

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

**IMPLEMENTÁCIA RIADENIA NA PRIEMYSELNOM
RIADIACOM SYSTÉME B&R X20**

DIPLOMOVÁ PRÁCA

FCHPT-5414-40286

2011

Bc. Katarína Chroustová

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

**IMPLEMENTÁCIA RIADENIA NA PRIEMYSELNOM
RIADIACOM SYSTÉME B&R X20**

DIPLOMOVÁ PRÁCA

FCHPT-5414-40286

Študijný program: I-AICHP Automatizácia a informatizácia v chémii a potravinárstve

Číslo a názov študijného programu: 5.2.14 Automatizácia

Školiace pracovisko: OIRP ÚIAM FCHPT

Vedúci diplomovej práce: Ing. Michal Kvasnica, PhD.

Konzultant diplomovej práce: Ing. Richard Valo

BRATISLAVA 2011

Bc. Katarína Chroustová



ZADANIE DIPLOMOVEJ PRÁCE

Študentka: **Bc. Katarína Chroustová**

ID študenta: 40286

Študijný program: automatizácia a informatizácia v chémii a potravinárstve

Študijný odbor: 5.2.14 automatizácia

Vedúci práce: Ing. Michal Kvasnica, PhD.

Konzultant: Ing. Richard Valo

Názov práce: **Implementácia riadenia na priemyselnom riadiacom systéme B&R X20**

Špecifikácia zadania:

Cieľom práce je implementácia riadiacich algoritmov na priemyselnom riadiacom systéme B&R X20. Pre tepelno-materiálovú sústavu sú jednotlivé úlohy formulované nasledovne:

- * identifikácia procesu z prechodovej charakteristiky pri meniacich sa otáčkach čerpadla
- * syntéza P a PI regulátorov pre riadenie teploty
- * implementácia prepočtov fyzikálnych jednotiek v jazyku C
- * vytvorenie vizualizačného modulu pre riadenie a zber dát

Rozsah práce: 50

Riešenie zadania práce od: 14. 02. 2011

Dátum odovzdania práce: 21. 05. 2011



Bc. Katarína Chroustová
Študentka

prof. Ing. Miroslav Fikar, DrSc.
Vedúci pracoviska

prof. Ing. Miroslav Fikar, DrSc.
Garant študijného programu

Pod'akovanie

Ďakujem vedúcemu mojej diplomovej práce, Ing. Michalovi Kvasnicovi, PhD., a konzultantovi, Ing. Richardovi Valovi, za odborné usmernenie, trpezlivosť a cenné rady, ktoré mi počas vypracovania tejto práce poskytli.

Čestné vyhlásenie

Týmto podpisom potvrdzujem, že som túto prácu vypracovala samostatne s použitím prameňov, ktoré uvádzam v zozname literatúry a že som ich použila spôsobom vo vede obvyklým.

Bratislava, 20. 05. 2011

.....

Bc. Katarína Chroustová

Abstrakt

Chroustová, Katarína: Implementácia riadenia na priemyselnom riadiacom systéme B&R X20. Diplomová práca. Slovenská Technická Univerzita, Fakulta chemickej a potravinárskej technológie. Vedúci diplomovej práce: Ing. Michal Kvasnica, PhD. Konzultant diplomovej práce: Ing. Richard Valo. Bratislava 2011. 78 s.

Hlavným cieľom diplomovej práce je navrhnúť riadenie laboratórneho zariadenia PCT23 v softvérovom prostredí B&R Automation Studio a vytvorené riadenie implementovať na reálny proces s využitím priemyselného riadiaceho systému B&R X20. Prvá kapitola je venovaná teoretickému oboznámeniu so softvérom a zariadením. Dôraz sa kladie na prácu s programom B&R Automation Studio, na prepojenie programu s prostredím Simulink a opis súčastí riadeného zariadenia. Poznatky uvedené v prvej časti práce sa využívajú v praktickej úlohe, ktorej riešenie je obsahom ďalšej kapitoly. Zadanie je rozdelené na čiastkové úlohy, riešením ktorých získame konečný výsledok. Výstupom z vypracovania diplomovej práce je projekt v Automation Studiu, ktorý bol reálne implementovaný prostredníctvom riadiaceho systému X20. Riadenie laboratórneho zariadenia PCT23 sa vykonávalo z vytvoreného vizualizačného prostredia.

Kľúčové slová: B&R Automation Studio, Simulink, zariadenie PCT23, akčné členy, senzory, regulátory, GUI.

Abstract

Chroustová, Katarína: Implementation of Control Algorithms on the X20 Industrial Control Platform. Diploma thesis. Slovak Technical University, Faculty of Chemical and Food Technology. Supervisor of the diploma thesis: Ing. Michal Kvasnica, PhD. Consultant of diploma thesis: Ing. Richard Valo. Bratislava 2011. 78 pp.

The purpose of diploma thesis is to design control algorithms for process plant trainer PCT 23 in B&R Automation Studio environment and to implement of created control on real process using X20 industrial control platform. The first chapter is devoted to the theoretical background on the software and the plant. The focus is put on the work with the B&R Automation Studio, on the interface between the software and the Simulink environment and the description of the plant's components. The theoretical information stated in the first chapters of the thesis is then utilized in the practical exercise solved in the subsequent chapters. The problem is divided into partial tasks, elaboration of which determines the final solution. The final result of the thesis is a project in the Automation Studio which can be implemented on the X20 control platform. Control of the laboratory plant was performed from the created visualization environment.

Key words: B&R Automation Studio, Simulink, Process plant trainer PCT 23, control members ,sensors, controllers, GUI

Obsah

Zoznam obrázkov	6
Zoznam tabuliek	8
Zoznam symbolov a skratiek	9
Úvod	12
1 Teoretická časť	13
1.1 B&R Automation Studio	13
1.1.1 Konfigurácia hardvéru a softvéru	14
1.1.2 Programovacie jazyky	15
1.1.3 Vizualizačný systém	19
1.2 Prepojenie B&R Automation Studio so Simulinkom	20
1.2.1 Funkčné bloky pre B&R Automation Studio Toolbox	21
1.3 Laboratórne zariadenie PCT23 (Process Plant Trainer)	26
1.3.1 Opis zariadenia	27
1.3.2 Riadiaca konzola	30
1.3.3 Akčné členy a senzory	31
2 Praktická časť	34
2.1 Riadenie laboratórneho zariadenia	34
2.1.1 Zapojenie zariadenie použité pri riešení úlohy	36
2.1.2 Prepojenie PLC s riadiacou konzolou a Automation Studiom	37
2.1.3 Implementácia prepočtov fyzikálnych jednotiek	38
2.1.4 Návrh riadenia pre 1. riadiacu slučku	42
2.1.5 Návrh riadenia pre 2. riadiacu slučku	47
2.1.6 Vytvorenie vizualizačného prostredia	63
Záver	68
Zoznam použitej literatúry	69
Prílohy	70

Zoznam obrázkov

Obr. 1 Stromová štruktúra projektu	14
Obr. 2 Ukážka programovacieho jazyka Automation Basic.....	16
Obr. 3 Ukážka jazyka ANSI C.....	17
Obr. 4 Ukážka jazyka Ladder diagram	18
Obr. 5 Automatické generovanie kódu	20
Obr. 6 B&R Automation studio Toolbox	21
Obr. 7 B&R Config blok.....	22
Obr. 8 B&R Input blok	22
Obr. 9 B&R Extended Input blok	25
Obr. 10 Laboratórne zariadenie PCT23	26
Obr. 11 Princíp činnosti peristaltického čerpadla	27
Obr. 12 Trojstupňový výmenník tepla [8]	28
Obr. 13 Zadržiavacia trubica	28
Obr. 14 Zásobník teplej vody	29
Obr. 15 Riadiaca konzola.....	31
Obr. 16 Schéma riadenia procesu	34
Obr. 17 Schéma zapojenia zariadenia PCT23 využitá pri riešení úlohy	36
Obr. 18 Ukážka časti kódu pre obmedzenie hodnôt signálov.....	38
Obr. 19 Ukážka trace editora	43
Obr. 20 Prechodová charakteristika systému.....	44
Obr. 21 Porovnanie PCH pôvodného a identifikovaného systému	45
Obr. 22 Riadenie teploty ohrevného média na vstupe do výmenníka P regulátorom	46
Obr. 23 Akčný zásah regulátora	47
Obr. 24 Skokové zmeny otáčok čerpadla N1 a príslušné PCH	48
Obr. 25 Prvá skoková zmena a PCH.....	49
Obr. 26 Druhá skoková zmena a PCH.....	49
Obr. 27 Tretia skoková zmena a PCH	50
Obr. 28 Normované prechodové charakteristiky	51
Obr. 29 Stredná hodnota prechodovej charakteristiky systému	52
Obr. 30 Porovnanie pôvodnej a identifikovanej prechodovej charakteristiky.....	53
Obr. 31 Schéma riadenia v Simulinku	57
Obr. 32 Riadenie teploty produktu bez filtra	58
Obr. 33 Akčný zásah regulátora pri riadení bez filtra.....	58

Obr. 34 Simulačná schéma s filtrom 1.rádu.....	59
Obr. 35 Riadenie teploty produktu s filtrovaním signálov	60
Obr. 36 Regulačná odchýlka.....	60
Obr. 37 Akčný zásah regulátora	61
Obr. 38 Hlavná obrazovka vizualizácie	63
Obr. 39 Obrazovka pre ponuku grafov	64
Obr. 40 Graf teploty produktu	65
Obr. 41 Obrazovka pre nastavenie parametrov P regulátora	66
Obr. 42 Obrazovka schémy zapojenia zariadenia PCT23	67

Zoznam tabuliek

Tab. 1 Dátové typy Input bloku	23
Tab. 2 Akčné členy zariadenia PCT23	32
Tab. 3 Teplotné senzory zariadenia PCT23	32
Tab. 4 Senzory výšok hladín, vodivosti a prietoku zariadenia PCT23	33
Tab. 5 Ukazovatele kvality riadenia	62

Zoznam symbolov a skratiek

AS	A utomation S tudio
CPU	C entral P rocessing U nit
GUI	G raphical U ser I nterface
IEC	I nternational E lectrotechnical C ommission
IL	I nstruction L ist
LD	L adder D iagram
P	P roporcionálny
PCH	P rechodová C harakteristika
PID	P roporcionálno – I ntegračno - D erivačný
PLC	P rogrammable L ogic C ontroller
SOL1	S olenoidný ventil 1
SOL2	S olenoidný ventil 2
SOL3	S olenoidný ventil 3
SOL4	S olenoidný ventil 4
SOL5	S olenoidný ventil 5
SFC	S equential F unction C hart
SPT	S et P oin T
ST	S tructured T ext
TRO	T rvalá R egulačná O dchýlka
URO	U zavretý R egulačný O bvod
USB	U niversal S erial B us
WYSIWYG	W hat Y ou S ee I s W hat Y ou G et
XGA	e Xtended G raphics A rray
XML	e Xtensible M arkup L anguage

Úvod

Moderná spoločnosť so sebou prináša neustály pokrok. Snažíme sa veci čo najviac zdokonaľovať a vylepšovať. Zavedením riadenia procesov môžeme vykonávať činnosti rýchlejšie, jednoduchšie a bezpečnejšie. Správne navrhnuté riadenie zvyšuje efektivitu, šetrí čas a finančné prostriedky.

V priemysle zohráva automatizácia veľmi dôležitú úlohu. Úspešnosť každého výrobného podniku je odrazom efektivity riadenia výroby. Plynulú prevádzku, požadujúce podmienky procesov a optimálny výsledný produkt dosiahneme zavádzaním uzavretých regulačných obvodov. Parametre regulátorov navrhujeme tak, aby sme zabezpečili stabilitu regulačného obvodu, minimalizovali vplyv porúch a aby riadená veličina sledovala žiadanú hodnotu čo najrýchlejšie a najpresnejšie.

Pri vypracovaní diplomovej práce sme využili softvér B&R Automation Studio a jeho prepojenie s programom Simulink, v ktorých sme navrhli riadenie laboratórneho zariadenia PCT 23. Toto zariadenie bolo vyrobené pre účely vzdelávania a simuluje správanie reálneho procesu. V prostredí Simulink bol vytvorený uzavretý regulačný obvod na riadenie teploty produktu, ostatné fázy vytvárania riadiaceho systému boli realizované v B&R Automation Studiu. Samotné riadenie sa vykonáva v priemyselnom riadiacom systéme B&R X20. Charakteristiky použitých programov a zariadenia sú obsahom prvej kapitoly.

Účelom diplomovej práce je spojenie nadobudnutých teoretických poznatkov s praktickým riešením komplexnej úlohy. Zadanie je rozdelené do viacerých častí, ktoré opisujú kroky pri riešení úlohy. Fázy na seba logicky nadväzujú, postupuje sa od identifikácie systémov, cez návrh regulátorov až po vytvorenie vizualizačného prostredia.

Práca je pokračovaním diplomového projektu, v ktorom sme sa oboznámili s programom B&R Automation Studio na jednoduchom príklade. Bol zameraný na detailne opísanie vytvárania projektu v tomto prostredí a slúžil ako návod pre nových užívateľov. Táto publikácia rieši zložitejší problém riadenia na reálnom zariadení a je určená pre tých, ktorí už majú skúsenosti s programovaním v B&R Automation Studiu.

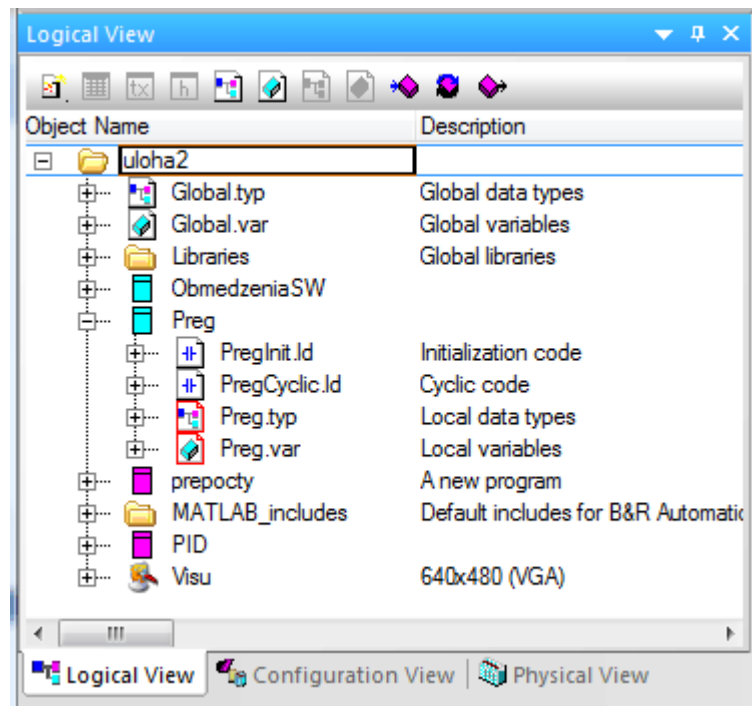
1 Teoretická časť

Táto kapitola práce je zameraná na teoretické poznatky a vedomosti. Prvé dve podkapitoly obsahujú charakteristiku programovacieho balíka B&R Automation Studio a jeho prepojenia s prostredím Matlab. Naznačujú základnú filozofiu práce s programom, pričom definujú jednotlivé súčasti a nástroje nachádzajúce sa v spomínanom balíku. Riadenie, vytvorené v Automation Studio, bolo implementované na laboratórnom zariadení PCT 23, ktorého opis tvorí obsah poslednej podkapitoly.

1.1 B&R Automation Studio

B&R Automation Studio(AS) je integrované softvérové vývojové prostredie poskytujúce nástroje pre viaceré fázy vytvárania projektu. V jednom balíku môžeme vytvoriť regulátory, prepojenia a vizualizácie, čo redukuje čas potrebný na integráciu projektu a náklady spojené s bežnou údržbou. Umožňuje zariadeniam jednoduchú a efektívnu konfiguráciu a programovanie s využitím užívateľského grafického prostredia v MS Windows. Jednou z množstva výhod programu je programovanie s využitím medzinárodne akceptovaných štandardov jazykov IEC1131 [1], [2].

Systémovo orientovaný prehľad projektu a rozdelenie do funkčných blokov poskytuje jednoduché riadenie a programovanie aj rozsiahlejších projektov. Jednotlivé sekcie projektu ako sú knižnice, premenné a samotné programy sú usporiadané v stromovej štruktúre. Tým sa vytváranie riadiaceho systému stáva prehľadným a znižuje sa riziko, že sa v programe stratíme. Deklarácie premenných, pridávanie knižníc či vkladanie častí programu sú veľmi rýchle a prístupné. AS ponúka možnosť extrahovať celý projekt alebo časti projektu a importovať ich do iného projektu. Tiež je možné uložiť zdrojový kód vo formáte XML, s výnimkou kódov písaných jazykmi IEC a ANSI C. [3]



Obr. 1 Stromová štruktúra projektu

1.1.1 Konfigurácia hardvéru a softvéru

Dôležitou charakteristikou vývojového systému je schopnosť kedykoľvek vytvoriť prepojenie medzi projektom a fyzickým systémom. B&R AS ponúka túto funkciu v rámci zobrazení projektu, kde je vzťah medzi hardvérovou a softvérovou konfiguráciou zobrazený graficky. Hardvérová konfigurácia sa nachádza na ľavej strane hlavného okna, kde je cieľový systém zobrazený graficky stromovou štruktúrou (záložka „*Physical View*“). Konfigurácia môže byť vytvorená používateľom bez pripojenia PLC (offline) alebo automaticky pri aktívnom pripojení PLC (online). Moduly sa zoradujú podľa typu (CPU, digital input, digital output, atď.) [1].

