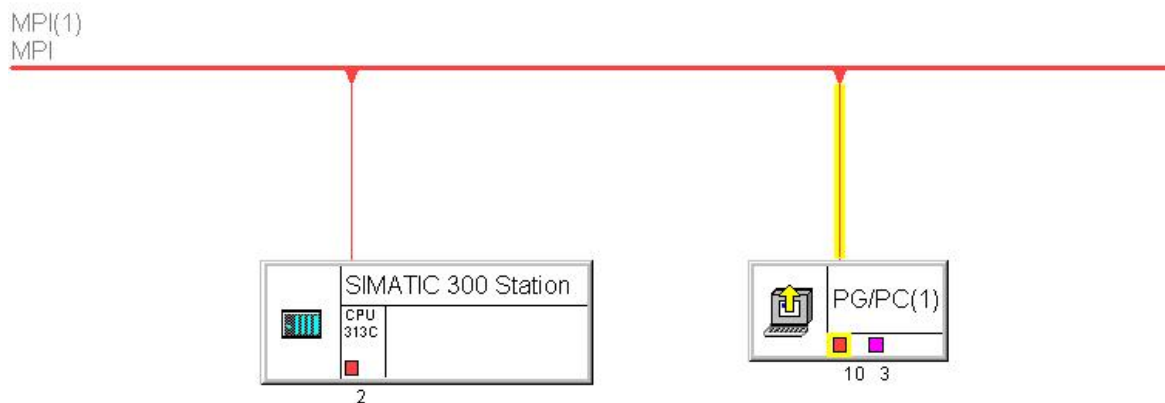
## 5 Konfiguračný a programový software STEP7

### 5.1 Vytvorenie projektu

Projekt bol vytvorený pomocou *Project Wizard*, v ktorom sa zadáva názov projektu, typ procesora, typ spojenia...

### 5.2 Konfigurácia siete

V tejto časti sa vytvorí prepojenie počítača s pracovnou stanicou, zvolilo sa prepojenie cez MPI rozhranie. Nakonfigurovaná sieť je na obr.4.

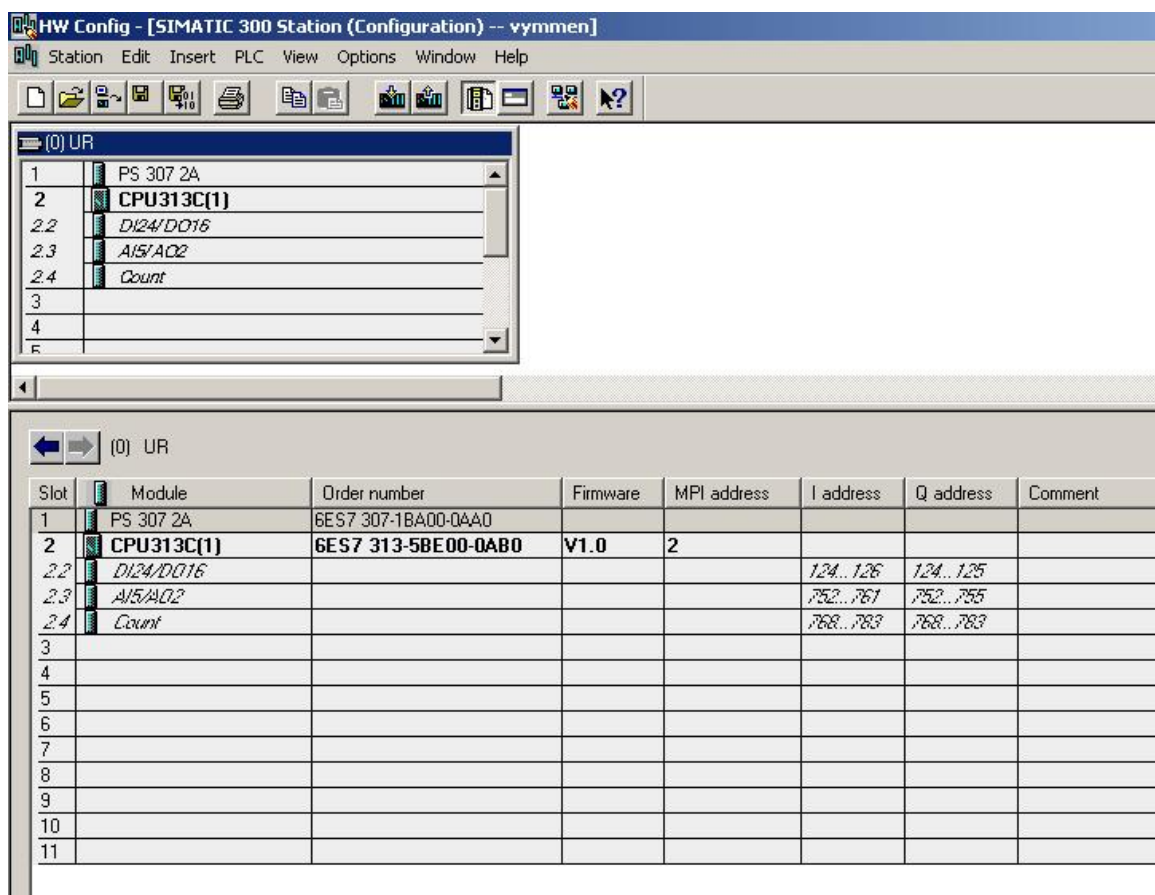


Obr. 4 Nakonfigurovaná sieť

### 5.3 Konfigurácia hardwaru

Prepojenie snímačov a akčných členov s PC je zabezpečené pomocou riadiacej stanice SIMATIC S7-300, s ktorou sú prepojené analógové vstupno – výstupné I/O moduly

AI5/AO2 x 12Bit a DI16/DO16 x DC24V schopné spracovávať prúdový alebo napäťový signál. Keďže všetky zariadenia ohrievača pracujú s prúdovým signálom (4 - 20mA), použil som na ich pripojenie k pracovnej stanici analógový I/O modul AI5/AO2 x 12Bit, ktoré majú 5 vstupných a 2 výstupné kanále, rozlišovaciu schopnosť 12 bitov, meraný a výstupný rozsah 0 - 20mA. Ako je vidieť na obr. 5 daný *Rail* predstavuje celú pracovnú stanicu a jeho riadky jednotlivé sloty, v ktorých sú odsadené jednotlivé moduly. Prvý slot je rezervovaný pre napájací modul PS, druhý pre CPU ale keďže CPU a I/O moduly pri tomto zapojení tvoria celok, druhý slot je rozvetvený, a v jednotlivých vetvách okrem prvej (CPU), sa nachádzajú I/O moduly a tzv. *Count*, ktorý slúži na nastavenie brány (softwarovej alebo hardwarovej) pre procesor.



Obr. 5 Okno konfigurácie hardwaru

## 5.4 Kalibrácia vstupných a výstupných signálov

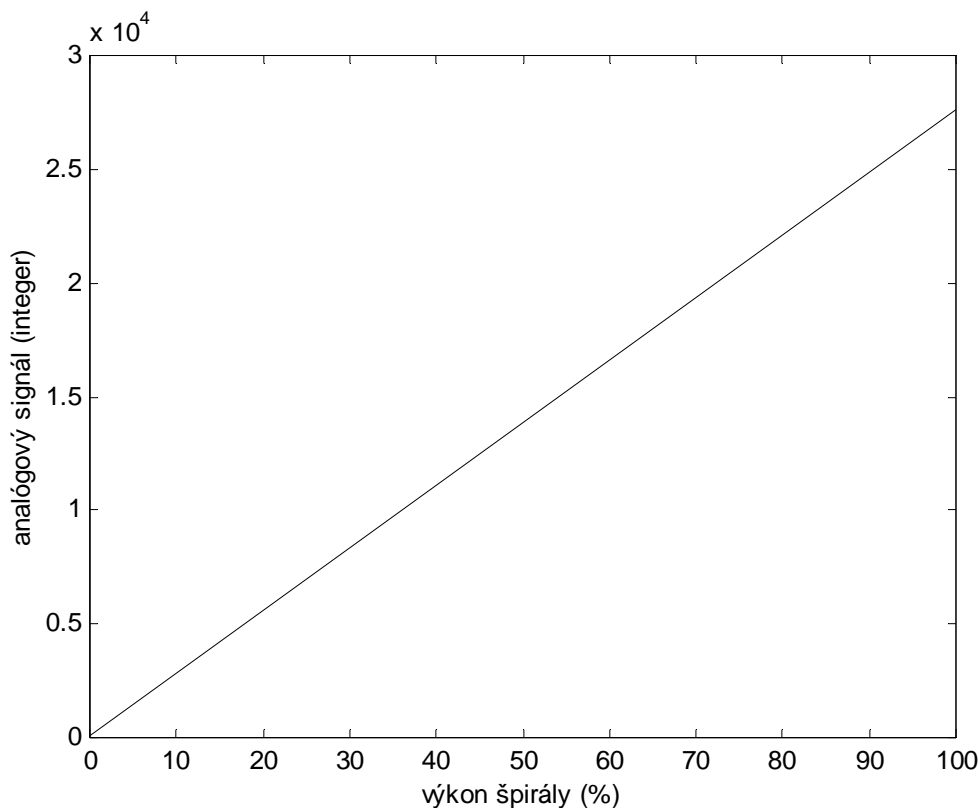
Keďže bolo potrebné snímať teplotu vzduchu (°C) a prietok vzduchu (%) a bolo potrebné posielat' údaje do špirály (%) a do ventilátora (%), museli byť výstupy z ohrievača prevedené z analógového signálu a vstupy do ohrievača na analógový signál. Pomocou mikroampérmetra bol snímaný prúdový signál (výkon špirály, otáčky ventilátora) pri rôznych hodnotách analógových signálov. Pri teplote a prietoku vzduchu boli generované rôzne hodnoty prúdového signálu a zisťovali sa hodnoty analógového signálu.