Softvérová konfigurácia graficky zobrazuje prepojenie s hardvérovou časťou (záložka „*Configuration View*“). Dôležitý vzťah medzi softvérom a hardvérom je priame priradenie symbolických premenných k fyzickej adrese. Program nepracuje s fyzickou adresou zariadenia, ale používa len názov symbolickej premennej. Pri zmene zapojenia sa tak nevyžadujú modifikácie programu.

Softvérové projekty pozostávajú z jedného alebo viacerých programových objektov, ktoré sú umiestnené v rôznych triedach. Triedy sú skupiny v programe, ktoré sú spracovávané s rovnakou dobou cyklu (napr. 10 ms) [1].

1.1.2 Programovacie jazyky

B&R Automation Studio poskytuje jednoduché a kompletne programovanie. Pri vytváraní projektu je možná akákoľvek kombinácia jazykov. Knižnice môžu byť taktiež vytvorené vo všetkých podporovaných jazykoch. Program umožňuje rozdelenie premenných na globálne - využívané v celom projekte, a lokálne - využívané len v programe, v ktorom sú definované. Balík obsahuje štandardné IEC knižnice, ktoré sú rozšírené o B&R knižnice. Všetky jazyky môžu pristupovať k rovnakým dátovým typom a využívať rovnaké knižnice a premenné [4].

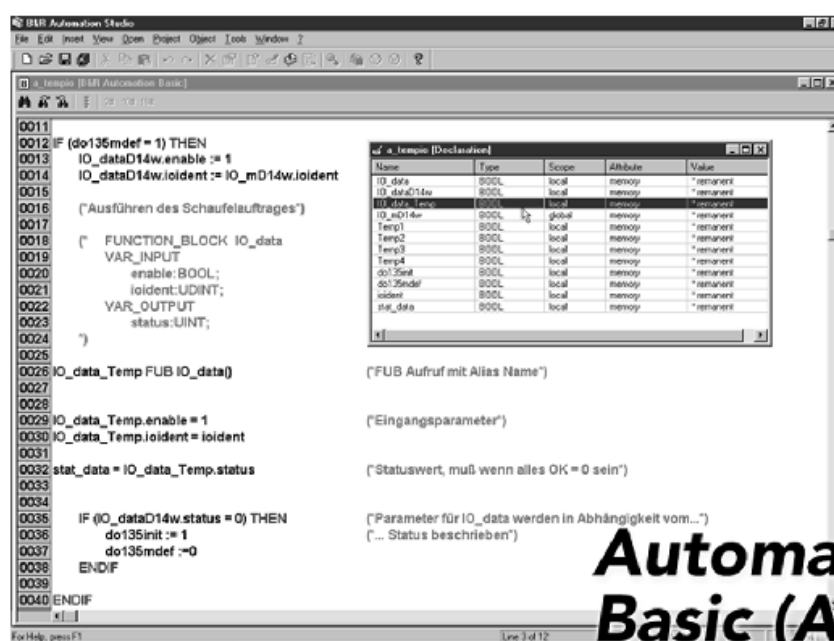
Pre jednoduchšie vytváranie a analýzu zdrojového kódu je syntax farebne rozlíšená pre všetky programovacie jazyky. Premenné sa môžu deklarovať počas programovania alebo v dialógových oknách. Ak sa v programe vyskytne chyba zobrazí sa okno s chybovým hlásením, dvojitým kliknutím na toto hlásenie sa otvorí príslušný programový editor, v ktorom je chybný riadok kódu zvýraznený. Pre podrobnejšie vysvetlenie častí programu môže užívateľ vkladať medzi riadky komentáre. Nízke rozlíšenie obrazovky je možné kompenzovať rôznym nastavením zoomu alebo použitím rôznych formátov písma [1].

Na vytváranie programov v Automation Studio môžu byť použité nasledujúce jazyky:

- B&R Automation Basic
- ANSI C
- IEC1131 programovacie jazyky:
 - IEC61131- 3 Ladder Diagram (LD)
 - IEC61131- 3 Instruction List (IL)
 - IEC61131- 3 Structured Text (ST)
 - IEC61131- 3 Sequential Function Chart (SFC)

▪ B&R Automation Basic

B&R Automation Basic je textový jazyk založený na inštrukciách vyvinutý spoločnosťou B&R. Okrem štandardných prvkov programovacích jazykov obsahuje jazykové elementy, ktoré sú prispôsobené pre B&R regulátory. Tento jazyk bol vyvinutý z programovacieho jazyka PL2000, je vhodný pre aktualizáciu projektov vytvorených v PG2000 (predchodca B&R AS) a rovnako aj na vytváranie nových projektov v oblasti automatizácie [1].

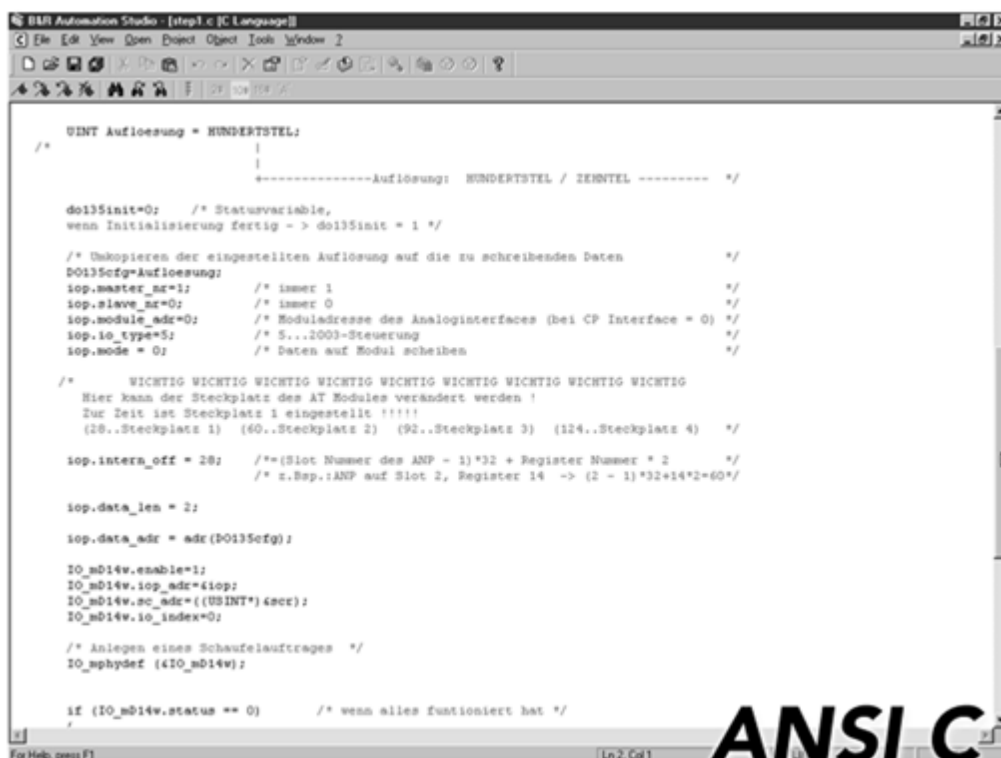


Automation Basic (AB)

Obr. 2 Ukážka programovacieho jazyka Automation Basic

▪ ANSI C

Jazyk ANSI C bol vyvinutý v polovici 70-tych rokov z jazyka B (Basic Combined Programming Language). V dnešnej dobe je to veľmi dobre známy štandardný strojovo orientovaný jazyk. Aj keď nie je štandard pre všetkých výrobcov PLC, B&R plne integruje tento jazyk v B&R Automation Studio. ANSI C jazyk dovoľuje volanie funkčných blokov a funkcií, väčšinou sa používa v programoch, kde sú vykonávané výpočty, sú vytvorené algoritmy alebo je použitý C kód [1], [4].



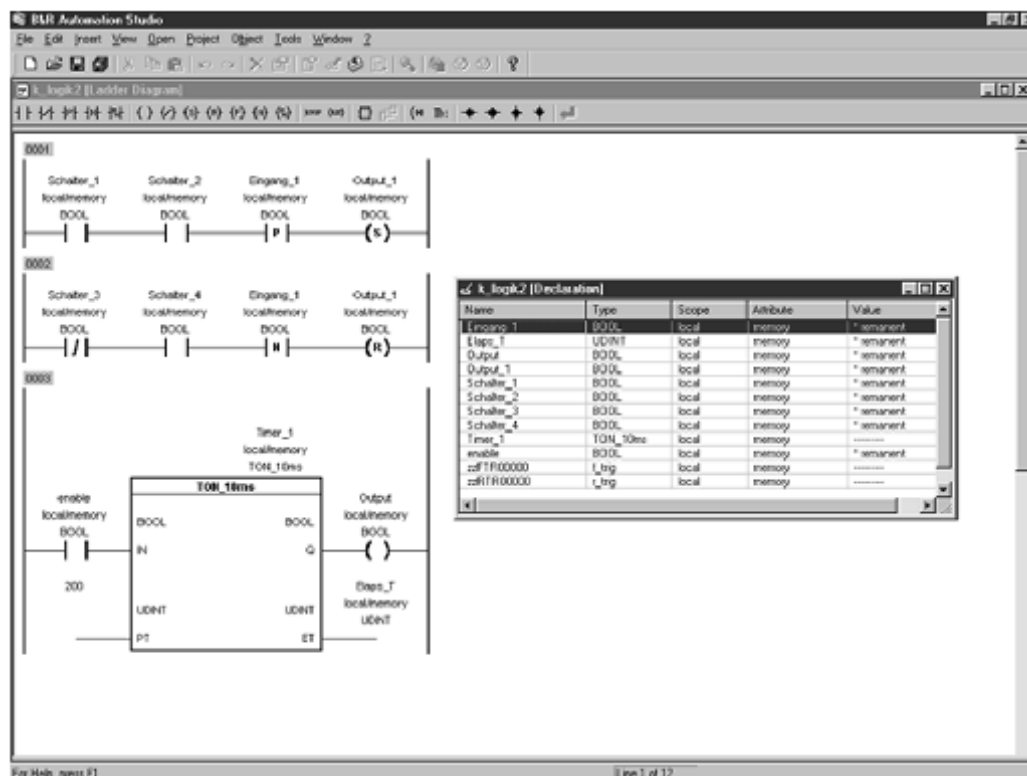
Obr. 3 Ukážka jazyka ANSI C

1.1.2.1 IEC1131 programovacie jazyky

IEC1131 norma sumarizuje požiadavky pre hardvér PLC a programovanie ich systémov. Cieľom je znížiť náklady na školenia, a rovnako aj na vytváranie a implementáciu jednotlivých metód. IEC 1131-3 norma poskytuje návod pre programovanie PLC. Bola vyvinutá medzinárodnou normalizačnou skupinou IEC (International Electrotechnical Commission) [1].

○ *Ladder Diagram(LD)*

Rebríkový diagram je metóda, ktorá je odvodená z elektromechanických relé systémov a definuje tok prúdu cez jednotlivé siete. Z tohto dôvodu rebríkové diagramy vyzerajú ako elektrické schémy a väčšinou sú obmedzené na spracovanie logických signálov (1=pravda, 0=nepravda). Grafický programovací jazyk využíva na programovanie symboly ako sú spínacie a rozpínacie prepínače a cievky, ktoré sú spájané do sietí a vytvárajú logickú štruktúru [1], [4].



Obr. 4 Ukážka jazyka Ladder diagram

- **Inststruction list (IL)**

IL je druhou generáciou programovacieho jazyka založeného na inštrukciách strojového kódu. Väčšinou je používaný na vytváranie krátkych programových sekcií s logickými funkciami (AND, OR, atď.). Tento jazyk možno prirovnať k programovaniu v assembleri [1].

- **Structured Text (ST)**

Štruktúrovaný text je programovací jazyk nasledujúci Pascal, ktorý nepoužíva inštrukcie strojového kódu ako napríklad jazyk IL. Namiesto toho používa abstraktné príkazy na vytvorenie štruktúry príkazov. Štruktúrovaný text sa používa vo všetkých oblastiach automatizačnej techniky [1].

- ***Sequential Function Chart (SFC)***

Sekvenčný funkčný diagram bol vytvorený na rozdelenie zložitých úloh do jasných krokov a na definovanie ovládania tokov medzi týmito krokmi. SFC bol vyvinutý z krokového sekvenčného programovania a je vhodný najmä pre procesy so zmenou stavu v krokoch, ako napríklad práčka (predumytie, umývanie, atď.) [1].

1.1.3 Vizualizačný systém

Vizualizačným systémom AS môžeme vytvárať riadkové displeje, integrované ovládanie a diaľkové XGA displeje s tlačidlami alebo dotykovou obrazovkou. Integráciou vizualizačného systému s ovládacími prvkami sa značne skracujú komunikačné časy, ktoré sú bežné pre vzdialené vizualizačné systémy [5].

Zabudovanými vizuálnymi komponentmi môžeme vytvárať procesné WYSIWYG diagramy. WYSIWYG je princíp verného prenosu vizuálnej informácie, resp. informácie modelovanej na počítači do reality tak, že zodpovedá presne modelovanému obrazu s čo najmenším, resp. nebadateľným skreslením [5], [6].

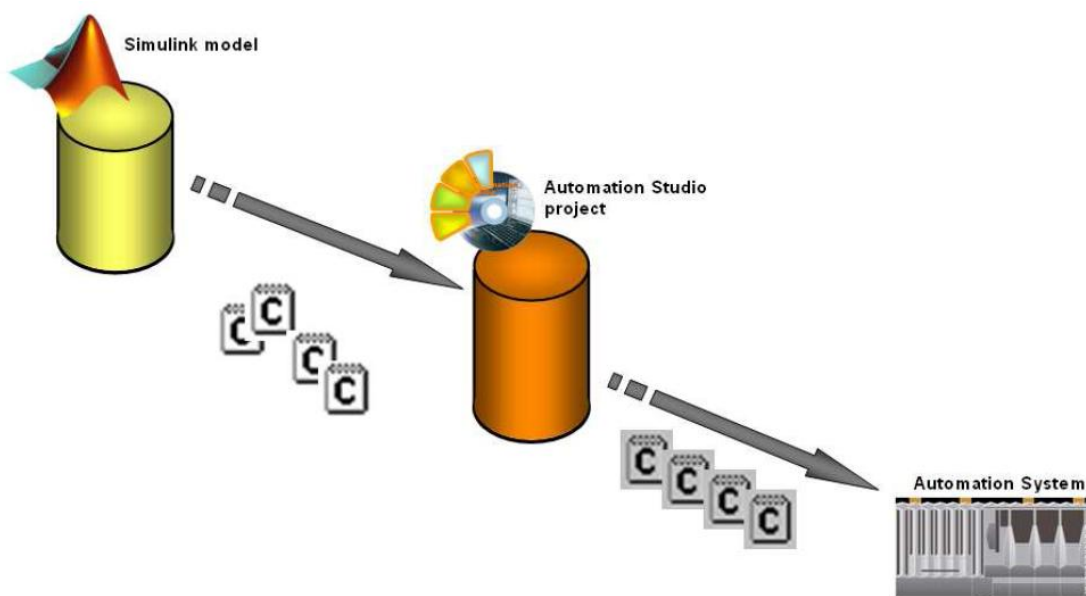
Grafické užívateľské prostredie (GUI) uľahčuje riadenie zariadení pomocou tlačidiel, klávesov a klúčov, ktorým sa pridávajú rôzne akcie. Priradením objektu k premennej zabezpečíme interakciu medzi vizualizáciou a reálnym zariadením. Prostredníctvom vzdialených displejov môže byť riadenie realizované mimo priestoru zariadenia, čo je výhodou najmä pri väčších zariadeniach. Obrazovky môžu zobrazovať aj hodnoty veličín, ktoré na zariadení nemajú svoj displej alebo zobrazovať grafický priebeh veličín prostredníctvom grafov údajov. Hodnoty premenných môžeme z vizualizácie aj zadávať, napr. môžeme meniť parametre regulátorov, žiadané hodnoty riadených veličín alebo môžeme zastaviť celý proces riadenia.

B&R Automation Studio pri vytváraní GUI ponúka veľké množstvo grafických prvkov, ako sú priamky, obdĺžniky, rôzne obrázkové mapy, vrátane ISO symbolov, a množstvo iných. Text je možné zadávať vo viacerých jazykoch. Užívateľ vizualizácie si môže sám zvoliť jazyk, v ktorom mu bude vizualizácia zobrazovaná.

1.2 Prepojenie B&R Automation Studio so Simulinkom

Matlab slúži ako účinný nástroj pri riešení technických, matematických a ekonomických problémov. Najväčšia výhoda tohto programu spočíva v manipulácii s veľkými maticami, čo vyplýva už z jeho názvu MATrix LABoratory. Simulink, ktorý Matlab obsahuje, umožňuje tvorbu grafických simulačných modelov na riadenie technologických procesov v reálnych podmienkach [7].