### 5.4.1 Výkon špirály

Prevod výkonu špirály na výstupný analógový signál je zobrazený na obr. 6.

Rovnica regresnej priamky na obr. 6 je

$$y = 275,4x + 60 \quad (1)$$



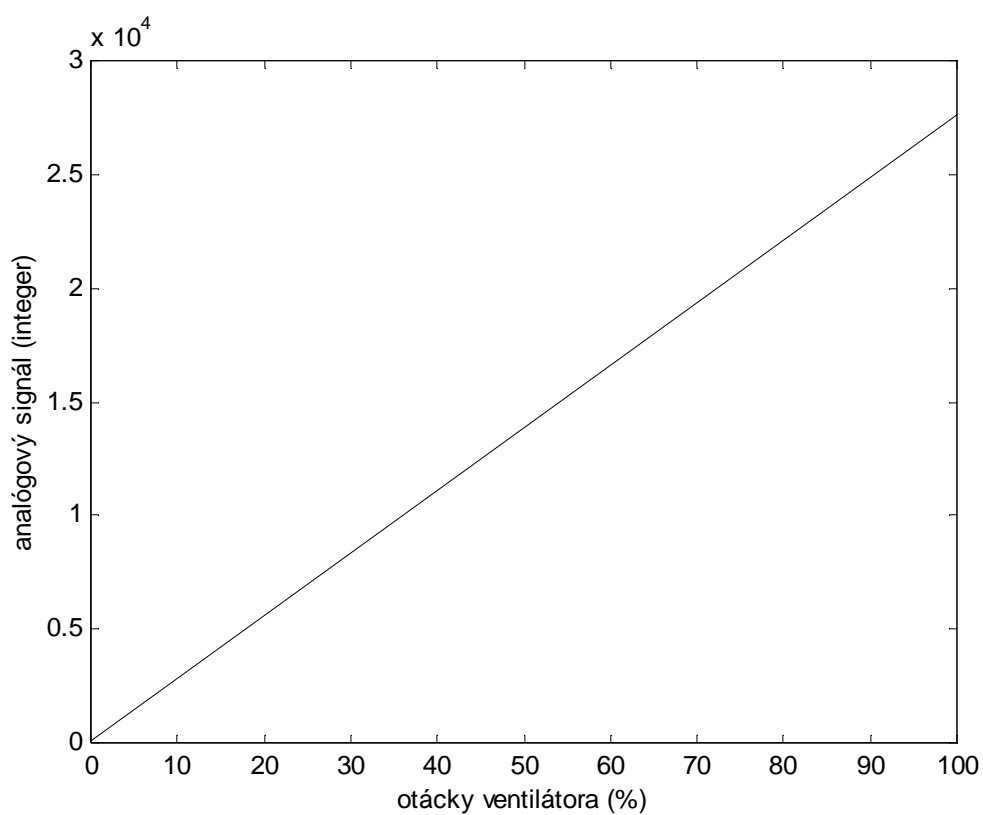
Obr. 6 Statická charakteristika prevodu výkonu špirály na výstupný signál integer

### 5.4.2 Otáčky ventilátora

Prevod otáčok ventilátora na výstupný analógový signál je zobrazený na obr. 7.

Rovnica regresnej priamky na obr. 7 je

$$y = 275,4x + 60 \quad (2)$$



Obr. 7 Statická charakteristika prevodu otáčok ventilátora na výstupný signál integer

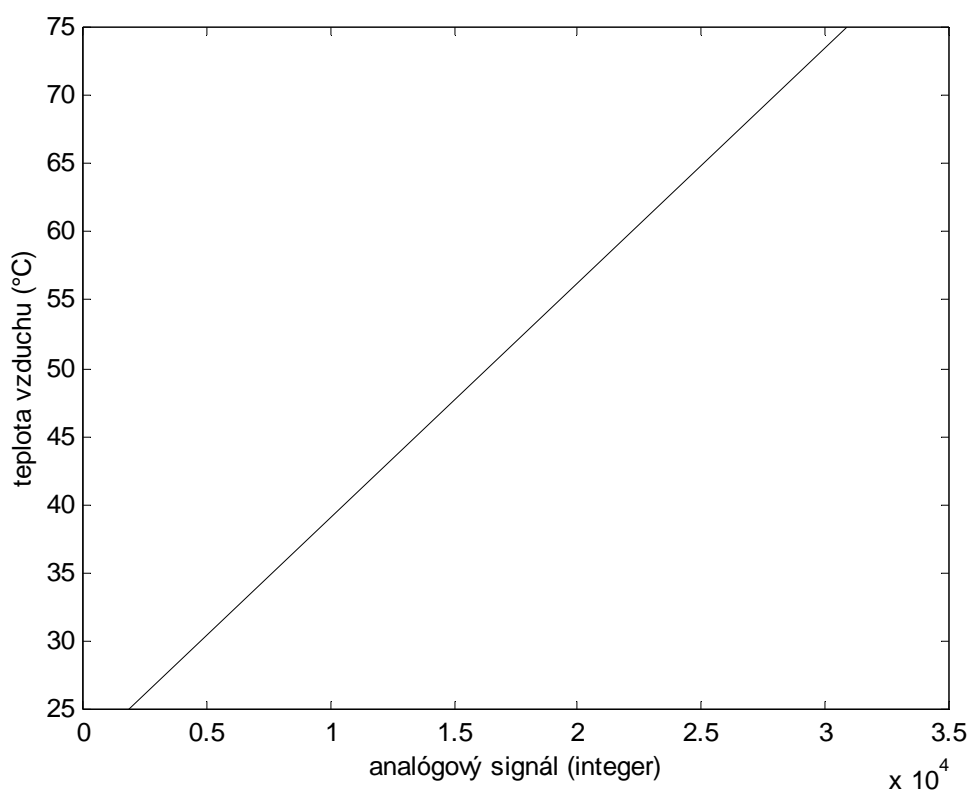
### 5.4.3 Teplota vzduchu

Prevod analógového signálu na stupne Celzia zobrazený na obr. 8.



Rovnica regresnej priamky na obr. 8 je

$$y = 17,23 \cdot 10^{-4} x + 21,78 \quad (3)$$



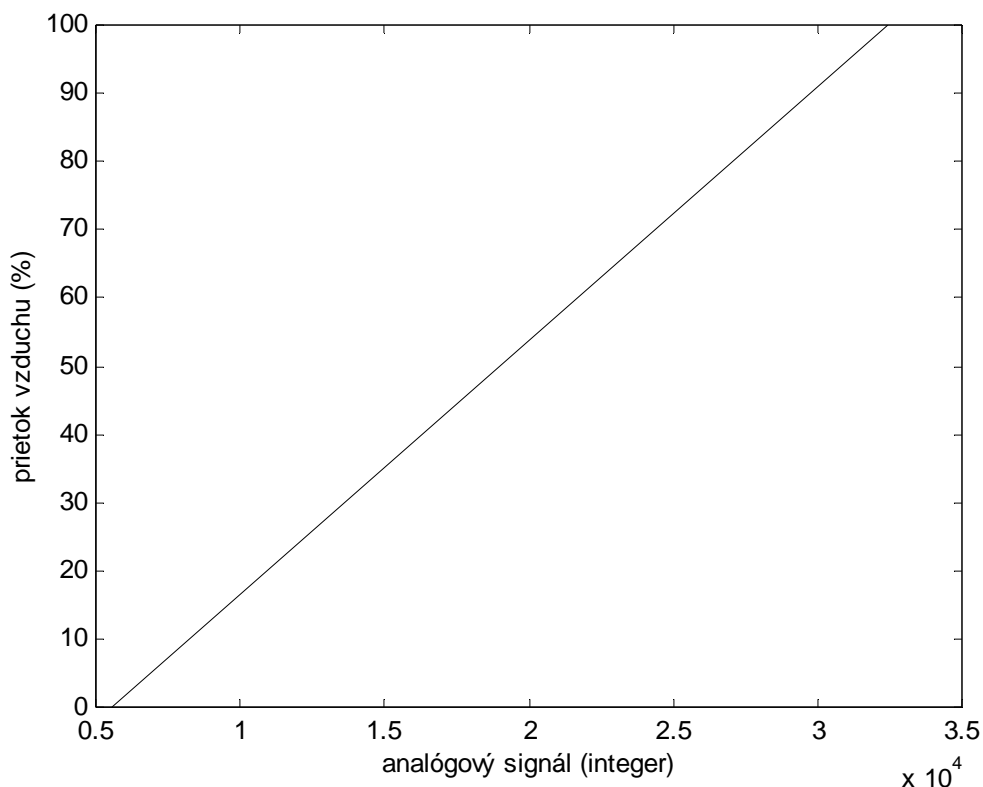
Obr. 8 Statická charakteristika prevodu integer-u na stupne Celzia

#### 5.4.4 Prietok vzduchu

Prevod analógového signálu na % prietoku vzduchu zobrazený na obr. 9.