Automatické zavádzanie Simulinkových modelov v C-kóde, špeciálne optimalizovanom pre použitie v B&R, ponúka nové možnosti pre vývoj a projektovanie simulačných modelov, ktoré by inak nebolo možné alebo by bolo časovo náročné realizovať. Základný princíp je jednoduchý, modul vytvorený v prostredí Simulink je automaticky preložený pomocou Real-Time Workshop a Real-Time Workshop Embedded Coder do optimálneho jazyka pre B&R systém [7].

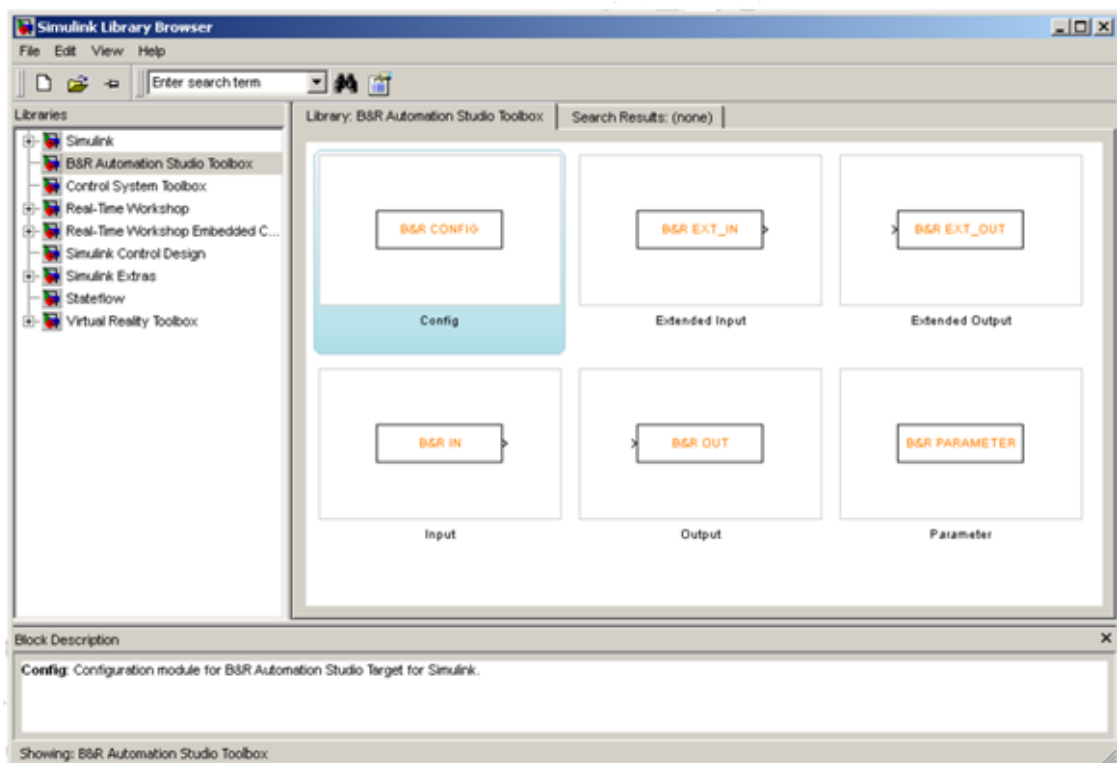


Obr. 5 Automatické generovanie kódu

Automatické generovanie kódu poskytuje jednoduchý prenos komplexných a prepracovaných modelov v Simulinku do projektu v Automation Studiu. Navrhnuté regulátory možno ľahko skúšať a optimalizovať bez prispôbovania veľkého množstva kódu a vytvárania chýb v kóde [7].

1.2.1 Funkčné bloky pre B&R Automation Studio Toolbox

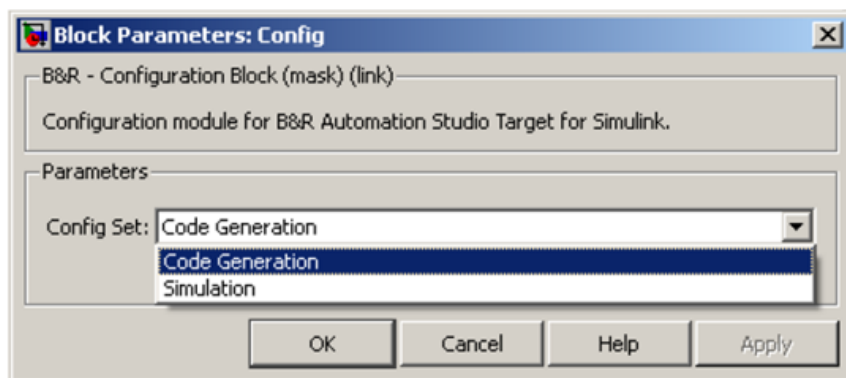
B&R Automation Studio Toolbox je automaticky nainštalovaný v priebehu nastavenia B&R Automation Target for Simulink. Obsahuje niekoľko funkčných blokov, ktoré sú opísané v tejto časti práce [7].



Obr. 6 B&R Automation studio Toolbox

1.2.1.1 B&R Config blok

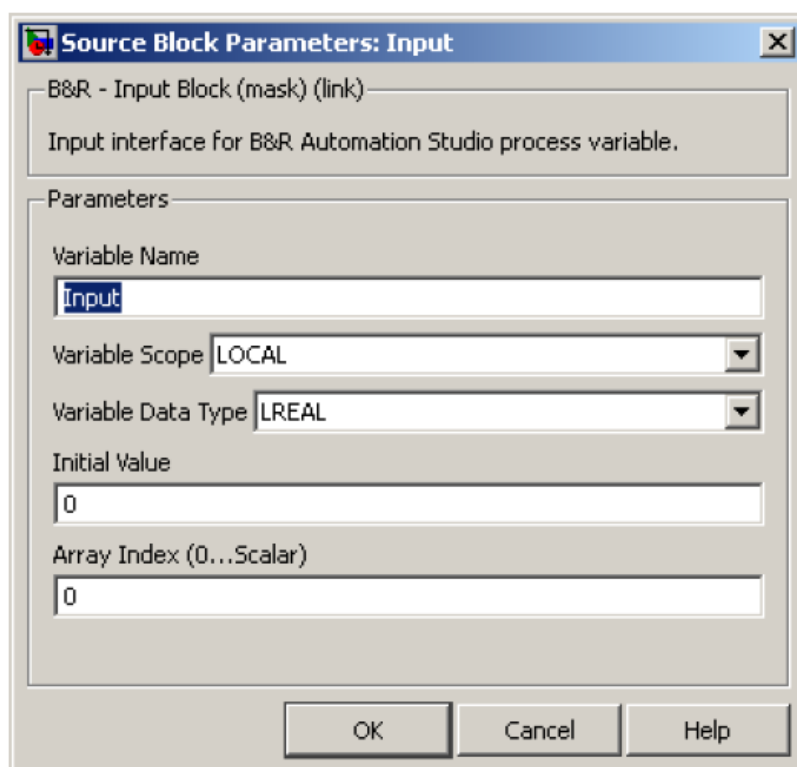
B&R Config blok sa používa na prepínanie medzi dvoma módmi operácie, medzi „*Simulation*“ a „*Code generation*“. Blok sa pridá do Simulinkovej schémy a z ponuky bloku sa vyberie mód simulácie alebo generácie kódu. Do schémy môže byť pridaný iba jeden B&R Config blok. Výberom konfigurácie „*Code Generation*“ systém automaticky vyvolá súbor „*bur.tlc*“ pre príslušný Simulinkový model [7].



Obr. 7 B&R Config blok

1.2.1.2 B&R Input blok

B&R Input blok slúži ako rozhranie medzi časťou vytvoreného projektu a Simulinkovým modelom. Pre každý Input blok je vytvorená premenná v cieľovom systéme. Hodnoty týchto premenných je možné zadávať z projektu a slúžia ako vstupy do Simulinkového modelu [7].



Obr. 8 B&R Input blok

„*Variable name*“ predstavuje názov premennej v Automation Studio. V položke „*Variable Scope*“ definujeme typ premennej, môže byť lokálna alebo globálna. Lokálne premenné sú v projekte automaticky vytvorené. Globálne premenné je potrebné zadať manuálne, ako pomôcka slúži vytvorený textový dokument v Automation Studio s názvom global.txt. V položke „*Variable Data Type*“ zvolíme dátový typ premennej. Dostupné dátové typy sú uvedené v tabuľke 1 (Tab. 1). Tieto dátové typy sú rovnaké pre všetky funkčné bloky [7].

Automation Studio	Simulink	Rozsah hodnôt
BOOL	boolean	FALSE, TRUE
DINT	int32	-2.147.483.648 ... 2.147.483.647
INT	int16	-32 768 ... 32 767
LREAL(default)	double	-1.7E+308 ... 1.7E+308
REAL	single	-3.4E+38 ... 3.4E+38
SINT	int8	-128 ... 127
UDINT	uint32	0 ... 4.294.967.295
UINT	uint16	0 ... 65635
USINT	int8	0 ... 255

Tab. 1 Dátové typy Input bloku

Pri manuálnom definovaní premenných musí užívateľ priradiť premennej rovnaký dátový typ aký je zvolený v Simulinkovom modeli. „*Initial Value*“ predstavuje počiatočnú hodnotu premennej, premenná bude v AS vytvorená s touto hodnotou. Ak premenná predstavuje pole, jeho prvky budú inicializované s rovnakou hodnotou [7].

1.2.1.3 B&R Output blok

B&R Output blok slúži ako rozhranie medzi Simulinkovým modelom a časťou vytvoreného projektu. Pre každý Output blok musí byť vytvorená premenná v cieľovom systéme. Hodnoty týchto premenných sú zobrazované v projekte a predstavujú výstupy

zo Simulinkového modelu. Parametre Output bloku sú rovnaké ako pri Input bloku, takisto môžu byť použité všetky dátové typy z tabuľky 1 [7].

1.2.1.4 B&R Parameter blok

B&R Parameter blok sa používa na vytváranie interných premenných pre jednotlivé bloky. Pre každý Parameter blok musí byť vytvorená premenná v cieľovom systéme. Hodnoty premenných sú zadávané z projektu a nastavujú hodnoty premenných jednotlivým blokom v simulácii. Parametre bloku sú opäť rovnaké ako pri Input bloku [7].

1.2.1.5 B&R Extended Input blok

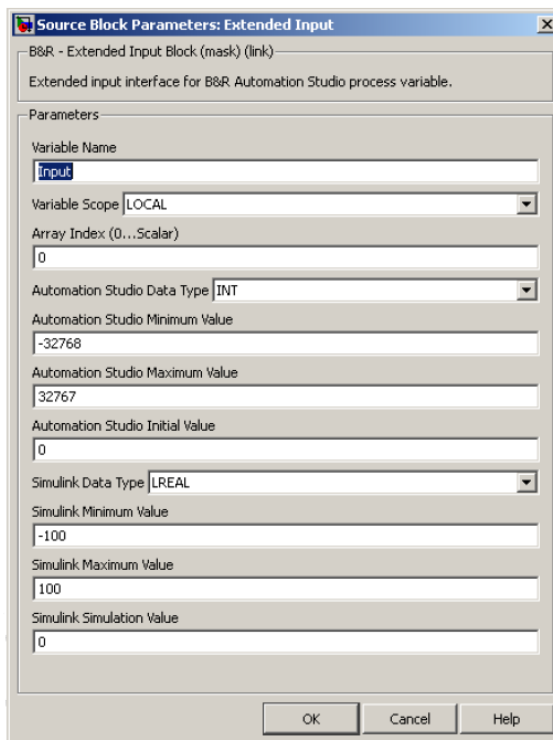
B&R Extended Input blok slúži ako rozhranie medzi Simulinkovým modelom a časťou vytvoreného projektu. Pre každý Extended Input blok je vytvorená premenná v cieľovom systéme. Blok prevádza hardvérové vstupy (zvyčajne z INT) na desatinné čísla (REAL alebo LREAL) vhodné na spracovanie v Automation Studio. Výpočet je vykonávaný v knižnici bloku [7].

Položke „*Variable name*“ predstavuje názov premennej v Automation Studio. „*Variable Scope*“ definuje typ premennej, môže byť lokálna alebo globálna. V položke „*Automation Data Type*“ zvolíme dátový typ premennej v Automation Studio. Dostupné dátové typy sú rovnaké ako pri Input bloku (Tab. 1). Položka „*Automation Studio Minimum Value*“ predstavuje minimálnu hodnotu vstupu v Automation Studio zodpovedajúcu minimálnej hodnote premennej v Simulinku. „*Automation Studio maximum Value*“ predstavuje maximálnu hodnotu vstupu zodpovedajúcu maximálnej hodnote premennej v Simulinku. „*Automation Studio Initial Value*“ je počiatočná hodnota premennej v B&R. „*Simulink data type*“ je dátový typ premennej použitý pri výpočte. „*Simulink Minimum Value*“ predstavuje minimálnu hodnotu vstupu v Simulinku zodpovedajúcu minimálnej hodnote v Automation Studio. „*Simulink Maximum Value*“ predstavuje maximálnu hodnotu vstupu v Simulinku zodpovedajúcu minimálnej hodnote v Automation Studio [7].

Prevod hodnoty premennej z Automation Studio na hodnotu premennej v Simulinku sa uskutočňuje výpočtom podľa vzťahu (1.1):

$$Sl_{value} = (SL_DataType \left[Sl_{Min} + \frac{Sl_{Max} - Sl_{Min}}{AS_{Max} - AS_{Min}} \cdot (AS_{value} - AS_{Min}) \right]) \quad (1.1)$$

[7]



Obr. 9 B&R Extended Input blok

1.2.1.6 B&R Extended Output blok

Blok B&R Extended Output prevádza výstupy z desatinných čísel (REAL alebo LREAL) na hardvérové výstupy (zvyčajne INT). Výpočet je vykonávaný v knižnici bloku. Parametre bloku sú rovnaké ako pri bloku B&R Extended Input (1.2.1.5). Prevod hodnoty premennej zo Simulinku na hodnotu v Automation Studio sa uskutočňuje výpočtom z rovnice (1.2):

$$AS_{value} = AS_DataType \left[AS_{Min} + \frac{AS_{Max} - AS_{Min}}{Sl_{Max} - Sl_{Min}} \cdot (Sl_{value} - Sl_{Min}) \right] \quad (1.2)$$

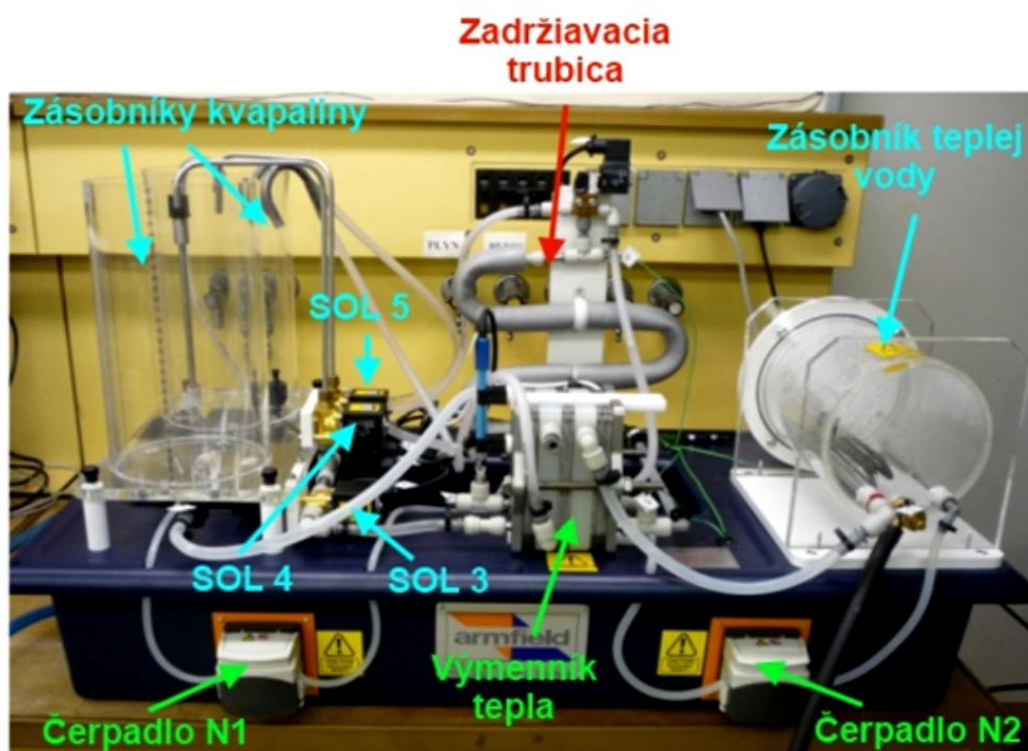
[7]

1.3 Laboratórne zariadenie PCT23 (Process Plant Trainer)

Armfield Process Plant Trainer je miniatúrna verzia reálneho priemyselného procesu s typickými problémami dynamiky a stability, pre ktoré musia byť riadiace systémy navrhnuté a prevádzkované. Zariadenie slúži na účely vzdelávania a umožňuje skúmať rozličné techniky riadenia súvisiace s priemyselnými zariadeniami [8].

Na zariadení sa nachádzajú nasledujúce charakteristiky procesu:

- Mnohorozmerné vstupy a výstupy
- Prítomnosť dopravného oneskorenia
- Recyklácia pre spotrebu minimálnej energie
- Alarmy s nápravnými krokmi [8]



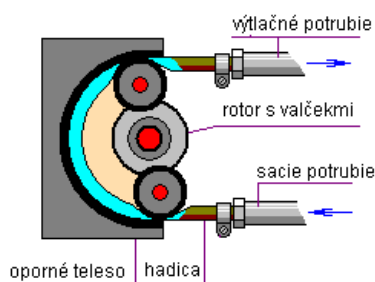
Obr. 10 Laboratórne zariadenie PCT23

1.3.1 Opis zariadenia

Laboratórne zariadenie zobrazené na Obr. 10 pozostáva z viacerých súčastí. Na ľavej strane modulu sú vedľa seba umiestnené dva zásobníky kvapaliny. Tieto nádoby môžeme naplniť kvapalinou pomocou solenoidných ventilov SOL 4 a SOL 5. Zásobníky možno prepojiť manuálnym otočeným ventilu nachádzajúceho sa medzi nimi v spodnej časti. Získame tak systém dvoch zásobníkov zapojených v sérii s interakciou. Zariadenie je pripojené na vodovod silikónovou hadicou s hrúbkou steny 1,6 mm a vnútorným priemerom hadice 8 mm. Hadicou je dodávaná voda požívaná v celom procese [8].

Surovina zo zásobníkov prechádza trojstupňovým výmenníkom tepla nachádzajúcom sa v strede zariadenia PCT23. Pozostáva z troch separátnych ale vnútorne prepojených častí : predhriatie suroviny / regenerácia, ohrievanie a chladenie (Obr. 12) .

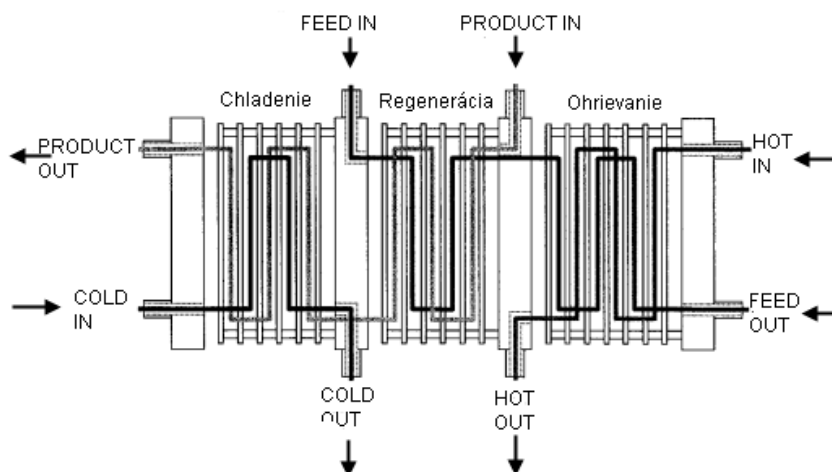
Kvapalinu zo zásobníka do výmenníka privádza peristaltické čerpadlo N1. Princíp činnosti čerpadla je grafický zobrazený na Obr. 11. Valčeky stláčajú hadicu o stenu. Stlačením hadice dochádza vo výtláčnom potrubí k zmene objemu a tekutina je vytlačaná smerom dopredu. Uvoľnením hadice vzniká v sacom priestore podtlak, čím čerpadlo samo nasáva tekutinu. Prietok tekutiny je proporcionálne závislý od otáčok rotora. Hadica je vyrobená zo silikónovej gumeny s hrúbkou steny 1,6 mm a vnútorným priemerom 3,2 mm.



Obr. 11 Princíp činnosti peristaltického čerpadla

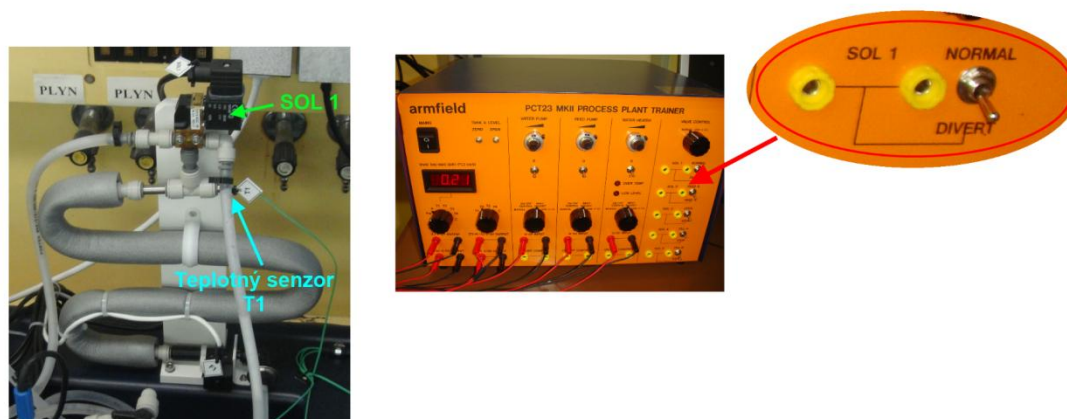
Vstup tekutiny do výmenníka je na Obr. 12 označený ako „*FEED IN*“. Nastavením otáčok čerpadla môžeme riadiť rýchlosť akou neohriata surovina do výmenníka priteká. Teplá voda zabezpečujúca ohrev vo výmenníku je privádzaná zo zásobníka teplej vody. Prietok teplej vody je ovládaný otáčkami čerpadla N2. Hrúbka

steny sacej hadice čerpadla je 1,6 mm a vnútorný priemer 4,8 mm. Horúce médium vstupuje do zariadenia v časti „*HOT IN*“. Vo výmenníku sa prestupom tepla ohreje surovina zo zásobníka na vopred určenú hodnotu. Ohriata surovina opúšťa výmenník otvorom „*FEED OUT*“.



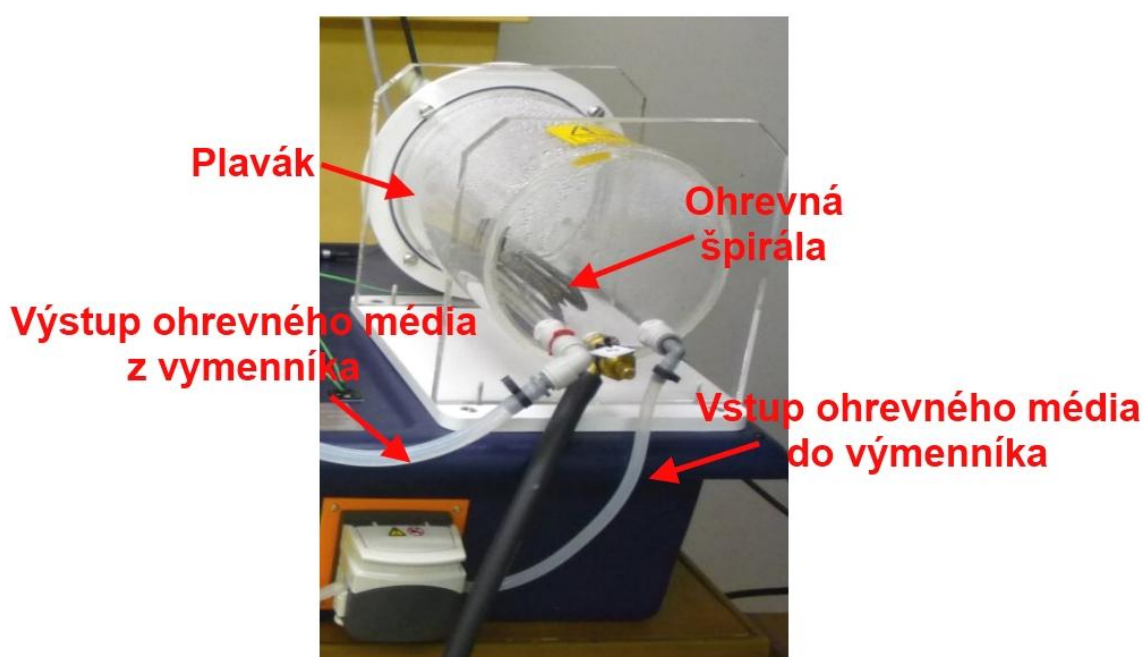
Obr. 12 Trojstupňový výmenník tepla [8]

Teplota vystupujúceho prúdu tekutiny z výmenníka je udržiavaná na žiadanej hodnote po určitú dobu izolačnou zadržiavacou hadicou (obr.13). Ak je SOL 1 prepnutý na polohu „*DIVERT*“, potom zadržiavacu trubicu opúšťa iba tekutina ohriata na požadovanú hodnotu. Zvyšok („*PRODUCT IN*“) sa najprv ochladí prichádzajúcou surovinou do výmenníka (regenerácia) a následne sa zníži jeho teplota použitím externe dodávanej chladiacej vody. Chladiaca voda vchádza do časti výmenníka označenej na Obr. 12 ako „*COLD IN*“ a opúšťa výmenník časťou „*COLD OUT*“ . [8]



Obr. 13 Zadržiavacia trubica

Voda v zásobníku teplej vody (Obr. 14) je ohrievaná elektrickou špirálou, ktorej sa na vstup posiela hodnota príkonu v kilowattoch (kW). Horúce médium vychádzajúce z výmenníka tepla („HOT OUT“) sa recirkuláciou vracia späť do zásobníka teplej vody, čím sa minimalizuje spotreba energie. Zásobník teplej vody je valcová nádoba vybavená plavákom, ktorý kontroluje požadovanú výšku hladiny vody. V prípade nízkej hladiny sa rozsvieti kontrolka na riadiacej konzole, ktorá upozorní na potrebu dodania vody. Voda sa do zásobníka dodáva otvorom v hornej časti nádoby, do ktorého sa pripojí hadica pre studenú vodu. Prietok studenej vody ovláda SOL 3.



Obr. 14 Zásobník teplej vody

Zapojeniu zariadenia zodpovedá napr. spojitá, vysokoteplotná pasterizácia, kde je v skutočnosti procesnou tekutinou mlieko. Ovocné džúsy alebo iné tekuté potraviny vyžadujú tepelné spracovanie ako ochranu pred mikroorganizmami. Veľmi podobné usporiadanie prenosu tepla sa používa aj v iných priemyselných operáciách, kde sa vyžaduje nepriame ohrievanie a chladenie na požadovanú hodnotu. Všetky tieto procesy sú obmedzené na minimálnu spotrebu energie a bezpečnosť pri práci so zariadením [8].

1.3.2 Riadiaca konzola

PCT23 obsahuje elektrickú konzolu, ktorá slúži na prístup rôznych signálov merania a riadenia procesu. Na konzole môžeme využiť rôzne operácie:

- Manuálne riadenie
- Zapisovanie dát s využitím PC alebo záznamu grafov
- Manuálne riadenie prostredníctvom PC
- Priame digitálne riadenie s využitím PC
- Použitie sériovej komunikácie pre individuálne riadenie regulátorov operátorom
- Použitie priemyselných, PID alebo programovateľných regulátorov

Prepojenie konzoly s elektrickou sieťou, zariadením a pripojenie signálov je umiestnené v zadnej časti konzoly. Zariadenie sa pripája prostredníctvom štyroch termočlánkových vstupov. Signály môžu byť pripojené pomocou USB kábla alebo cez I/O port. Konzola sa dostane do činnosti stlačením hlavného vypínača (Obr. 15).

Pod hlavným tlačidlom sa nachádza displej, ktorý zobrazuje hodnotu veličiny zvolenej na prepínači pod displejom. Pod prepínačom sú umiestnené výstupy pre hladinu a prietok, ktorých rozsah sa pohybuje od 0 – 5 V . V tejto časti sú aj výstupy pre snímače teploty. Prepínačom nad výstupom zvolíme, ktorú teplotu posielame na výstup.

Stredná časť konzoly slúži na ovládanie čerpadiel a ohrievača. Tieto akčné členy môžu byť riadené manuálne alebo počítačom. Pri manuálnom riadení nastavíme prepínače do polohy „*MANUAL*“. Potom môžeme otáčaním horných tlačidiel nastaviť hodnotu signálu pre konkrétne čerpadlo alebo ohrievač. Ak chceme zariadenie riadiť počítačom prepínače prepneme do polohy „*INPUT SOCKET*“. Hodnota všetkých vstupných signálov je dovolená v rozmedzí od 0 do 5 V.

V časti pre riadenie ohrievača sa nachádzajú kontrolné svetlá, ktoré signalizujú nízku hladinu v zásobníku teplej vody alebo prekročenie povolenej teploty. Pri rozsvietení kontrolky pre vysokú teplotu teplej vody sa ohrievač automaticky vypne.



Obr. 15 Riadiaca konzola

Posledná časť konzoly obsahuje ovládanie solenoidných ventilov zariadenia. Opäť môžu byť riadené manuálne alebo počítačom. SOL 1 slúži na zvolenie či ohriata voda z výmenníka tepla bude konečný produkt alebo pri hodnote teploty inej ako je požadovaná sa bude vracat' kvapalina späť do výmenníka. SOL 2 rozhoduje z ktorého zásobníka bude surovina dodávaná do výmenníka. SOL 3 slúži na púšťanie chladiacej vody, SOL 4 na napĺňanie zásobníka A a SOL 5 na napĺňanie zásobníka B.

1.3.3 Akčné členy a senzory

Riadenie procesov sa realizuje prostredníctvom akčných členov. Vykonávajú akčný zásah zmenou vstupných signálov na mechanickú prácu. Snímanie veličín na zariadení zabezpečujú senzory. Zariadenie je vybavené senzormi na snímanie teploty, výšky hladiny, prietoku a vodivosti. Nižšie je zobrazený prehľad dostupných akčných členov a senzorov s ich meracími rozsahmi.

▪ Akčné členy

PCT23 má dva druhy akčných členov. Dvojpolohové, ktoré sa prepínajú medzi dvoma polohami, sú buď otvorené alebo zatvorené. Ďalšie akčné členy sú pumpy, ktoré privádzajú vodu do výmenníka tepla. Tekutina môže byť pumpovaná rôznou rýchlosťou v závislosti od otáčok čerpadla. Akčný zásah pri ohreve teplej vody vykonáva špirála v závislosti od príkonu.

Označenie	Rozsah signálu	Akčný zásah
SOL 1	0 alebo 1	Rozdelenie produktu z výmenníka tepla
SOL 2	0 alebo 1	Výber zásobníka na odber suroviny
SOL 3	0 alebo 1	Ovládanie chladiacej vody
SOL 4	0 alebo 1	Napĺňanie zásobníka A
SOL5	0 alebo 1	Napĺňanie zásobníka B
N1 (Feed pump)	0 – 100 %	Čerpanie studenej vody
N2 (Hot water pump)	0 – 100 %	Čerpanie teplej vody
WH(Water Heater)	0 – 2 kW	Ohrev teplej vody

Tab. 2 Akčné členy zariadenia PCT23

▪ Teplotné senzory

Označenie	Snímací rozsah	Snímaná veličina
T1	0 – 150 °C	Teplota na výstupe zo zadržiavacej trubice
T2	0 – 150 °C	Teplota teplej vody
T3	0 – 150 °C	Teplota produktu
T4	0 – 150 °C	Teplota ohriatej kvapaliny na výstupe z výmenníka tepla

Tab. 3 Teplotné senzory zariadenia PCT23

▪ **Senzory výšok hladín, vodivosti a prietoku**

Označenie	Snímací rozsah	Snímaná veličina
C1	0 – 200 mS/ cm	Vodivosť produktu
F1	0 – 500 ml/ min	Prietok produktu na vstupe do zadržiavacej trubice
L1	0 – 250 mm	Výška hladiny v zásobníku A
LL	0 alebo 1	Nízka hladina v zásobníku B
HL	0 alebo 1	Vysoká hladina v zásobníku B

Tab. 4 Senzory výšok hladín, vodivosti a prietoku zariadenia PCT23

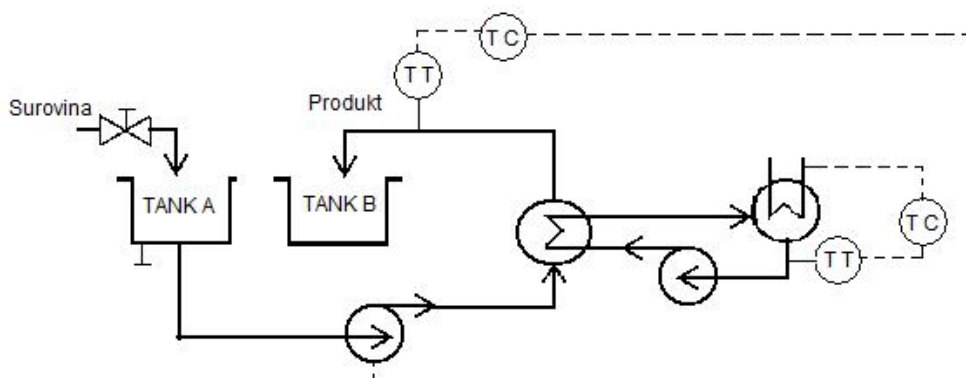
2 Praktická časť

Kapitola obsahuje riešenie praktickej úlohy, ktorá spočíva v riadení laboratórneho zariadenia PCT23 pomocou B&R X20. Riadenie je zabezpečené vytvorením uzavretých regulačných obvodov (URO) s využitím proporcionálneho (ďalej len P) a proporcionálno-itegračno-derivačného (ďalej len PID) regulátora. AS poskytlo prostredie pre programovanie zásad riadenia a prepočet jednotiek. Samotné riadenie sa vykonáva v PLC, do ktorého sa uloží vytvorený projekt. Zadanie úlohy a jednotlivé fázy riešenia sú opísané v nasledujúcich podkapitolách.