Rovnica regresnej priamky na obr. 9 je

$$y = 37,202 \cdot 10^{-4} x - 20,714 \quad (4)$$

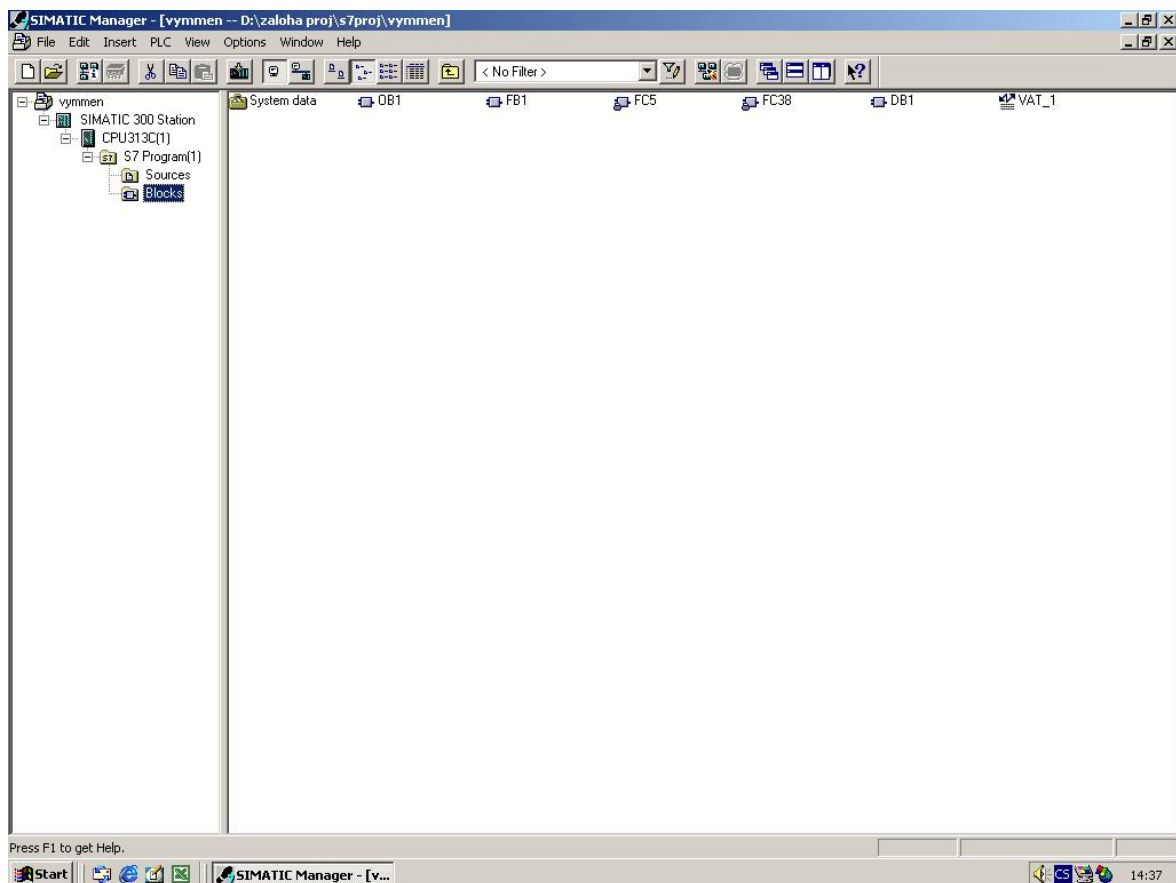


Obr. 8 Statická charakteristika prevodu integer-u na % prietoku vzduchu

## 5.5 Programovanie blokov a konfigurácia premenných

Prehľad použitých blokov je na obr. 9. Hlavný program bol napísaný v organizačnom bloku OB1, ktorý bol napísaný pomocou FBD (programovanie pomocou blokovej schémy). Program v OB1 využíva aj funkčný blok FB1, ktorý je zvlášť naprogramovaný a s ktorému je priradený dátový blok DB1, ktorý poskytuje údaje pre tento blok. Program je napísaný v tzv. sieťach, do ktorých sa vkladajú jednotlivé bloky, viď príloha A.

Všetky premenné okrem premenných použitých v FB1, boli zadefinované v *Symbols Editor*, ktorý umožňuje definovať nové premenné, priradiť im symbolický názov, meniť ich adresy a pridávať k nim komentár. Taktiež je v ňom možné priradiť symbolický názov danému typu bloku a pripísať k nemu komentár. Premenné v bloku FB1 boli definované v editore symbolov, ktorý sa nachádza priamo v bloku FB1. *Symbol editor* pre ohrievač vzduchu je uvedený na obr. 10.



Obr. 9 Okno projektu s informáciou o použitých blokoch

S7 Program(1) (Symbols) -- vymmen\SIMATIC 300 Station\CPU313C(1)					
	Sta	Symbol	Address	Data type	Comment
1		teplota_R	MD 20	REAL	teplota vzduchu R
2		Cycle Execution	OB 1	OB 1	organizačný blok
3		VAT_1	VAT 1		tabuľka premenných
4		tepl	PIW 754	INT	teplota vzduchu
5		sucinpl	MD 30	REAL	výsledok súčinu pri teplote vzduchu
6		spir	PQW 754	INT	vstup do reálneho zariadenia, výkon spirály
7		teplota	MD 40	REAL	teplota vzduchu
8		zasah2	MD 200	REAL	akčný zásah, výkon spirály
9		vent	PQW 752	INT	vstup do reálneho zariadenia, otáčky ventilátora
10		teplota_DI	MD 10	DINT	teplota výdychu DI
11		priet	PIW 752	INT	prietok výdychu
12		DI_STRNG	FC 5	FC 5	Double Integer to String
13		STRNG_I	FC 38	FC 38	String to Integer
14		zasah1	MD 110	REAL	akčný zásah, otáčky ventilátora
15		sucinpriet	MD 70	REAL	výsledok súčinu pri prietoku vzduchu
16		priet_DI	MD 50	DINT	prietok vzduchu DI
17		prietok_vzduchu	MD 80	REAL	prietok vzduchu
18		priet_R	MD 60	REAL	prietok vzduchu R
19					

Obr. 10 Symbol editor pre ohrievač vzduchu

Po napísaní celého programu je potrebné prekopírovať ho do RAM pamäte. Otočením kľúča do polohy **RUN** sa program spustí. Ak je potrebné program zastaviť, kľúč sa otočí do polohy **STOP**.

Pre otestovanie celého programu bol použitý blok *Variable Table*, do ktorého boli zapísané adresy na akčné veličiny (výkon špirály a otáčky ventilátora).





## 6 Vizualizačný software WinCC

### 6.1 Práca s tagmi

WinCC je prepojený s programom napísaným v programe STEP7 a s programom Microsoft Excel prostredníctvom externých tagov. Tag je virtuálny dátový kanál, cez ktorý prechádzajú dáta.

#### 6.1.1 Tagy spájajúce WinCC a Step7

Externé tagy, spájajúce WinCC a STEP7 sa definujú v *Tag Managment-e* po pridaní komunikačného driver-u *SIMATIC S7 PROTOCOL SUITE*. Pre ovládanie laboratórneho prietokového ohrievača vzduchu boli vytvorené tagy na snímanie teploty a prietoku vzduchu, tagy pre výkon špirály a ventilátora. obr. 11 ukazuje vytvorené tagy aj s adresami. Každý tag bol pripojený na rovnakú adresu, aká bola deklarovaná v programe STEP7.

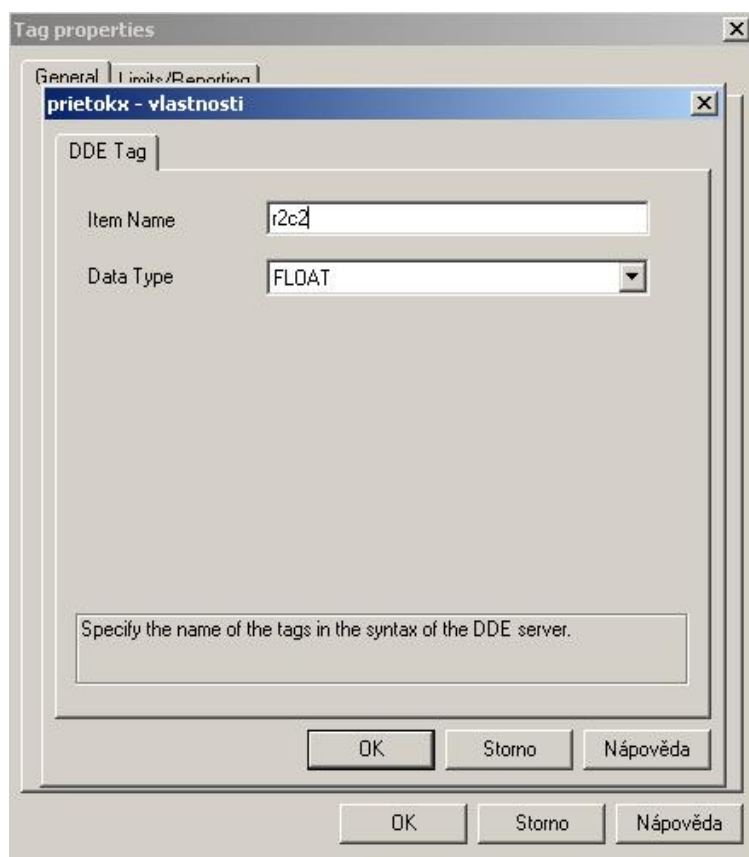
Name	Type	Parameters	Last Change
 teplota	Floating-point number 32-bit IEEE ...	MD40	3/30/2004 5:11:43 PM
 prietok	Floating-point number 32-bit IEEE ...	MD80	3/30/2004 5:12:02 PM
 spirála	Floating-point number 32-bit IEEE ...	MD200	4/20/2004 2:35:24 PM
 vent	Floating-point number 32-bit IEEE ...	MD110	4/20/2004 2:26:40 PM