2.1 Riadenie laboratórneho zariadenia

▪ Formulácia úlohy

Cieľom je vytvoriť komplexné riadenie tepelno-materiálovej sústavy v prostredí B&R Automation Studio. Hlavnou riadenou veličinou je teplota suroviny na výstupe z výmenníka zaznamenaná teplotným senzorom T1. Riadiacou veličinou sú otáčky čerpadla N1. Pri riadení sa predpokladajú konštantné otáčky čerpadla N2 s hodnotou 30% a konštantný objem horúceho média s konštantnou teplotou 65 °C na vstupe do výmenníka tepla.



Obr. 16 Schéma riadenia procesu

V procese sa nachádzajú dve riadiace slučky :

1. Riadenie teploty vody v zásobníku teplej vody.

Pre riešenie úlohy potrebujeme zabezpečiť konštantnú teplotu teplej vody, ktorá je riadenou veličinou v tejto riadiacej slučke. Vodu ohrieva elektrická špirála. Hodnota príkonu špirály predstavuje akčný zásah. Pre riadiacu slučku vytvoríme URO, v ktorom je zapojený P regulátor. Regulátor vypočíta vhodný akčný zásah pre dosiahnutie žiadanej hodnoty teploty. Recirkulačný cyklus udržiava v zásobníku konštantný objem kvapaliny.

2. Riadenie teploty vystupujúcej suroviny z výmenníka.

Riadenou veličinou je teplota suroviny. Akčný zásah zabezpečujú otáčky čerpadla N1. Riadenie na žiadanú hodnotu zabezpečuje URO so zapojeným PID regulátorom. Predpokladom pri vytváraní riadenia je konštantná teplota horúceho média a konštantné otáčky čerpadla, ktoré dopravuje horúcu vodu do výmenníka tepla. Rovnakú teplotu vody v zásobníku teplej vody zabezpečuje P regulátor v 1. riadiacej slučke. Pre túto slučku je tak potrebné navrhnuť PID regulátor, ktorý dosiahne požadovanú hodnotu teploty bez trvalej regulačnej odchýlky (TRO).

Úlohu pre jednoduchšie riešenie rozdelíme na menšie čiastkové úlohy. Spojením výsledkov z čiastkových úloh dostaneme konečné riešenie komplexnej úlohy.

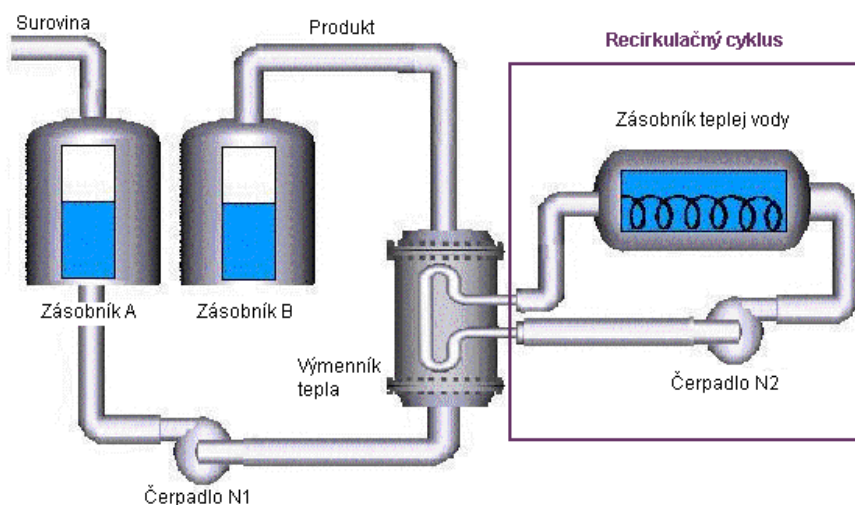
Čiastkové úlohy

- Prepojenie PLC s riadiacou konzolou a Automation Studiom
- Implementácia prepočtov fyzikálnych jednotiek
- Návrh riadenia pre 1. riadiacu slučku
- Návrh riadenia pre 2. riadiacu slučku
- Vyhodnotenie kvality navrhnutých riadení
- Vytvorenie vizualizačného prostredia

2.1.1 Zapojenie zariadenie použité pri riešení úlohy

Schéma zapojenia zariadenia použitom pri riešení zadanej úlohy je na obrázku 17. Do zásobníka A konštantne priteká a oteká voda. Prítok vody je ovládaný manuálne SOL 1. Ventil je buď otvorený alebo zatvorený, prietok vstupného prúdu nemôžeme meniť. Voda priteká stále rovnakým prietokom alebo nepriteká vôbec. Zariadenie neumožňuje merať tento prietok, preto je hodnota veličiny neznáma. Teplota vstupujúceho prúdu vody bola meraná v priebehu celého procesu kvapalinovým teplomerom, pričom namerané hodnoty sa pohybovali okolo 12 °C.

Voda zo zásobníka A putuje do výmenníka tepla. Dodávanie vody zabezpečuje peristaltické čerpadlo N1. Otáčky čerpadla v percentách nastavuje akčný zásah generovaný PID regulátorom. Vo výmenníku tepla dochádza k prestupu tepla, kde teplá voda odovzdá svoje teplo studenej vode. Produkt sa zhromažďuje v zásobníku B.



Obr. 17 Schéma zapojenia zariadenia PCT23 využitá pri riešení úlohy

Teplá voda je do výmenníka privádzaná zo zásobníka teplej vody čerpadlom N2. Otáčky čerpadla sú nastavené na konštantnú hodnotu 30 %. Teplota ohrevného média na vstupe do výmenníka je 65 °C, konštantnú hodnotu veličiny zabezpečuje URO. Horúce médium sa z výmenníka vracia späť do zásobníka teplej vody v recirkulačnom cykle.

Z výmenníka tepla vychádza produkt – ohriata surovina na žiadanú hodnotu. Produkt sa zhromažďuje v zásobníku B.

2.1.2 Prepojenie PLC s riadiacou konzolou a Automation Studiom

Riadenie je implementované prostredníctvom PLC B&R X20. PLC je prepojené s riadiacou konzolou zariadenia PCT23. PLC posiela do konzoly elektrický signál a prijíma signály zo snímačov. Povolený rozsah týchto signálov je od 0 do 5 V, čo zodpovedá 0 až 16 384 INT . Samotné programovanie a návrh riadenia sú realizované v AS.

V AS sme vytvorili nový projekt. Hardvérovú konfiguráciu sme definovali v režime „online“, keďže sme mali aktívne pripojené PLC. Systém automaticky rozpoznal pripojené zariadenie a konfiguráciu pridal do projektu. Vytvorili sme premenné, ktoré predstavujú vstupy a výstupy zariadenia. Premenné sú používané v celom projekte, preto sú vytvorené ako globálne. V položke „Global.var“ sme pridal premennú pre teplotu produktu(názov premennej „T1“), teplotu teplej vody („T2“), hladinu vody v zásobníku A(„LI“), príkon ohrievača teplej vody(„WaterHeater“), otáčky čerpadla N1(„FeedPump“), otáčky čerpadla N2(„WaterPump“) a prietok produktu na vstupe do zadržiavacej trubice („Fw“).

Potom sme jednotlivým premenným priradili fyzické adresy v položke „Physical View“. Zapojené boli analógové vstupy a výstupy, digitálne signály neboli použité. Priradenie premenných musí zodpovedať reálne pripojeným vstupom a výstupom v PLC. Analógovým vstupom sú priradené premenné v poradí „LI“, „Fw“, „T2“ a „T1“. Analógové výstupy sú rozdelené na dva moduly. Každý z modulov obsahuje dva výstupy. Prvému výstupu v prvom module zodpovedá premenná „WaterPump“, druhému „FeedPump“. V druhom module je ako prvý vstup priradená premenná „WaterHeater“.

PLC je teraz prepojené s prostredím AS aj s riadiacou konzolou. Hodnoty vstupov a výstupov pre zariadenie môžeme nastavovať v AS. Ako bolo spomenuté vyššie povolené hodnoty signálov posielaných do riadiacej konzoly sú v rozmedzí od 0 do 5 V, resp. 0 – 16 384 INT. Väčšie alebo menšie hodnoty signálov, ktoré sú mimo tohto rozsahu, by spôsobili poškodenie zariadenia. Preto v AS vytvoríme program, ktorý obmedzí hodnoty signálov na povolenú hranicu. Programy sa v AS vytvárajú v záložke „Logical View“.

Program je vytvorený v jazyku Automation basic a má názov „ObmedzeniaSW“. Pri inicializácii programu sa všetkým fyzickým vstupom a výstupom priradí nulová

hodnota. Ak je hodnota signálu väčšia ako 5 V, čo zodpovedá integeru 16 384, programu v cyklickej časti priradí premennej hodnotu 16 384. Ak bude hodnota záporná, menšia ako nula, potom program premennej priradí hodnotu nula.

Tento princíp sa použil pre všetky signály, okrem výstupu pre ohrievač. Odporúčaný príkon špirály výrobcom je 1 kW, rozsah signálu je tak od 0 – 8620 INT. Ak bude hodnota premennej „WaterHeater“ väčšia ako 8620, program priradí premennej hodnotu 8620. Dolné obmedzenie je rovnaké ako pri ostatných premenných, ak je hodnota premennej menšia ako nula, program jej priradí hodnotu nula. Nižšie sa nachádza ukážka príkazu pre jednu premennú (obr.18). Celý kód programu sa nachádza v prílohe A.

```
if(WaterHeater>8620) then
WaterHeater=8620
endif
if(WaterHeater<0) then
    WaterHeater=0
endif
```

Obr. 18 Ukážka časti kódu pre obmedzenie hodnôt signálov

2.1.3 Implementácia prepočtov fyzikálnych jednotiek

Hodnoty veličín, ktoré posiela PLC do AS sú dátového typu integer. Chceme aby údaje boli zobrazené vo fyzikálnych jednotkách (dátový typ REAL). Musíme teda vytvoriť program, ktorý prepočíta hodnotu veličiny z INT na príslušnú jednotku.

Prepočet z fyzikálnej jednotky na INT stačí vytvoriť pre premenné, ktorým z AS nastavujeme hodnotu. Ide o akčné zásahy regulátorov, ktoré predstavujú hodnoty príkonu ohrevnej špirály a otáčok čerpadla „N1“. Hodnota otáčok čerpadla „N2“ tiež zadávame z AS, preto aj pri tejto premennej uskutočníme prepočet z dátového typu REAL na INT. Na priradenie zadávaných hodnôt využívame premenné, ktoré predstavujú „setpoint(SPT)“ akčného zásahu. Hodnota tohto SPT je následne priradená zodpovedajúcej premennej. Priradenie sa nevykonáva v programe pre prepočty, ale v ostatných programoch, kde regulátory vypočítavajú akčné zásahy.

Hodnota premennej z INT sa fyzikálnu jednotku sa prepočíta podľa vzorca (2.1):

$$x = x_{min} + \frac{x_{max} - x_{min}}{y_{max} - y_{min}} (y - y_{min}) \quad (2.1)$$

kde	x	- je hodnota premennej vo fyzikálnej jednotke
	x_{min}	- je dolná hranica rozsahu premennej vo fyzikálnych jednotkách
	x_{max}	- je horná hranica rozsahu premennej vo fyzikálnych jednotkách
	y_{min}	- je dolná rozsahu premennej v INT
	y_{max}	- je horná hranica rozsahu premennej v INT
	y	- je hodnota premennej v INT

V našom prípade hodnoty x_{min} a y_{min} sú nulové a vzorec môžeme zredukovať na tvar:

$$x = \frac{y * x_{max}}{y_{max}} \quad (2.2)$$

Hodnota premennej z fyzikálnej jednotky na INT sa prepočíta vzorcom:

$$y = \frac{x * y_{max}}{x_{max}} \quad (2.3)$$

Program na prepočet jednotiek má názov „*prepocety*“ a je písaný jazykom ANSI C. V programe sa vykonávajú výpočty podľa vyššie uvedených vzorcov. Na podporu matematických výpočtov bola do projektu pridaná knižnica „*AsMath*“.

▪ Prepočet príkonu ohrevnej špirály

Rozsah príkonu ohrevnej špirály v kilowattoch(kW) je od 0 – 2 kW. Rozsah príkonu v INT je od 0 do 16 384. Z toho vyplýva, že hodnota „ x_{max} “ je 150 a hodnota „ y_{max} “ je 16 384. Pri prepočte sme najprv zmenili dátový typ premennej z INT na REAL. Potom sme uskutočnili samotný prepočet.

Vzorec na prepočet príkonu z INT na jednotku kilowatt po dosadení horných hraníc rozsahov :

$$WH_{kW} = \frac{WH_{INT} * 2}{16384} \quad (2.4)$$

kde WH_{kW} - je hodnota príkonu špirály v kilowattoch

WH_{INT} - je hodnota príkonu špirály v INT

Vzorec na prepočet príkonu z kilowattov na INT:

$$WH_{INT} = \frac{SPT_{WH} * 16384}{2} \quad (2.5)$$

kde SPT_{WH} - je hodnota príkonu špirály nastavená akčným zásahom regulátora

Princíp prevodu veličín z dátových typov je stále rovnaký, menia sa len horné rozsahy premenných. Konkrétne vzorce použité pri prepočtoch ostatných premenných sú uvedené nižšie.

▪ **Prepočet otáčok čerpadla N1 z INT na percentá(%)**

Vzorec na prepočet otáčok čerpadla N1 z dátového typu INT na %:

$$FP_{per} = \frac{FP_{INT} * 100}{16384} \quad (2.6)$$

kde FP_{per} - je hodnota otáčok čerpadla N1 v %

FP_{INT} - je hodnota otáčok čerpadla N1 v INT

Vzorec na prepočet otáčok čerpadla N1 z percent na INT :

$$FP_INT = \frac{SPT_FP * 16384}{100} \quad (2.7)$$

Rovnaké výpočty boli použité aj na prepočet teploty vody v zásobníku teplej vody, keďže rozsahy hodnôt oboch veličín sú rovnaké.

▪ Prepočet teploty produktu

Vzorec na výpočet teploty produktu v stupňoch Celzia

$$T1_stC = \frac{T1_INT * 150}{16384} \quad (2.8)$$

kde $T1_stC$ - je hodnota teploty produktu v stupňoch Celzia
 $T1_INT$ - je hodnota teploty produktu v integeroch

Rovnaký prepočet bol použitý aj na prepočet teploty vody v zásobníku teplej vody, keďže rozsahy hodnôt oboch veličín sú rovnaké.

▪ Prepočet výšky hladiny

Vzorec na prepočet výšky hladiny z dátového typu INT na milimetre:

$$L1_mm = \frac{L1_INT * 250}{16384} \quad (2.9)$$

kde $L1_mm$ - je hodnota výšky hladiny v milimetroch
 $L1_INT$ - je hodnota výšky hladiny v integeroch

Všetky novo použité premenné sme po vytvorení programu museli zadefinovať. Keďže tieto veličiny sú použité v celom projekte, priradíme ich k globálnym premenným a zvolíme príslušný dátový typ premennej. Pridanými premennými sú „*T1_stC*“, „*T2_stC*“, „*L1_mm*“, „*WH_kW*“, „*FP_per*“, „*WP_per*“, „*SPT_WH*“, „*SPT_FP*“, „*SPT_WP*“, ktoré sú dátového typu REAL. Kód vytvorený na prepočet jednotiek sa nachádza v prílohe B.


2.1.4 Návrh riadenia pre 1. riadiacu slučku



Prvá riadiaca slučka udržiava konštantnú teplotu horúceho média na vstupe do výmenníka tepla. Veličina, ktorú riadime je teplota vody v zásobníku teplej vody. Riadiacou veličinou je príkon elektrickej špirály, ktorá ohrieva vodu.

Ak chceme systém riadiť musíme poznať jeho dynamiku. Preto prvým bodom riešenia tejto úlohy je identifikácia systému z prechodovej charakteristiky (PCH), časového priebehu výstupnej veličiny. Výstupom z identifikácie je vstupno-výstupný model systému v tvare prenosovej funkcie.

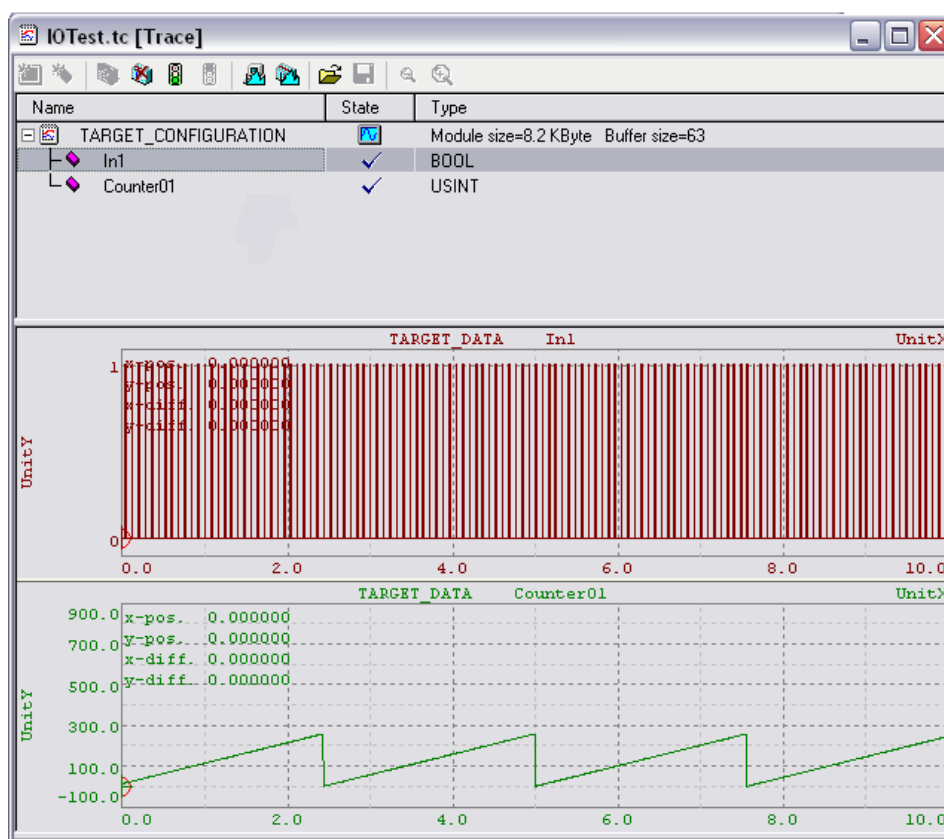
▪ Identifikácia systému

Na vykreslenie PCH potrebujeme dáta. Prostredie na zber a ukladanie dát poskytl AS. Zabudovaná funkcia „*trace*“ umožňuje zvoliť premenné z AS, ktorých hodnoty sa ukladajú. Na spustenie „*trace*“ editora musíme mať otvorenú softvérovú konfiguráciu („*Configuration View*“) a musíme byť v „*online*“ móde. V otvorenom okne softvérovej konfigurácie nastavíme kurzor na príslušný program, z ktorého chceme zaznamenávať premenné. Potom vyberieme z horného menu položku „*Open/Trace*“.

Otvorí sa nové okno, v ktorom pridáme novú konfiguráciu sledovania premenných. Vo vlastnostiach konfigurácie nastavíme periódu vzorkovania hodnôt a počet hodnôt, ktoré chceme uložiť pre každú premennú. V našom prípade sme zvolili periódu vzorkovania 3 a počet ukladaných hodnôt 30 000. Potom pridáme premenné, ktoré chceme sledovať a túto konfiguráciu nainštalujeme stlačením ikony . Po

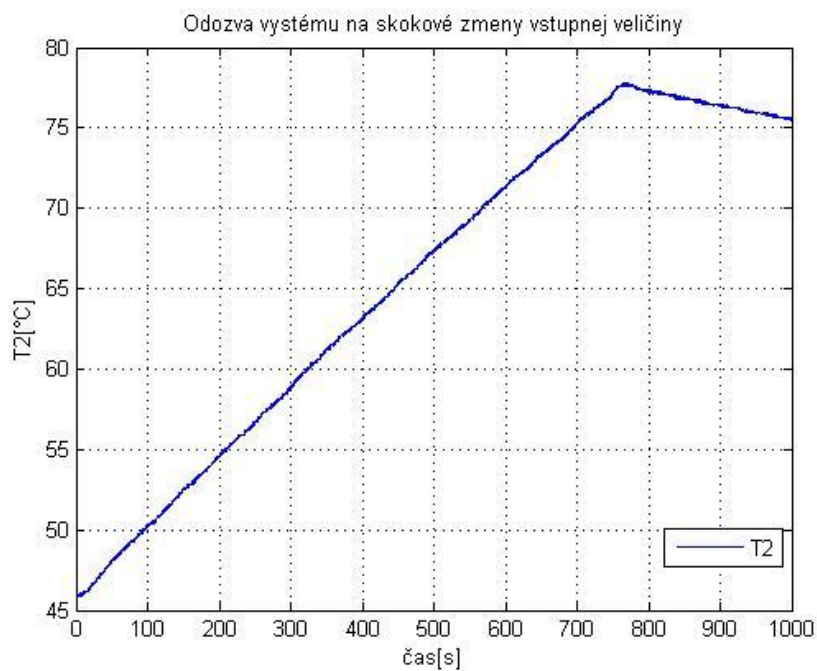
inštalácií sa dáta začnú ukladať. Prerušiť ukladanie môžeme ikonou  a znova ho môžeme spustiť ikonou .

Dáta sa ukladajú do internej pamäte PLC. Po ukončení zberu dát dáme zobrazit' uložené hodnoty a následne ich uložíme do pamäte PC. Dáta možno ukladať ako typ „mch“, „mat“, „.csv“ a „.txt“. My sme zvolili uloženie v textovom formáte. Ukážka trace editora so zobrazenými údajmi je na Obr. 19 [9].



Obr. 19 Ukážka trace editora

Po nastavení konfigurácie „trace“ editora spustíme ukladanie údajov a vykonáme skokovú zmenu vstupnej veličiny – príkonu ohrevnej špirály. Odozvu teploty vody na zmenu príkonu vidíme na obr. 20.



Obr. 20 Prechodová charakteristika systému

Teplota vody konštantne stúpa, klesať začne až po aktivácii ochrany zariadenia pred prehriatím. Systém nedosiahne ustálený stav – je astatický s integračným charakterom. Prenos takéhoto systému je vyjadrený v tvare (2.10):

$$G_s(s) = \frac{K}{s} \quad (2.10)$$

kde K - je zosilnenie systému.

Zosilnenie systému je smernica priamky PCH. Smernicu vypočítame z derivácie rovnice priamy (2.11):

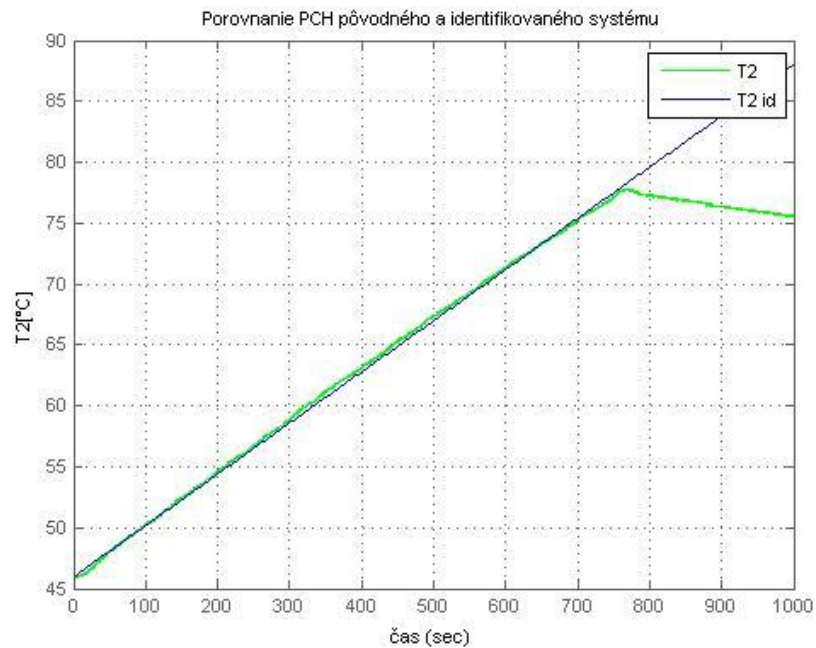
$$y = ax + b \quad (2.11)$$

Dosadením súradníc bodov priamky sme získali parametre $a = 0,042$ a $b = 46,256$. Derivácia priamky je samotný parameter „ a “, čo je aj zosilnenie systému.

Identifikovaný prenos je:

$$G_s(s) = \frac{0,042}{s}$$

Na obr. 21 je porovnanie pôvodnej a identifikovanej PCH systému.



Obr. 21 Porovnanie PCH pôvodného a identifikovaného systému

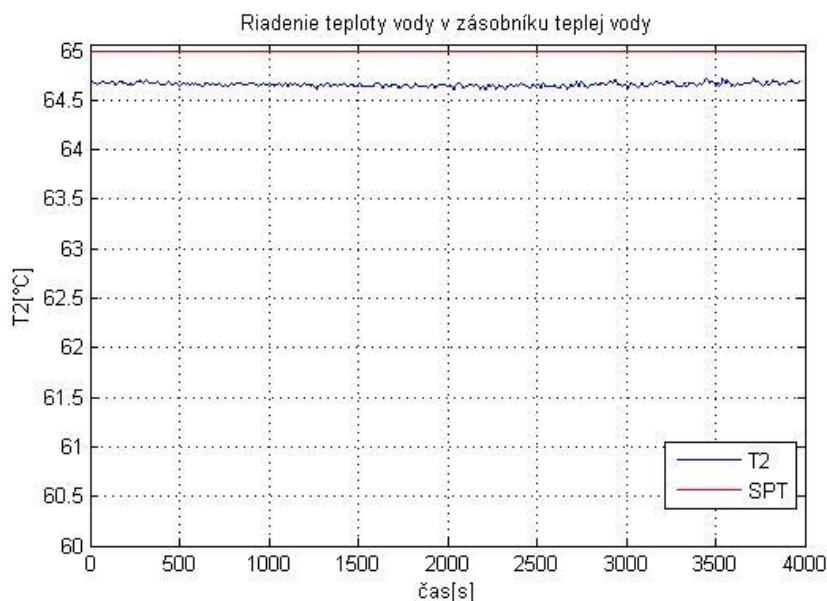
Po získaní prenosu systému sme pre systém navrhli P regulátor. Keďže regulátor tvorí iba proporcionálna zložka, stačilo nájsť vhodné zosilnenie regulátora (Z_R). Pri výpočte Z_R sme vychádzali z Chien–Hrones-Reswickovej metódy. Keďže metóda predpokladá identifikovaný systém prvého rádu s dopravným oneskorením a náš systém je astatický, získané zosilnenie slúžilo iba ako štartovacia hodnota. Následne sme hodnotu Z_R menili, kým regulátor neuriadil daný systém na žiadanú hodnotu s požadovanou presnosťou. Vhodné zosilnenie P regulátora na riadenie teploty vody v ohrievači bolo 0,7.

Na riadenie URO sme vytvorili program v prostredí AS v jazyku „*Ladder Diagram*“. Spustenie riadenia ovláda binárna premenná „start“. Ak je riadenie zapnuté,

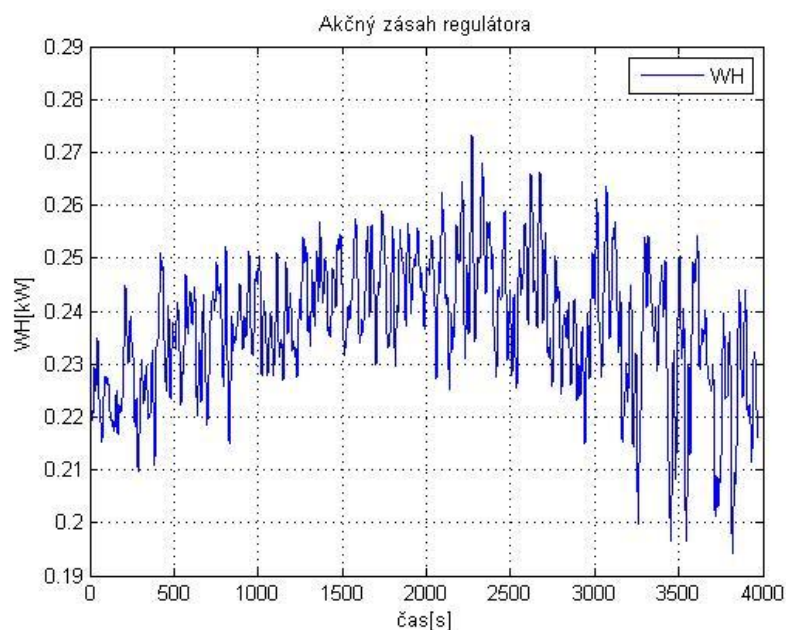
odčítava sa aktuálna hodnota teploty vody v ohrievači od žiadanej. Tým získame regulačnú odchýlku, ktorú násobíme zadanou hodnotou zosilnenia regulátora. Výpočtom dospejeme k vhodnému akčnému zásahu regulátora.

Povolený rozsah príkonu ohrevnej špirály je od 0 do 1 kW. Preto aj akčný zásah obmedzíme na hodnoty z tohto intervalu. Ak bude hodnota väčšia ako 1 priradíme akčnému zásahu hodnotu 1. Ak bude menšia ako 0 priradíme hodnotu 0. Nakoniec programu pomocou bloku „move“ vložíme hodnotu akčného zásahu vypočítaného regulátorom do premennej pre príkon špirály v kW („WH_kW“). Ukážka vytvoreného programu sa nachádza v prílohe C.

Na obr. 22 je graf riadenia teploty v zásobníku teplej vody pomocou P regulátora. Regulátor neodstraňuje TRO, lebo neobsahuje integračnú zložku. Priemerná regulačná odchýlka je 0.3394 °C. Na dosiahnutie konštantnej teploty vody je toto riadenie postačujúce. Akčný zásah regulátora je zobrazený na obr. 23.



Obr. 22 Riadenie teploty ohrevného média na vstupe do výmenníka P regulátorom



Obr. 23 Akčný zásah regulátora

2.1.5 Návrh riadenia pre 2. riadiacu slučku

Druhá riadiaca slučka zabezpečuje riadenie teploty produktu na výstupe z výmenníka pri konštantnom prietoku teplej vody, konštantnom objeme a teplote ohrevného média. Riadenou veličinou je teplota produktu a riadiacou veličinou sú otáčky čerpadla N1.

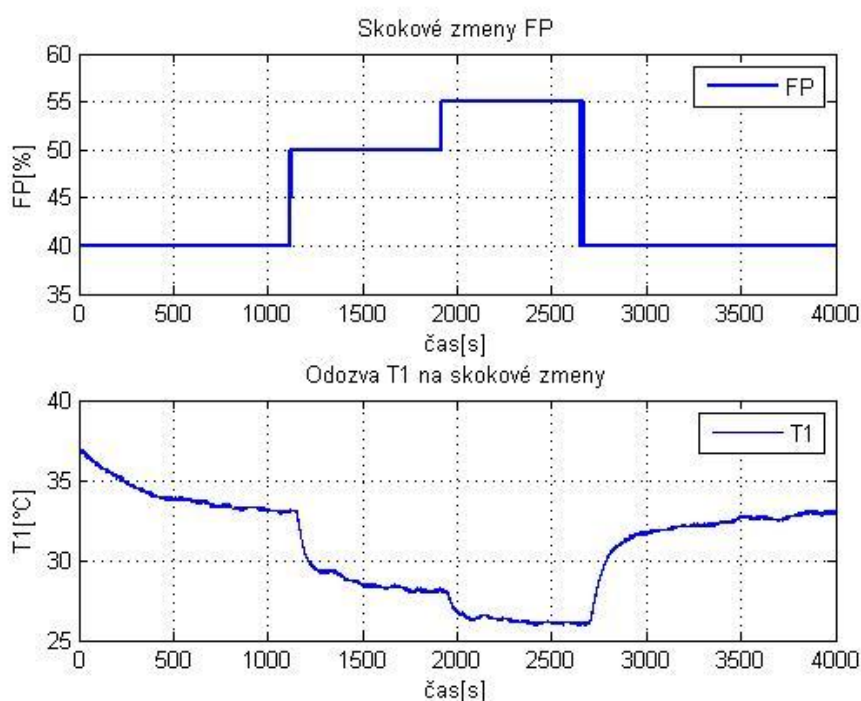
Konštantný prietok média zabezpečíme jednoducho nastavením otáčok čerpadla N2 na 30%. Horúce médium sa z výmenníka tepla vracia recirkuláciou do zásobníka teplej vody, čím je udržiavaný konštantný objem. Voda v zásobníku teplej vody je ohrievaná elektrickou špirálou, ktorej príkon nastavuje P regulátor v prvej riadiacej slučke. Vypočítaným akčným zásahom dosiahneme žiadajú hodnotu teploty ohrevného média. V našom prípade bola teplota 65 °C. Riadenie P regulátorom dostatočne udrží teplotu na konštantnej hodnote.

Po splnení všetkých predpokladov navrhujeme samotné riadenie teploty produktu. Do URO zapojíme PID regulátor. Aj pri tomto návrhu riadenia postupujeme vo viacerých etapách. Pred výpočtom parametrov regulátora musíme reálny systém zidentifikovať a overiť správnosť identifikácie.