Obr. 11 Zoznam tagov spájajúcich WinCC a STEP7

#### 6.1.2 Tagy spájajúce WinCC a Microsoft Excel

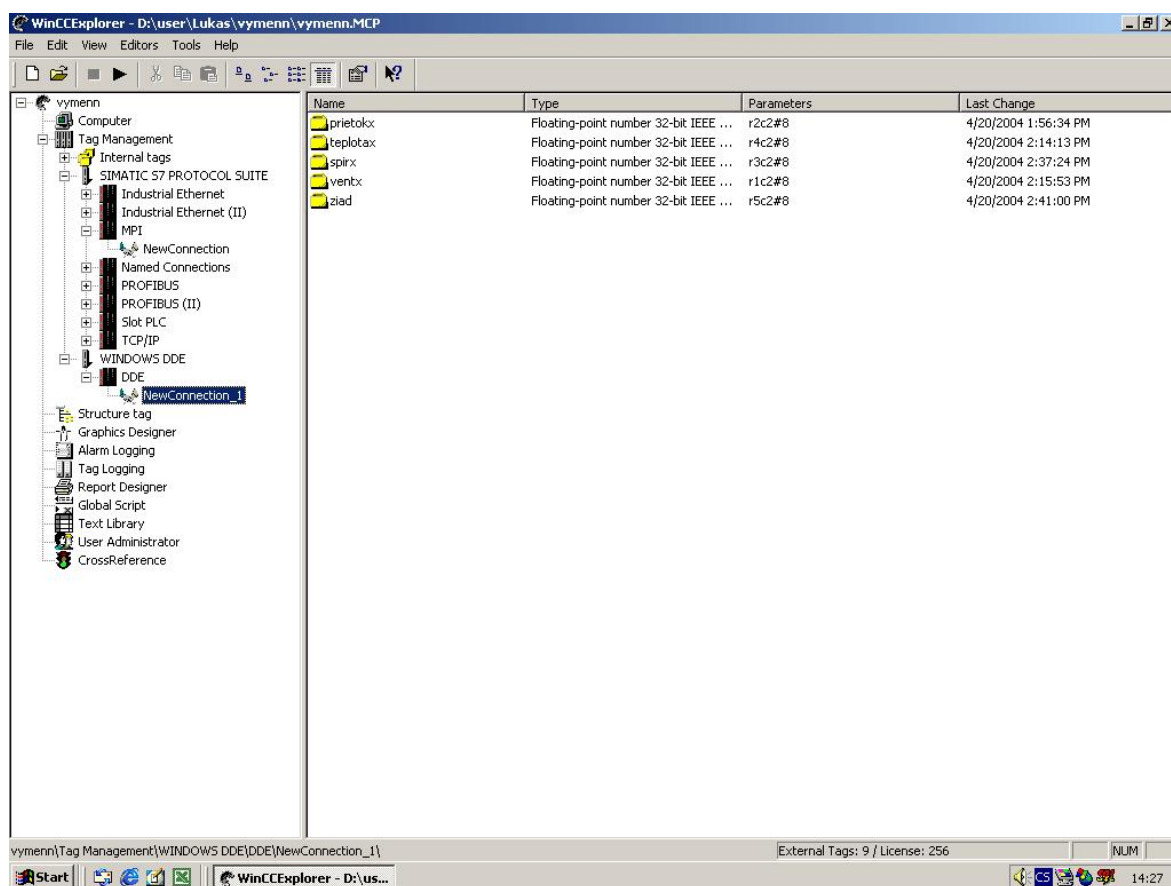
Externé tagy, spájajúce WinCC a Microsoft Excel cez DDE rozhranie sa definujú po pridaní komunikačného driver-u *WINDOWS DDE*. Pri vytváraní tohto spojenia bolo treba

zadefinovať názov programu, s ktorým sa chceme spojiť a názov súboru, vytvoreného v programe Microsoft Excel, s ktorým chceme komunikovať. Bolo potrebné vytvoriť všetky tagy ako v prípade spojenia WinCC so STEP7 a ešte navyše tag pre žiadanú hodnotu teploty. Pri ich vytváraní bolo potrebné zadať, do ktorej bunky v súbore vytvorenom v programe Microsoft Excel sa má daná hodnota zobraziť, prípadne z ktorej bunky sa má hodnota načítať ( obr. 12).



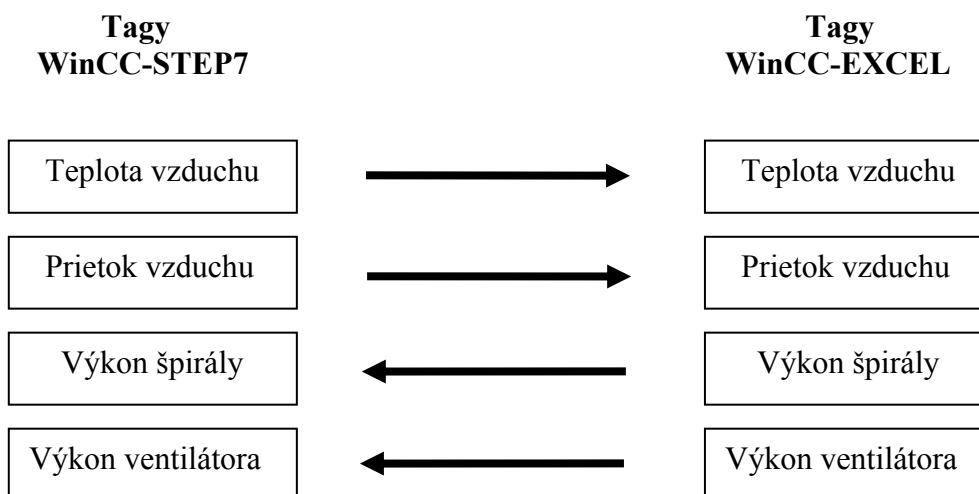
Obr. 12 Konfiguračné okno tagu pre prietok vzduchu

Po nastavení týchto údajov, je WinCC po aktivácii projektu schopný komunikovať cez DDE rozhranie s programom Microsoft Excel. Nakonfigurované okno s tagmi je na obr. 13.



Obr. 13 Nakonfigurované okno s tagmi

Po nastavení všetkých tagov bolo potrebné tagy spájajúce WinCC a STEP7 s tagmi spájajúcimi WinCC a Microsoft Excel navzájom prepojiť. Bolo potrebné kopírovať hodnoty medzi tagmi. Tok hodnôt medzi jednotlivými tagmi je na obr. 14.



Obr. 14 Tok hodnôt medzi jednotlivými tagmi

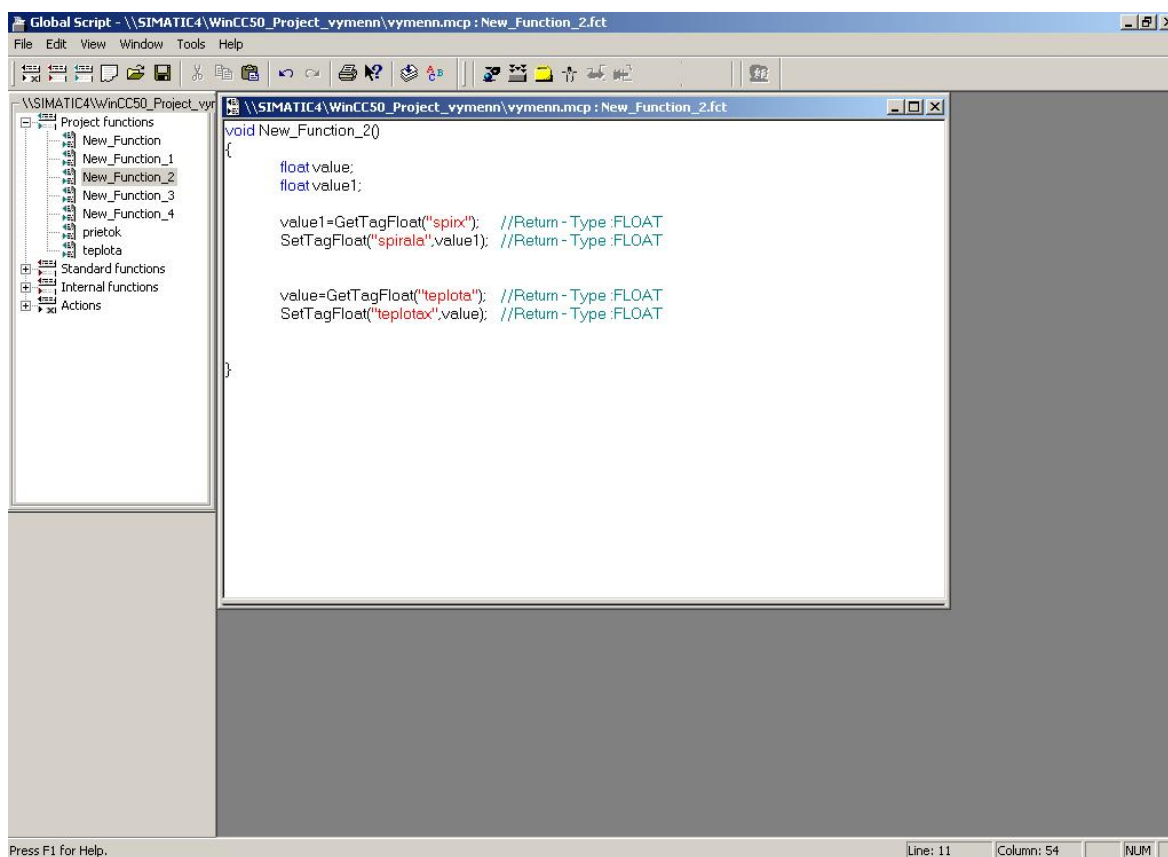
Kopírovanie jedného tagu do druhého bolo uskutočnené pomocou programu (*Project functions*), napísaného v module *Global Script*. Boli použité nasledovné príkazy:

*GetTagFloat*("tag") – načíta hodnotu tagu do vybranej premennej

*SetTagFloat*("tag",premenná) – nastaví hodnotu tagu na hodnotu premennej

Na začiatku programu bolo potrebné zadať typ premennej. Keďže išlo o čísla s desatinnými miestami, typ premennej bol *float*.

Na obr. 15 je otvorený modul *Global Script* s napísaním programom pre teplotu vzduchu a pre výkon špirály.