▪ Identifikácia systému

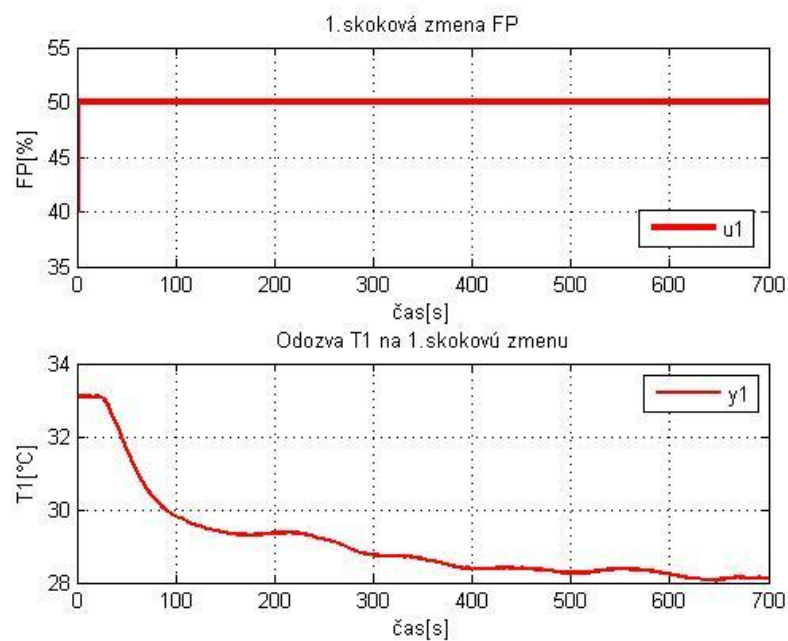
Na získanie modelu systému sme použili deterministické metódy identifikácie. Pri týchto metódach predpokladáme, že vstupná veličina je deterministická, čiže neuvažujeme pôsobenie náhodných veličín na systém. Výsledný model je v tvare prenosovej funkcie. Pred získaním PCH, je nutné, aby bol systém v ustálenom stave. Ako vstupný signál bola použitá skoková zmena vstupnej veličiny – hodnota otáčok čerpadla N1. Ostatné vstupné veličiny boli konštantné. [10]

Po aktivácii zberu dát v „trace“ editore sme vykonali tri skokové zmeny vstupnej veličiny z hodnoty 40 % na 50 %, z 50 % na 55 % a z 55 % na 40 %. Skokové zmeny a odozvy na skokové zmeny sú zobrazené na obr. 24.

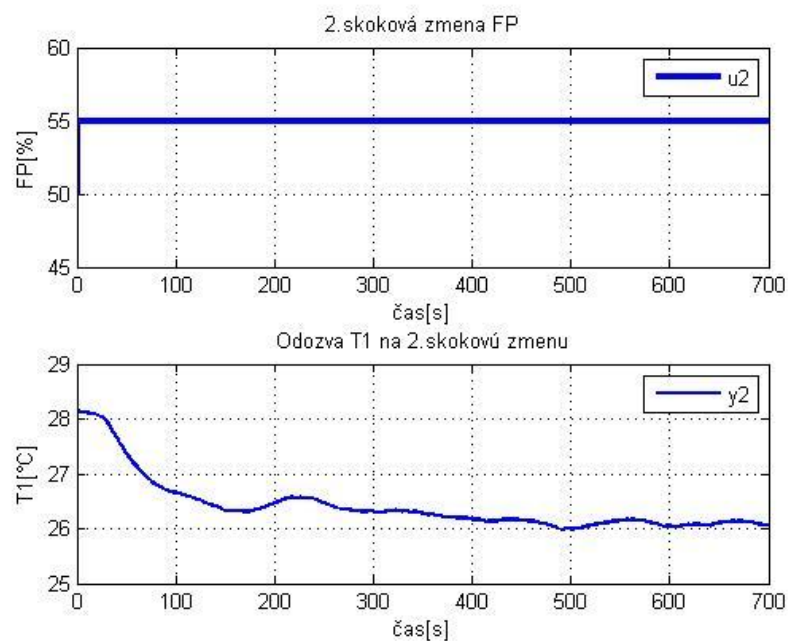


Obr. 24 Skokové zmeny otáčok čerpadla N1 a príslušné PCH

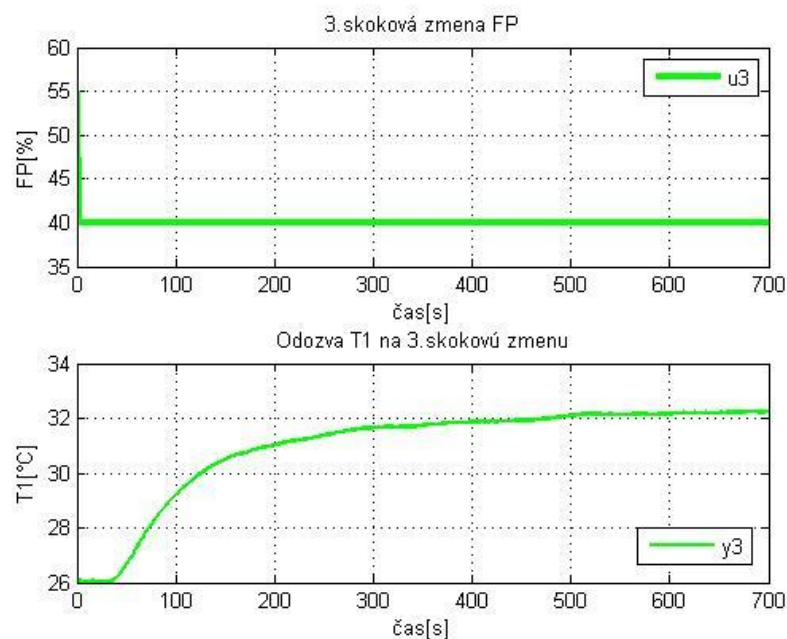
Vektory skokových zmien a PCH sme rozdelili podľa jednotlivých skokov (obr. 25,26,27)



Obr. 25 Prvá skoková zmena a PCH



Obr. 26 Druhá skoková zmena a PCH



Obr. 27 Tretia skoková zmena a PCH

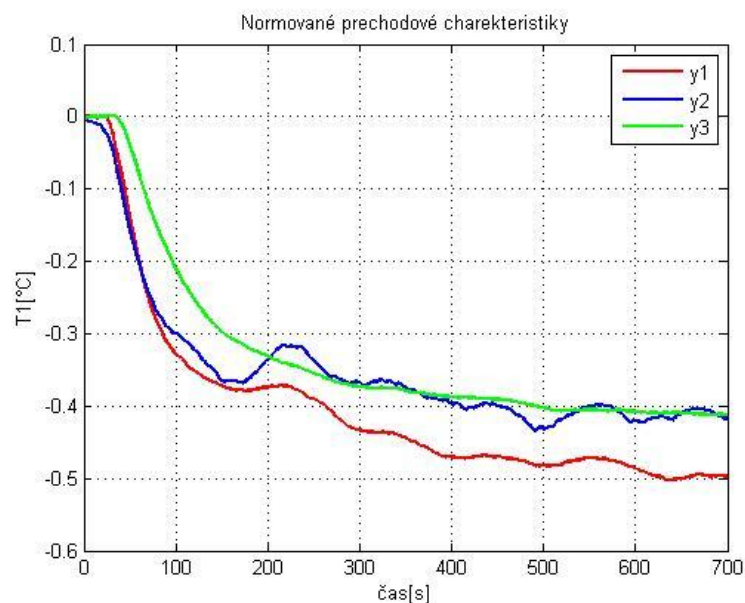
Získané prechodové charakteristiky sme upravili tak, aby sme ich mohli použiť pri identifikácii. Najprv sme prechodovú charakteristiku posunuli, aby počiatočná hodnota výstupu bola v nule. Potom sme charakteristiku normalizovali vzhľadom k veľkosti skoku vstupnej veličiny a to nasledovným spôsobom (2.12):

$$\hat{y}(t - t_0) = \frac{y(t) - y(t_0)}{u_\infty - u_0} \quad (2.12)$$

kde

- $\hat{y}(t - t_0)$ - je normovaný výstup
- t_0 - je počiatočný čas
- t - je aktuálna hodnota času
- $y(t)$ - je hodnota výstupu v čase t
- $y(t_0)$ - je hodnota výstupu v čase t_0
- u_∞ - je hodnota skokovej zmeny v nekonečne
- u_0 - je hodnota skokovej zmeny v čase t_0

Priebeh výstupu pre normovaný vstup je zobrazený na obrázku 28.



Obr. 28 Normované prechodové charakteristiky

Z grafu je zrejmé, že systém je nelineárny, keďže výstupy systému pri skokových zmenách majú rôzne hodnoty. Na základe minimalizácie štvorcov odchýliek medzi jednotlivými nameranými bodmi y_{ik} a predpokladanou teoretickou hodnotou \hat{y}_i získame vzťah (2.13):

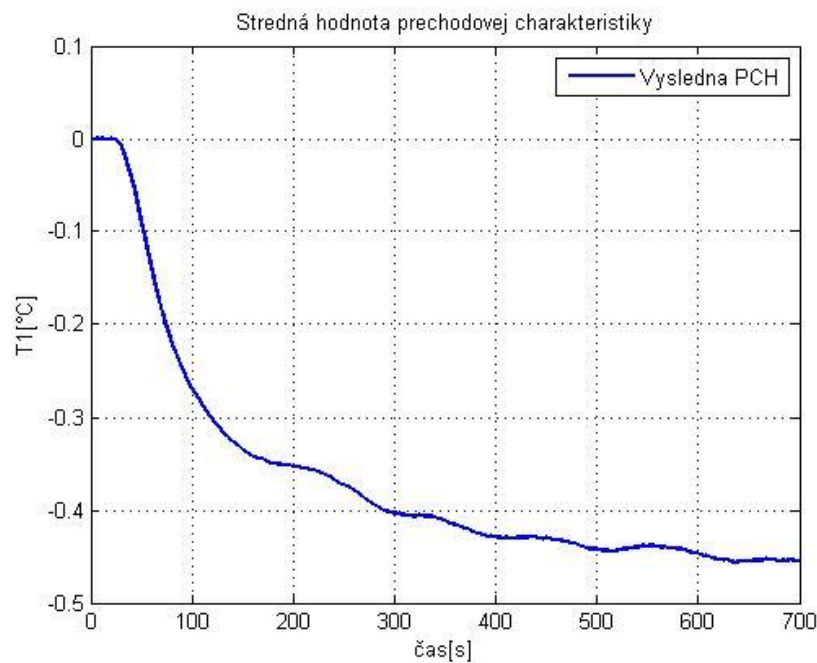
$$\hat{y}_i = \frac{\sum_{k=1}^N \Delta u_k y_{ik}}{\sum_{k=1}^N (\Delta u_k)^2} \quad (2.13)$$

kde i je i -ty bod prechodovej charakteristiky,
 k k -te meranie, $k=1, \dots, N$,
 Δu_k skoková zmena vstupu pri k -tom meraní,
 y_{ik} hodnota výstupu pri k -tom meraní v i -tom intervale
 \hat{y}_i výsledná hodnota PCH v čase $t=i\Delta t$, kde Δt je perióda vzorkovania.

Ak $\Delta u_k = 1$, potom dostaneme aritmetický priemer

$$\hat{y}_i = \frac{\sum_{k=1}^N y_{ik}}{N} \quad (2.14)$$

V našom prípade použijeme posledný vzťah (2.14) na získanie prechodovej charakteristiky, ktorú budeme identifikovať (Obr. 29) [10].



Obr. 29 Stredná hodnota prechodovej charakteristiky systému

Po spracovaní prechodovej charakteristiky identifikuje systém prenosom prvého rádu v tvare (2.15):

$$F(s) = \frac{K}{Ts+1} e^{-Ds} \quad (2.15)$$

kde K - je zosilnenie systému

T - je časová konštanta systému

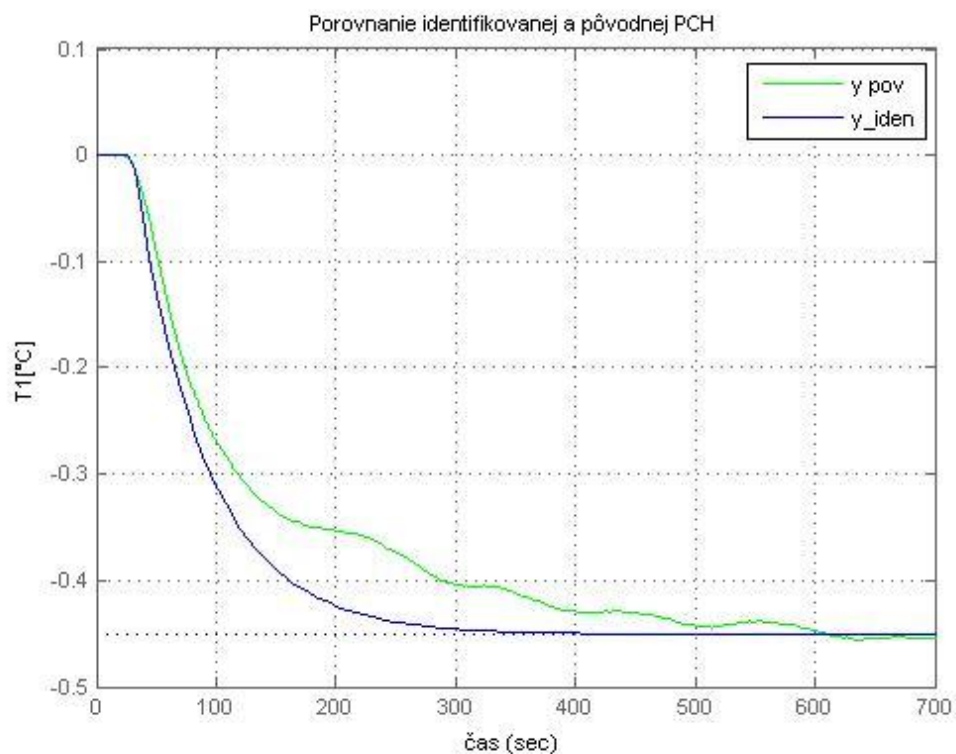
D - je dopravné oneskorenie systému.

Zosilnenie K je hodnota prechodovej charakteristiky v nekonečne $K = y(\infty)$. Hodnota T sa dá určiť ako čas, za ktorý dosiahne výstupná veličina 63% svojej ustálenej hodnoty. D odčítame ako oneskorenie medzi časom kedy sme vykonali skokovú zmenu a časom kedy systém na zmenu zareagoval. [10]

Popísaným spôsobom sme získali hodnoty $K = -0.4486 \text{ } ^\circ\text{C}$, $T = 65 \text{ s}$, $D = 30 \text{ s}$.
Identifikovaný prenos nášho systému je:

$$F(s) = \frac{-0.4486}{65s+1} e^{-30s}$$

Nakoniec sme porovnali identifikovanú prechodovú charakteristiku s pôvodnou prechodovou charakteristikou (Obr.30). M – file použitý pri identifikácii systému sa nachádza v prílohe D.



Obr. 30 Porovnanie pôvodnej a identifikovanej prechodovej charakteristiky

▪ Návrh PID regulátora

Po identifikácii modelu systému sme navrhli PID regulátor bez interakcie, ktorý zabezpečí riadenie teploty produktu na žiadanú hodnotu bez regulačnej odchýlky. Prenos PID regulátora(2.16) je zložený z troch zložiek proporčionálnej, integračnej a derivačnej:

$$G_R(s) = \frac{U(s)}{E(s)} = Z_R \left(1 + \frac{1}{T_i s} + T_D s \right) \quad (2.16)$$

kde	U(s)	- je Laplaceov obraz riadiacej veličiny u(t)
	E(s)	- je Laplaceov obraz regulačnej odchýlky e(t)
	Z _R	- je zosilnenie regulátora
	T _i	- je integračná časová konštanta
	T _D	- je derivačná časová konštanta

[11]

Na návrh parametrov regulátora sme použili experimentálne metódy. Tieto metódy na určenie parametrov regulátora skúmajú dynamické správanie riadeného procesu, využívajú sa parametre získané pri identifikácii z PCH. [12]

Parametre PID regulátora sme vypočítali viacerými metódami :

- Coohen – Coonovou metódou
- Chien – Hrones – Reswickovou metódou s 0 % preregulovaním
- Chien – Hrones – Reswickovou metódou s 20 % preregulovaním
- Metódou priamej syntézy regulátora
- Ziegler – Nicholsonovou metódou

Pomocou simulačnej schémy s použitím prenosu systému sme simulovali riadenie modelu so všetkými navrhnutými regulátormi. Najlepšiu kvalitu riadenia z hľadiska času regulácie a maximálneho preregulovania mala metóda priamej syntézy regulátora.