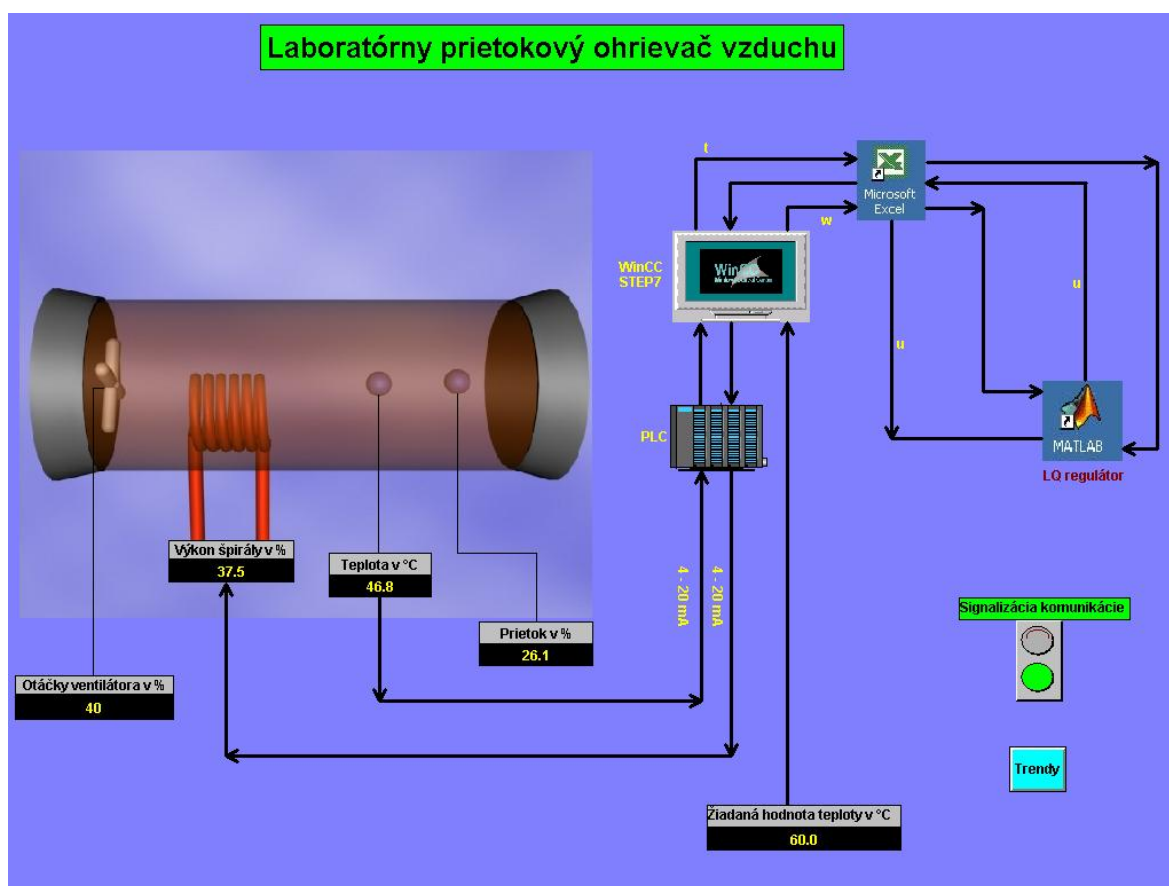
Obr. 15 Otvorené okno modulu *Global Script* s napísaným programom

Spustenie programu sa vykonáva vo vizualizačnej obrazovke pomocou tzv. *C-Action*, ktorá vykonáva programy vložené do nej.

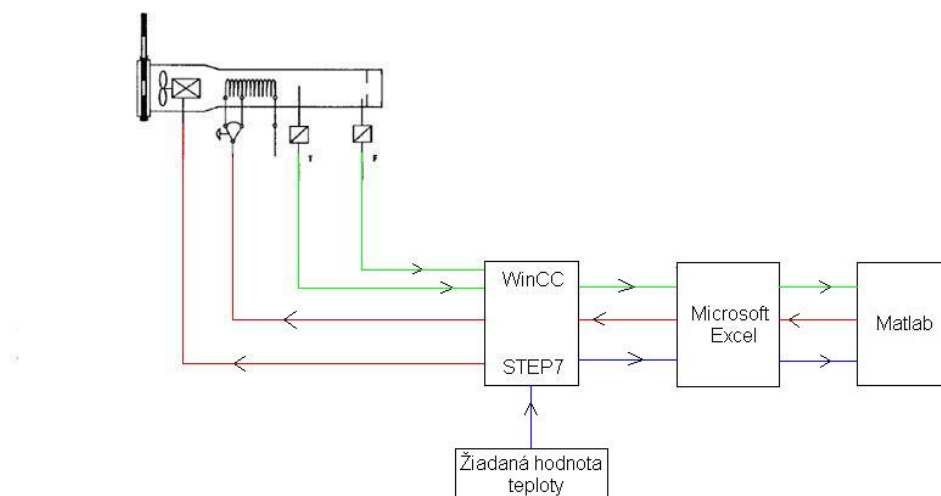


## 6.2 Vytvorenie vizualizačnej obrazovky

Pre laboratórny prietokový ohrievač vzduchu boli vytvorené dve vizualizačné obrazovky. Prvá zobrazuje model prietokového ohrievača s ovládacími prvkami, na druhej sú zobrazené trendy jednotlivých veličín v čase. Obrázok prietokového ohrievača vzduchu bol vytvorený v programe 3D Studio Max R3, ktorý bol vložený na pozadie obrazovky. V pravej časti bol tiež umiestnil indikátor kopírovania hodnôt medzi tagmi. Pokiaľ bola zobrazená zelená farba, *project functions* s názvom „New\_Functions\_2.fct” napísaný v module *Global Script* je spustený. To, ako často sa majú kopírovať hodnoty medzi tagmi, bolo nastavené pri definovaní *C-Action*, pomocou *trigger*-a na 250 ms čo znamená, že každých 250 ms sa spustí program a prekopírujú sa hodnoty medzi tagmi. Ak je kopírovanie úspešné *C-Action* vráti „1“ a indikátor svieti na zeleno, ak je neúspešné, vráti hodnotu „0“ a indikátor sa rozsvieti na zelenú.



Obr. 16 Vizualizačná obrazovka pre ohrievač vzduchu a ovládacími prvkami a znázornením tokov informácií

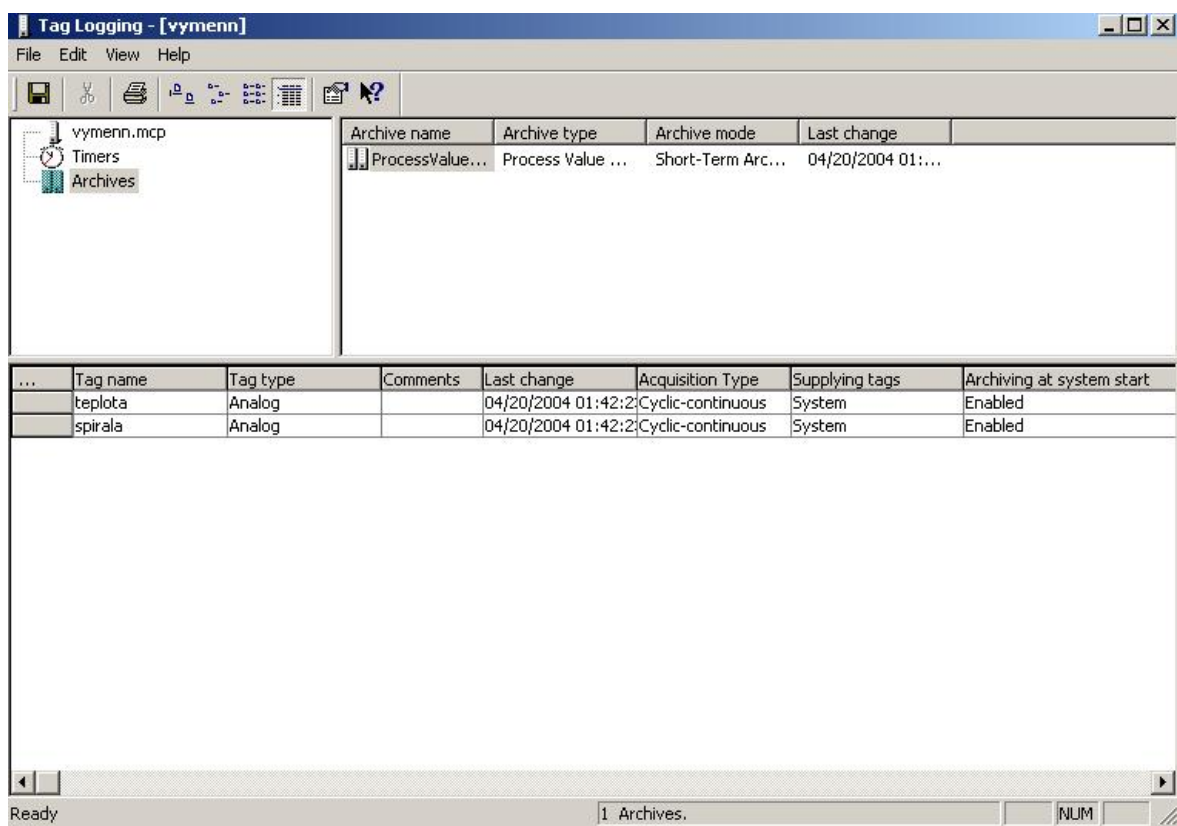


Obr. 17 Schematický náčrt ohrievača vzduchu so znázornením tokom informácií

K jednotlivým zobrazeným zariadeniam boli pripojené vstupno – výstupné polia, ktoré sa vyznačujú tým, že môžu ovplyvňovať alebo byť ovplyvňované vznikajúcimi udalosťami. Pod udalosťou sa rozumie akákoľvek zmena, či už zmena hodnoty tagu (napr. zmena teploty) alebo zásah užívateľa.

### 6.3 Tvorba archívov

Na prácu s archívami slúži modul *Tag logging*. Vytvorený archív je uložený na pevnom disku (súbor s príponou DBF). Keďže bolo potrebné archivovať hodnoty teploty vzduchu a výkonu špirály, boli pridané do nového archívu tieto dva tagy (obr. 18). Interval ukladania hodnôt bol 500 ms.



Obr. 18 Archív pre teplotu vzduchu a výkon špirály

## 7 Komunikácia Matlab – Microsoft Excel

Microsoft Excel je aj v prípade komunikácie s WinCC aj s Matlabom server, pretože v oboch prípadoch odpovedá na ich požiadavky. Komunikácia medzi Matlabom a Excelom cez DDE rozhranie je realizovaná cez príkazy uvedené v kapitole 3.1.2. Vlastný skript, ktorý začína aj končí komunikáciu sa skladá z týchto hlavných častí:

- 1) Otvorenie komunikácie s Microsoft Excel cez DDE rozhranie
- 2) Načítanie hodnôt z konkrétnych buniek do premenných
- 3) Operácie s týmito premennými
- 4) Poslanie nových hodnôt do programu Microsoft Excel a ukončenie komunikácie so serverom

Na pravidelné načítavanie a uskutočnenie operácií bolo potrebné vytvoriť *timer* pomocou príkazov uvedených v kapitole 3.2.

## 8 Návrh LQ regulátora pre riadenie ohrievača vzduchu

Cieľom riadenia je zabezpečiť žiadanú hodnotu teplotu vzduchu pomocou riadiacej veličiny výkonu špirály pri konštantných otáčkach ventilátora. Najprv však bolo potrebné systém zidentifikovať. Otáčky ventilátora boli nastavené na konštantných 40 %. Pre identifikáciu bolo potrebné namerať prechodové charakteristiky pri rôznych zmenách výkonu špirály (viď tab. 7).

Výkon špirály [%] ( $t < 0$ s )	50	50	50	50	50	50
Výkon špirály [%] ( $t \geq 0$ s )	60	70	80	40	30	20

Tab. 7 Skokové zmeny výkonu špirály

Na základe viacerých prechodových charakteristík, ktoré sa získali pri rôznych skokových zmenách výkonu špirály sa vytvorila priemerovaná prechodová charakteristika [3] podľa vzorca

$$\hat{y}_i = \frac{\sum_{k=1}^{N=6} \Delta u_k \cdot y_{ik}}{\sum_{k=1}^{N=6} (\Delta u_k)^2} \quad (5)$$

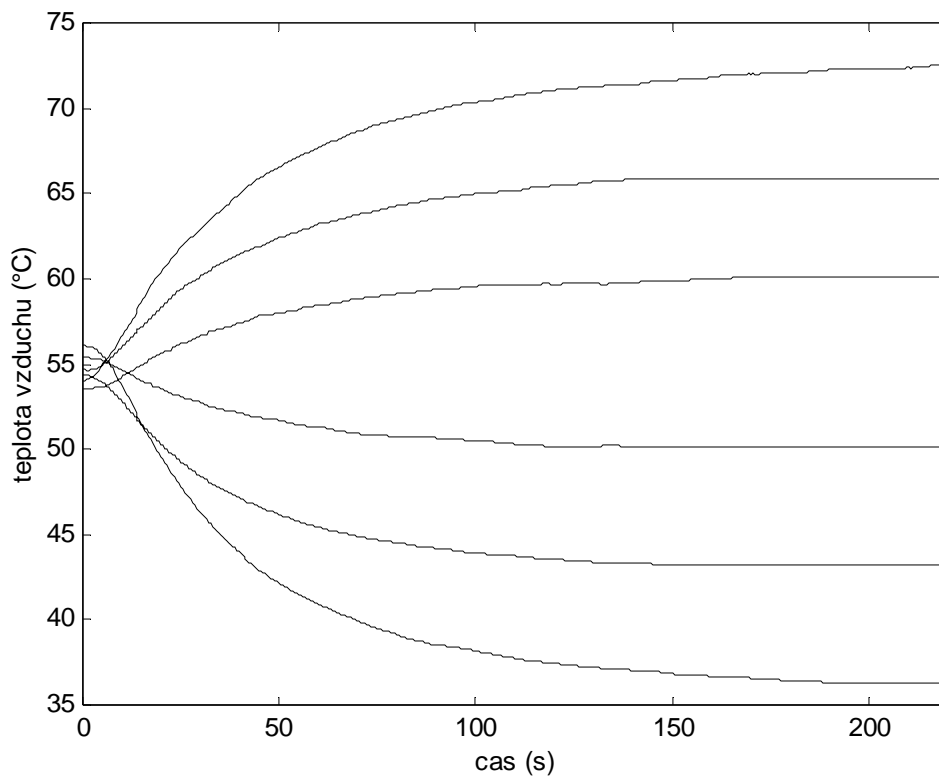
kde  $i$  je  $i$ -ty bod prechodovej charakteristiky

$k$  je  $k$ -te meranie

$\Delta u_k$  je skoková zmena vstupu pri  $k$ -tom meraní

$y_{ik}$  je hodnota výstupu pri  $k$ -tom meraní v  $i$ -tom intervale

$\hat{y}_i$  je výsledná hodnota prech. charakteristiky v čase  $t = i\Delta t$ , kde  $\Delta t$  je perióda vzorkovania



Obr. 19 Prechodové charakteristiky systému

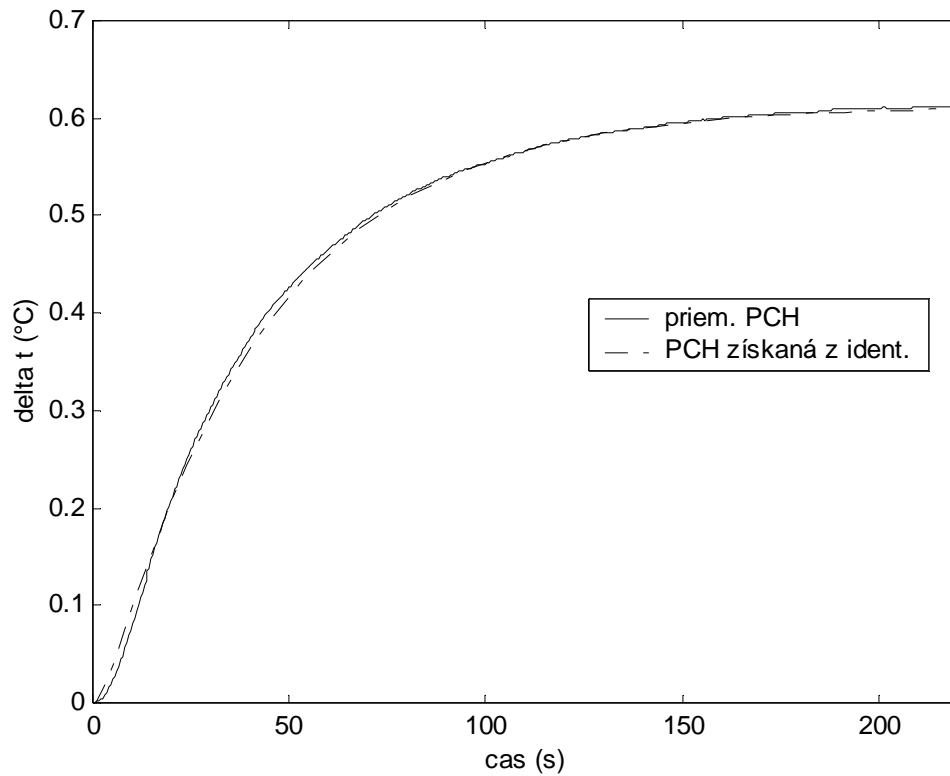
Priemerovaná prechodová charakteristika bola identifikovaná ako aperiodický systém 2. rádu s rozdielnymi časovými konštantami nasledovným postupom

- 1) Zosilnenie systému je hodnota PCH v nekonečne  $K = y(\infty)$ .
- 2) Zistia sa hodnoty konštánt  $T_u, T_n$  a vypočíta sa ich podiel  $T_n / T_u = f_1(k)$ .
- 3) Pomocou tabuľky so závislosťami  $f_i(k)$  sa odčíta pre dané  $f_1(k)$  hodnota  $k$ .
- 4) Pre dané  $k$  sa zistí hodnota  $f_2(k)$  a odhad časovej konštanty  $T_1 = T_n / f_2(k)$ .
- 5) Hodnota časovej konštanty  $T_2$  sa vypočíta podľa vzťahu  $T_2 = k.T_1$ .

Identifikovaný prenos je potom

$$G(s) = \frac{b_0}{a_2 s^2 + a_1 s + a_0} = \frac{0,6125}{120s^2 + 44s + 1} \quad (6)$$

Ako vidieť na obr.20 medzi priemerovanou prechodovou charakteristikou a prechodovou charakteristikou získanou z identifikácie prenosovej funkcie nie je skoro žiadna odchýlka, takže identifikovaný prenos je správny.



Obr. 20 Porovnanie priemerovanej prechod. charakteristiky a prechod. charakteristiky získanej z identifikácie prenosovej funkcie

## 8.1 Polynomický LQ regulátor s integračnou činnosťou

Parametre regulátora sú dané diofantickou rovnicou

$$AsP + BQ = M \quad (7)$$

kde  $M = N.G$  (8)

$N$  je stabilná časť polynómu rovnice spektrálnej faktorizácie

$$N.N^* = a.a^* \quad (9)$$

a  $G$  je stabilná časť polynómu rovnice spektrálnej faktorizácie [4]

$$h = G.G^* = \varphi.a^*.f^*.a.f + \psi.b^*.b \quad (10)$$

kde  $a = 120s^2 + 44s + 1$

$$b = 0,6125$$

$$a^* = 120.(-s)^2 + 44.(-s) + 1$$

$$b^* = 0,6125$$

$$f = s$$

$$f^* = -s$$

$$\varphi = 0,02 \text{ (váhový koeficient)}$$

$$\psi = 1 \text{ (váhový koeficient)}$$

Po dosadení všetkých hodnôt je  $h$

$$h = -290s^6 + 34s^4 - 0,02s^2 + 0,38$$

Polynóm  $G$  sa vypočíta pomocou *polynomického toolboxu* v Matlabe príkazom *spf* s výsledkom

$$G = 17s^3 + 13s^2 + 4s + 0,61$$

Polynóm  $N$  je po dosadení a po urobení spektrálnej faktorizácie v Matlabe

$$N = 120s^2 + 44s + 1$$

Po dosadení polynómov  $G, N$  do rovnice (8) nadobudne  $M$  hodnotu

$$M = 2040s^5 + 2308s^4 + 1069s^3 + 262,2s^2 + 30,84s + 0,61 \quad (11)$$

Polynomický LQ regulátor je v tvare



$$G_R(s) = \frac{Q(s)}{P(s)} = \frac{q_2 s^2 + q_1 s + q_0}{p_2 s^2 + p_1 s + p_0} \quad (12)$$

Po dosadení polynómu  $M$  do rovnice (7) dostaneme sústavu rovníc:

$$\begin{aligned} a_2 p_2 &= M_5 \\ a_2 p_1 + a_1 p_2 &= M_4 \\ a_2 p_0 + a_1 p_1 + a_0 p_2 &= M_3 \\ a_1 p_0 + a_0 p_1 + b_0 q_2 &= M_2 \\ a_0 p_0 + b_0 q_1 &= M_1 \\ b_0 q_0 &= M_0 \end{aligned} \quad (13)$$

Po dosadení parametrov z (6) a z (11) dostaneme spojitý LQ regulátor v tvare

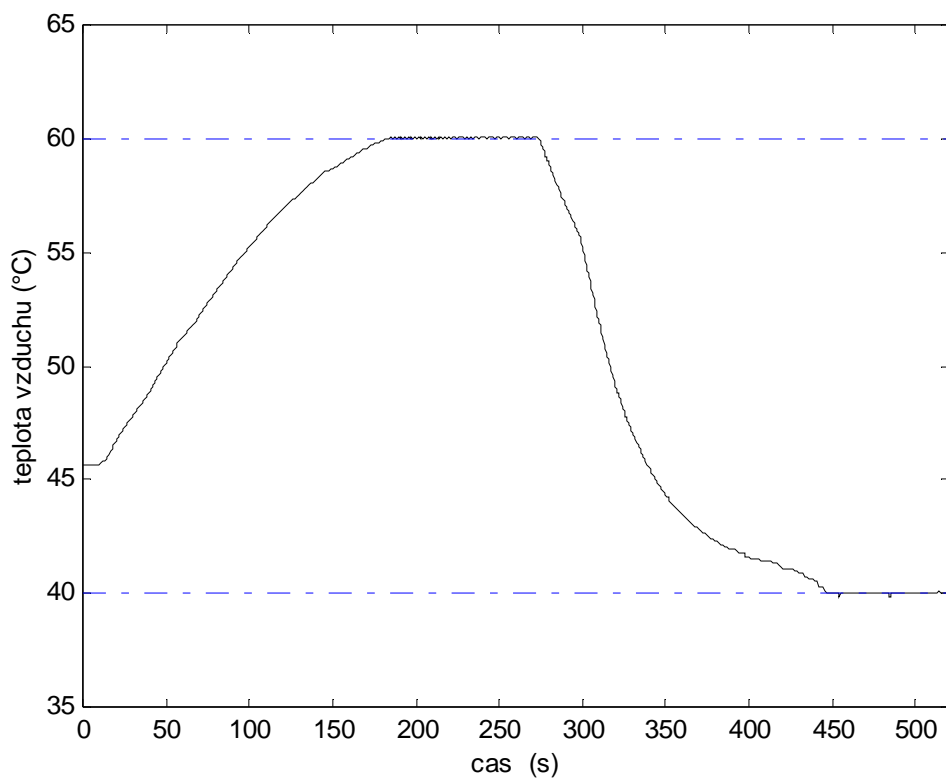
$$G_R(s) = \frac{110s^2 + 39s + 0,89}{85s^2 + 45s + 7,1}$$

Keďže údaje z procesu sú odoberané v určitých periódach, je potrebné systém riadiť diskretným LQ regulátorom, získaným po prevode spojitého LQ regulátora pri perióde vzorkovania 0,1 s pomocou Matlabu príkazom *c2d* v tvare

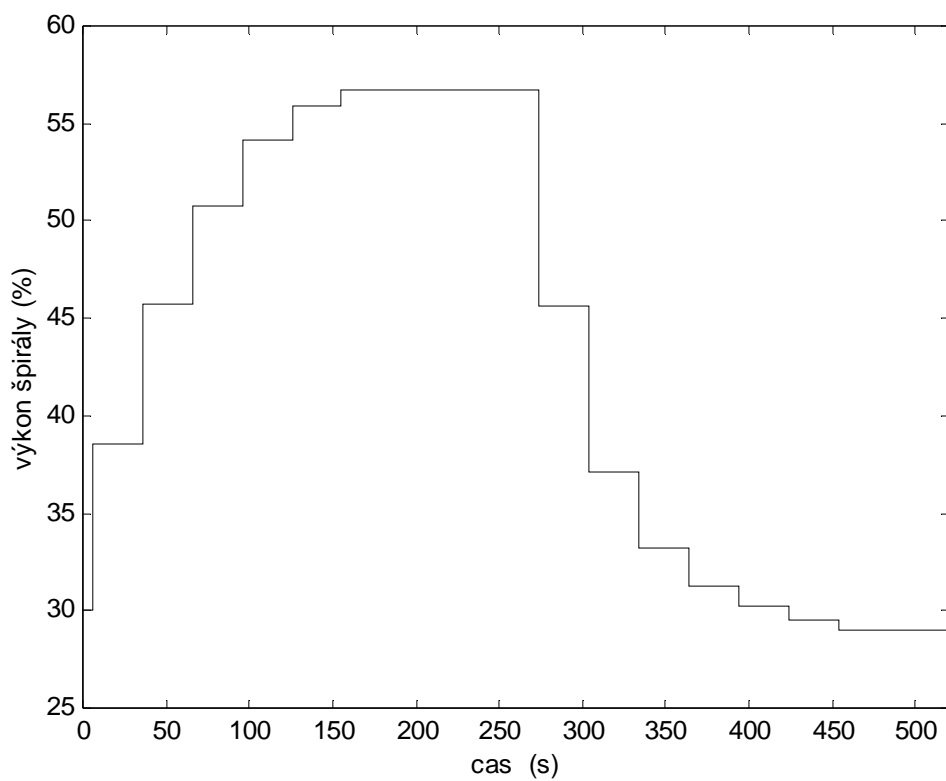
$$G_R(z) = \frac{1,294z^2 - 2,543z + 1,249}{z^2 - 1,948z + 0,9484}$$

## 8.2 Riadenie ohrievača vzduchu navrhnutým regulátorom

Na začiatku merania boli otáčky ventilátora nastavené na 40 %. Počas celého merania boli konštantné. Výkon špirály bol nastavený na 30 %. Keď sa teplota vzduchu dostala do ustáleného stavu, žiadaná hodnota teploty sa nastavila na 60 °C a spustil *timer*. Nastavenie timera, skript, ktorý sa ním spúšťa ako aj simulačnú schému na výpočet akčného zásahu regulátorom je uvedený v prílohe B. Po dosiahnutí žiadanej hodnoty bola zmenená na 40 °C. Po dosiahnutí tejto žiadanej hodnoty sa timer zastavil.



Obr. 21 Riadenie teploty na žiadanú hodnotu polynomickým LQ regulátorom so znázornením žiadaných hodnôt teploty vzduchu.



Obr. 22 Časový priebeh výkonu špirály ako akčnej veličiny

	D4	▼	<i>fx</i>
	A	B	C
1	<b>Otáčky ventilátora (%)</b>	40	
2	<b>Prietok vzduchu (%)</b>	25.6	
3	<b>Výkon špirály (%)</b>	55.9	
4	<b>Teplota vzduchu (°C)</b>	58.7	
5	<b>Žiadaná hodnota teploty (°C)</b>	60	
6			
7			
8			
9			

Obr. 23 Otvorené okno programu Microsoft Excel počas riadenia

## Záver

Úlohou tejto práce bolo prepojenie Matlabu so Simatic S7 cez Microsoft Excel pomocou DDE rozhrania a aplikácia tejto komunikácie na riadenie laboratórneho prietokového ohrievača vzduchu. Prvým krokom bolo vytvorenie projektu v programe STEP7 a pripojenie prietokového ohrievača k pracovnej stanici. Po otestovaní manuálneho ovládania priamo v programe STEP7 pomocou tabuľky premenných, sa pristúpilo k tvorbe vizualizačného prostredia v programe WinCC. Ďalej bolo vytvorené spojenie Matlabu s programom Microsoft Excel cez DDE rozhranie pomocou príkazov Matlabu na nadviazanie a uskutočnenie spojenia. Po nameraní prechodových charakteristík, vytvorení priemerovanej prechodovej charakteristiky a jej identifikácii ako systému 2. rádu bol navrhnutý LQ regulátor s integračnou činnosťou pre riadenie teploty vzduchu pomocou výkonu špirály pri konštantných otáčkach ventilátora. Nakoniec bolo uskutočnené meranie, pri ktorom sa riadila teplota vzduchu na žiadanú hodnotu pomocou výkonu špirály. Hodnota teploty vzduchu bola z programu WinCC posielaná do programu Microsoft Excel cez DDE rozhranie a odtiaľ ich cez DDE načítaval Matlab, ktorý vypočítaval pravidelne hodnoty akčnej veličiny výkonu špirály a posielal ich späť do programu Microsoft Excel, odkiaľ ich načítal program WinCC a posielal ich do ohrievača vzduchu.

Výsledkom tejto práce je možnosť prepojenia pracovnej stanice Simatic S7 a programov Microsoft Excel a Matlab na báze DDE rozhrania a riadenie systému externým programom s návrhom LQ regulátora s integračnou činnosťou. Všetky body zadania diplomovej práce boli splnené.

V budúcnosti by bolo potrebné navrhnuť a otestovať viacero typov regulátorov, vhodných pre jednotlivé typy riadení ako aj možnosť prepojenia pracovnej stanice Simatic S7 a programu Matlab bez použitia programu Microsoft Excel.

## Literatúra

- [1] [www.mathworks.com](http://www.mathworks.com)
- [2] [www4.ad.siemens.de](http://www4.ad.siemens.de)
- [3] Fikar, M., Mikleš, J.: Identifikácia systémov, STU, Bratislava, 1999
- [4] Dostál, P., Meszáros, A., Mikleš, J., (1994). A modified LQ tracking problem, Journal of EE, 45, 4, 129 - 133
- [5] Amira GmbH; LTR700; Laboratory setup Airstream and Temperature Control Plant; Amira GmbH; Technologiezentrum Duisburg 2000
- [6] SIMATIC, User Manual, Standart Software for S7-300
- [7] Kožka, Š., Kvasnica, M.,: Programovanie PLC SIMATIC 300 ( Základná príručka ), KIRP, Bratislava 2001
- [8] SIMATIC, Manual, Programming with STEP7 V5.2
- [9] SIMATIC, Manual, Configuring Hardware and Communications Connection STEP7 V5.2
- [10] [www.siemens.com](http://www.siemens.com)

# Príloha A

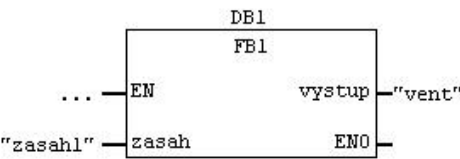
## Výpis riadiaceho programu zo STEP7

OB1 : "Main Program Sweep (Cycle)"

Comment:

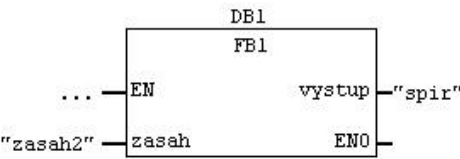
Network 1: ventilator

Comment:



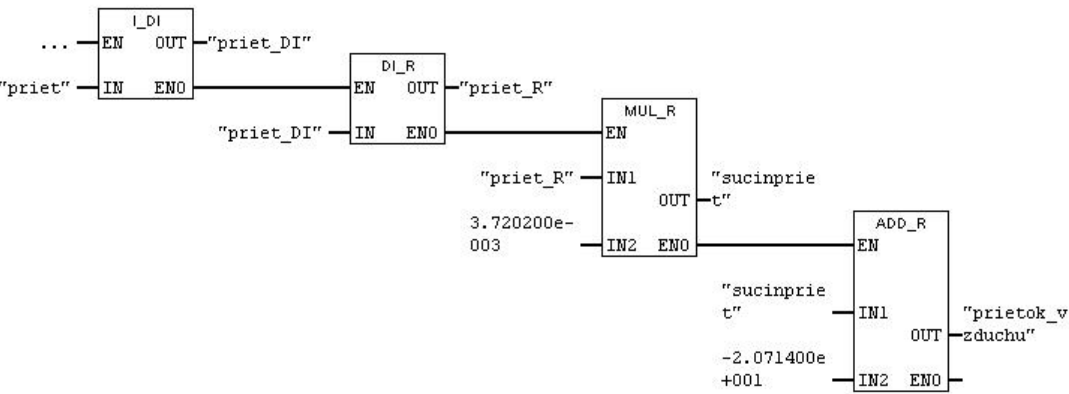
Network 2: spirala

Comment:



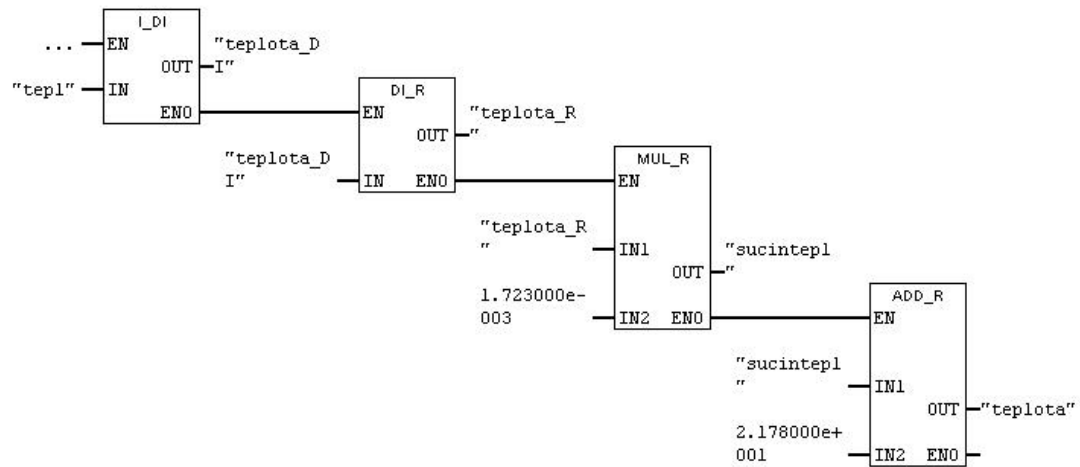
Network 3: prietok vyduchu

Comment:



#### Network 4 : teplota

Comment:

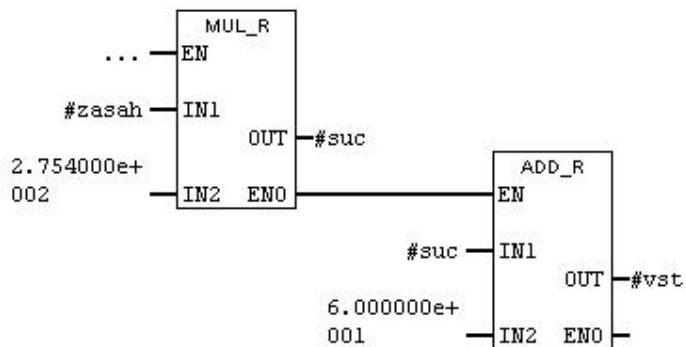


FB1 : Title:

Comment:

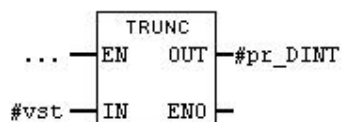
#### Network 1 : kalibrácia vstupných veličín $y=a*x+b$

Comment:



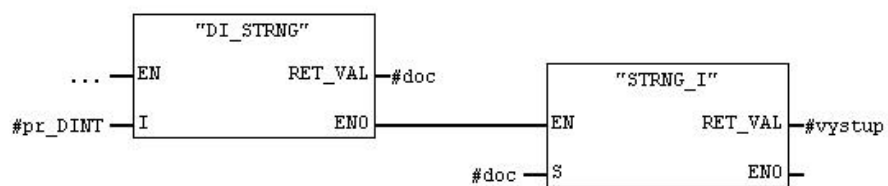
#### Network 2 : prevod reálneho čísla na DINT

Comment:



**Network 3 : Prevod DINT na INT**


Comment:





## Príloha B

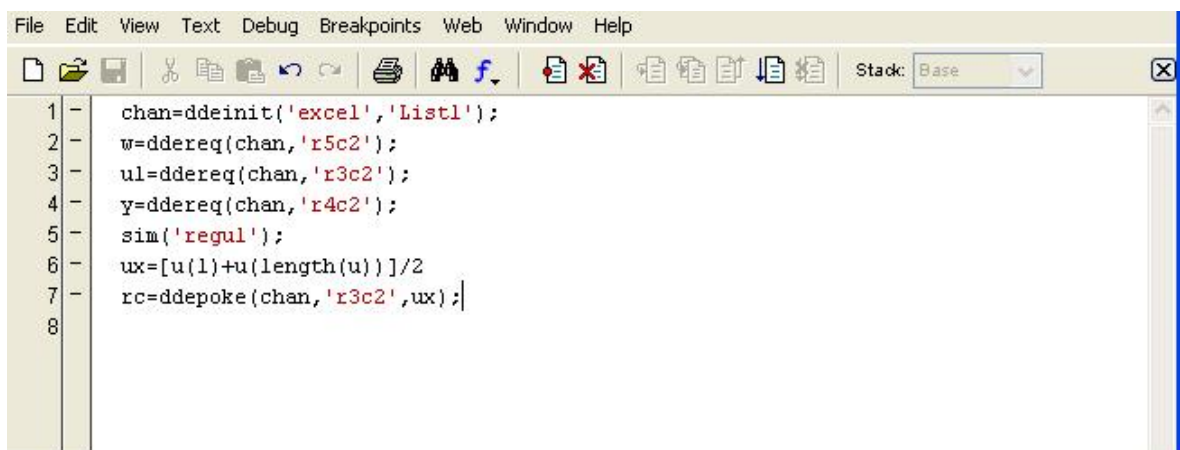
### Definovanie nového timera



A screenshot of the MATLAB editor window. The menu bar includes File, Edit, View, Text, Debug, Breakpoints, Web, Window, and Help. The toolbar contains icons for file operations, editing, and debugging. The stack on the right shows 'Base'. The editor contains the following code:

```
1 - t=timer('TimerFcn','komun','Period',30);
2 - set(t,'ExecutionMode','FixedRate');
3 - start(t)
```

### Výpis skriptu *komun.m* použitého na komunikáciu s Microsoft Excel cez DDE



A screenshot of the MATLAB editor window showing the script *komun.m*. The menu bar and toolbar are the same as in the previous image. The stack on the right shows 'Base'. The editor contains the following code:

```
1 - chan=ddeinit('excel','List1');
2 - w=ddereq(chan,'r5c2');
3 - ul=ddereq(chan,'r3c2');
4 - y=ddereq(chan,'r4c2');
5 - sim('regul');
6 - ux=[u(1)+u(length(u))]/2
7 - rc=ddepoke(chan,'r3c2',ux);
8
```

Simulačná schéma na výpočet akčnej veličiny pomocou regulátora použitá v *komun.m*