Táto metóda vyžaduje voľbu konštanty T_{URO} , ktorá je menšia ako časová konštanta systému T . Parametre regulátora sa vypočítajú podľa vzťahov (2.17),(2.18),(2.19):

$$Z_R = \frac{2*(T+D)}{3*K*(T_{URO}+D)} \quad (2.17)$$

$$T_i = T + 0,5 * D \quad (2.18)$$

$$T_D = \frac{0,5*D*T_{URO}}{T_{URO}+D} \quad (2.19)$$

Hodnoty parametre riadeného procesu sú: $K = -0,4486$ °C, $T = 65$ s , $D = 30$ s. Po Hodnotu T_{URO} sme zvolili 35s. Po dosadení hodnôt do vyššie uvedených vzorcov dostaneme parametre regulátora:

$$Z_R = -1.9601$$

$$T_i = 80$$

$$T_D = 8.0769$$

Navrhnutý regulátor je spojitý a reálne zapojené zariadenie pracuje diskkrétne. Preto potrebujeme regulátor diskretizovať. Prenos diskrétného regulátora je v tvare (2.20) :

$$G_R(z) = \frac{c_0 z^2 + c_1 z + c_2}{z^2 - z} \quad (2.20)$$

Konštanty c_0 , c_1 a c_2 sú definované ako:

$$c_0 = Z_R + \frac{Z_R * T_s}{T_i} + \frac{Z_R * T_D}{T_s} \quad (2.21)$$

$$c_1 = -Z_R - \frac{2 * Z_R * T_D}{T_s} \quad (2.22)$$

$$c_2 = \frac{Z_R * T_D}{T_s} \quad (2.23)$$

kde T_s - je perióda vzorkovania

V našom prípade je T_s rovná 0,1 s. Hodnoty parametrov diskrétného regulátora potom sú:

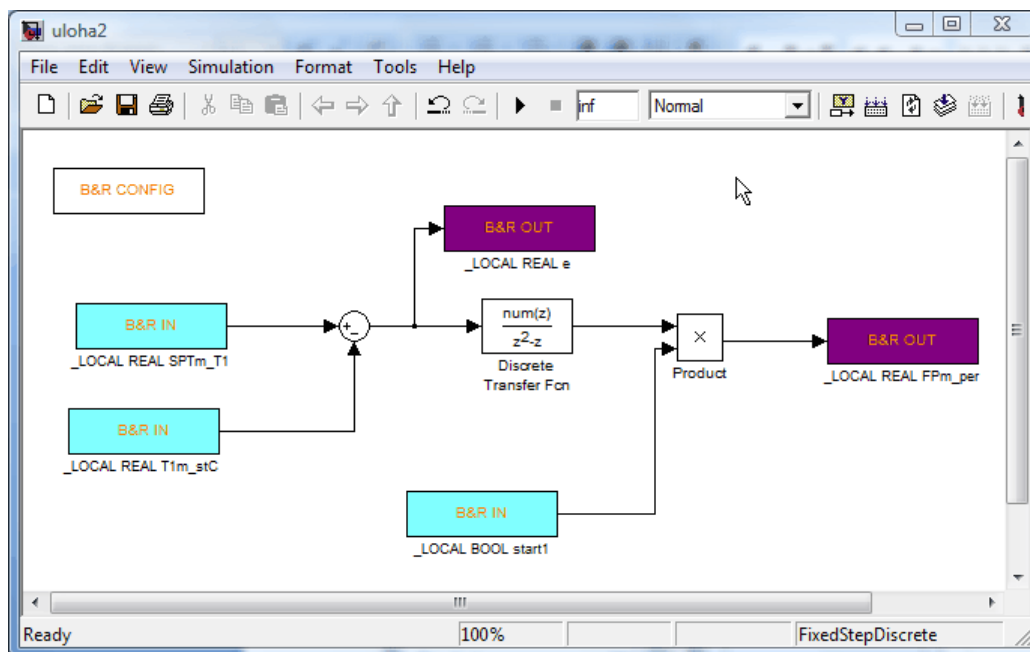
$$c_0 = -160.2812$$

$$c_1 = 318.5974$$

$$c_2 = -158.3187$$


▪ Vytvorenie riadiacej schémy

Navrhnutý regulátor zapojíme do URO. Schému riadenia vytvoríme v Simulinku s použitím B&R funkčných blokov (obr. 31). V bloku „*B&R Config*“ nastavíme mód na „*Code Generation*“, čím sa v AS automaticky vygeneruje kód programu v jazyku C. V menu „*Simulation / Configuration parameters*“ v položke „*Solver*“ zvolíme periódu vzorkovania. V našom prípade sme použili hodnotu 0,1 s. Rovnakú hodnotu nastavíme aj u všetkých blokov v schéme. V položke „*B&R Basic Settings*“ nastavíme cestu k adresáru, v ktorom sa nachádza projekt v AS, zvolíme názov programu a konfiguráciu. Konfigurácia sa musí zhodovať s vytvorenou konfiguráciou v AS.



Obr. 31 Schéma riadenia v Simulinku

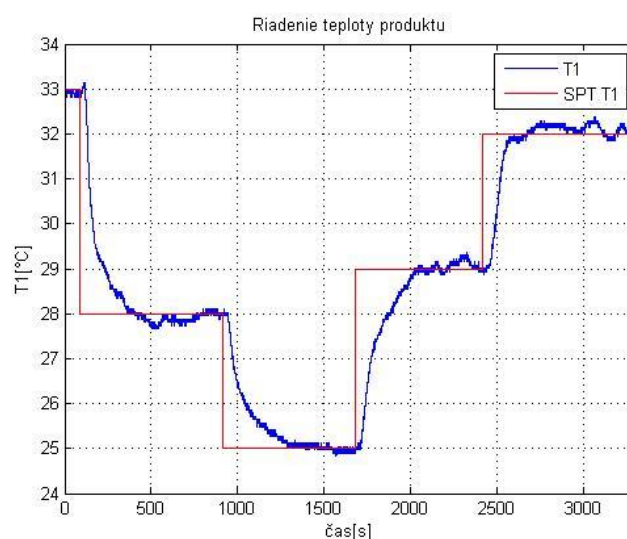
Do bloku pre prenos regulátora vstupuje regulačná odchýlka, čo je rozdiel medzi žiadanou hodnotou teploty produktu a aktuálnou hodnotou teploty. Z bloku regulátora vystupuje akčný zásah, ktorý je násobení hodnotou premennej „start1“. Táto premenná reprezentuje stav automatického riadenia. Ak je režim aktívny, hodnota premennej je 1. Otáčkam čerpadla potom bude priradená hodnota akčného zásahu vypočítaného regulátorom. V opačnom prípade je hodnota premennej 0 a aj hodnota akčného zásahu je 0. Otáčky čerpadla sa potom nastavujú manuálne.

Po vytvorení schémy spustíme automatické generovanie kódu kliknutím na ikonu  („Incremental build“). Premenné použité v schéme sú lokálne a automaticky sa pridávajú do AS pri generovaní kódu. Ide o premenné „SPTm_T1“, „T1m_stC“, „start1“ a „FPm_per“. V AS sa vytvorí nový program s názvom, ktorý sme zvolili v konfigurácii. V našom prípade sme volili názov „PID“.

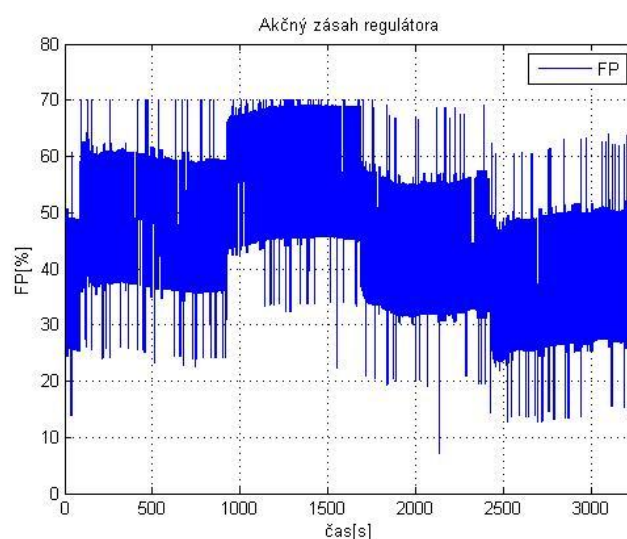
Ostáva priradiť lokálne premenné použité v programe k globálnym premenným, ktoré predstavujú fyzicky pripojené vstupy. V kóde programu v časti „uloha2.c“ pridáme priradenie premennej „T1_stC“ do premennej „T1m_stC“ a premennej „SPT_T1“ do premennej „SPTm_T1“. Na konci kódu vložíme obmedzenie na hodnoty akčného zásahu regulátora. Ak bude hodnota väčšia ako 70%, priradí sa hodnota 70% a pri hodnote menšej ako 0 sa priradí hodnota 0. Za obmedzenia vložíme priradenie

hodnoty premennej „*FPm_per*“ do premennej „*SPT_FP*“. Zdrojový kód sa nachádza v prílohe E.

Nakoniec projekt v AS „*zbuildujeme*“, nastavíme ukladanie dát funkciou „*trace*“ a kvalitu navrhnutého riadenie otestujeme. Zadali sme žiadanú hodnotu teploty a sledovali či systém dosiahne túto hodnotu. Riadenie teploty produktu navrhnutým regulátorom je zobrazené na obr. 32. Na obr. 33 je zobrazený časový priebeh akčného zásahu regulátora.



Obr. 32 Riadenie teploty produktu bez filtra



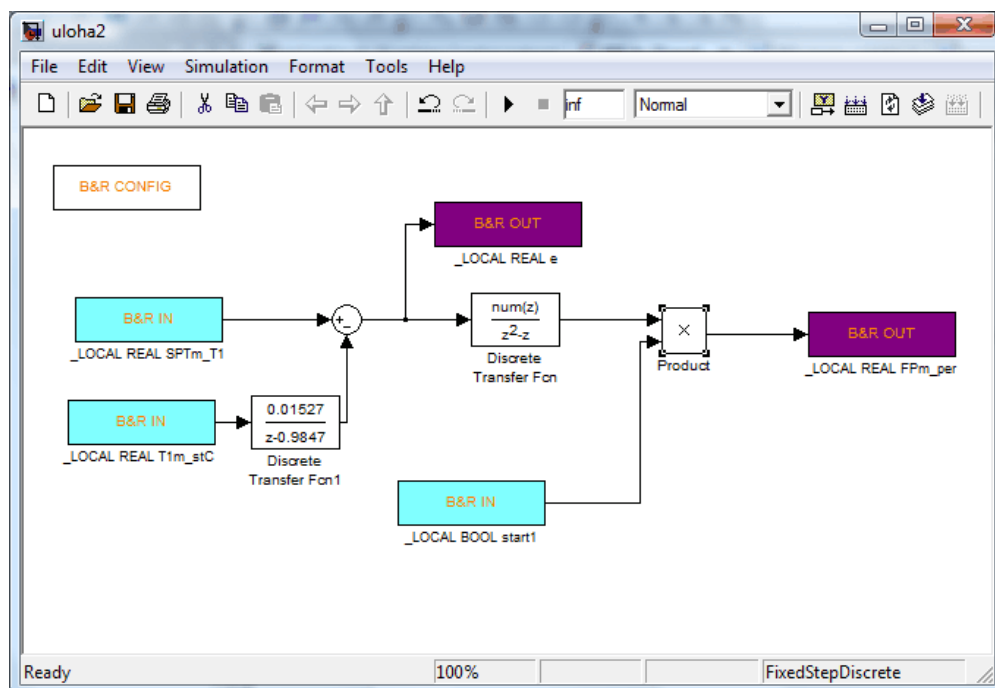
Obr. 33 Akčný zásah regulátora pri riadení bez filtra

Z grafov vidíme, že signál akčného zásahu je zašumený, preto aj riadenie teploty nedosiahne žiadanú hodnotu bez TRO. Tento efekt je spôsobený zašumeným signálom, ktorý prichádza do regulátora. Šum odstránime zapojením filtra 1.rádu na signál teploty produktu. Časová konštanta filtra musí byť menšia ako časová konštanta systému.

Prenos navrhnutého filtra v diskkrétnej oblasti:

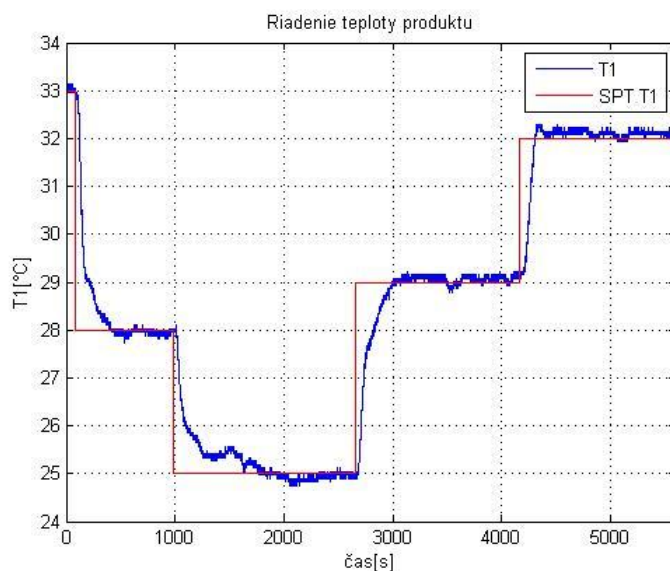
$$G_F(z) = \frac{0,01527}{z - 0.9847}$$

Navrhnutý filter vložíme do simulinkovej schémy za blok teploty produktu (obr.34).

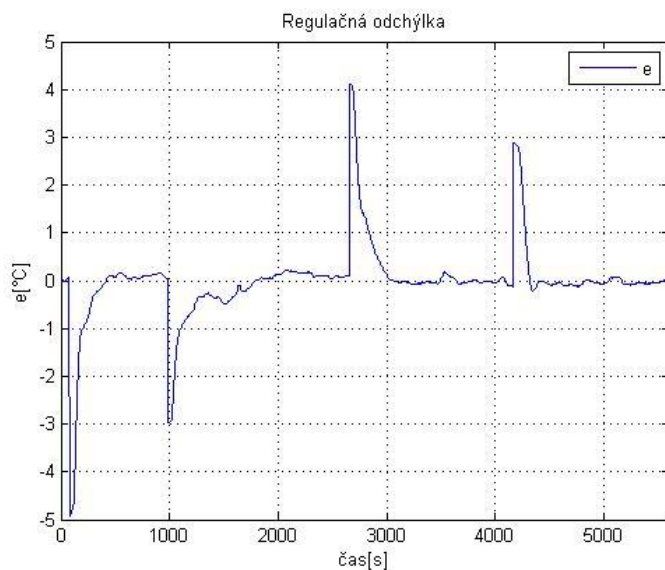


Obr. 34 Simulačná schéma s filtrom 1.rádu

Po zavedení filtra sa kvalita riadenia vylepšila (obr.35). Zmenšil sa čas regulácie a po dosiahnutí žiadanej hodnoty systém nekmitá a sleduje žiadanú hodnotu teploty. Na obr.37 vidíme, že pri riadení systému sa vyskytli poruchy vstupnej veličiny v čase 100 a po čase 4000. Tieto poruchy mohli spôsobiť prekmit na výstupe teploty produktu. Regulačná odchýlka je zobrazená na obr.36.

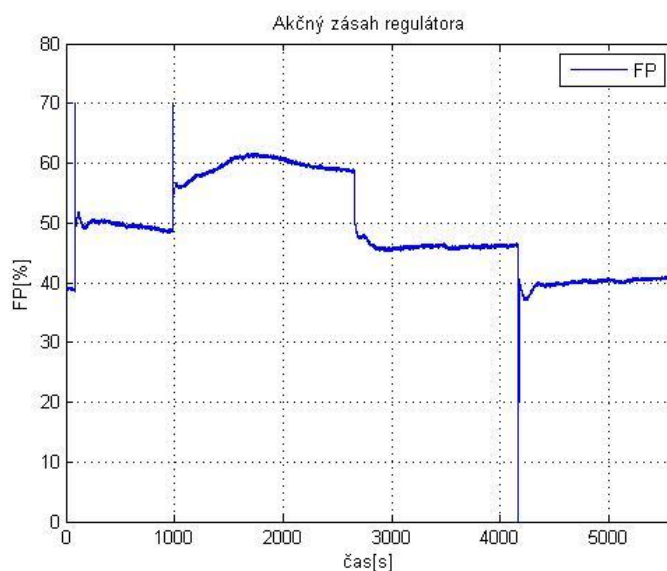


Obr. 35 Riadenie teploty produktu s filtrovaním signálov



Obr. 36 Regulačná odchýlka

Viditeľný rozdiel nastal u akčného zásahu regulátora (obr.37). Kým pri riadení bez filtra sa hodnoty otáčok čerpadla pohybovali vo veľkom intervale, pri riadení s filtrom sa plynulo menia. Tým aj chránime čerpadlo od predčasného vyradenia z prevádzky.



Obr. 37 Akčný zásah regulátora

▪ Vyhodnotenie kvality navrhnutých riadení

Pri návrhu URO sa kladie požiadavka čo najrýchlejšieho a najpresnejšieho sledovania žiadanej veličiny riadenou veličinou. Pre kvantifikáciu tejto požiadavky používame rôzne kritéria. [13]

Maximálne preregulovanie (e_{max}) – udáva v percentách normovanú veľkosť maximálnej regulačnej odchýlky po prvom prekročení žiadanej veličiny. Matematické vyjadrenie je dané vzťahom (2.24).

$$e_{max} = \frac{y_{max} - y_{\infty}}{y_{\infty}} 100 \quad (2.24)$$

Čas regulácie(T_{reg}) – je čas, po ktorom regulačná odchýlka nebude väčšia ako stanovená hodnota. Obyčajne sa používa delta okolie žiadanej veličiny, v ktorom sa po dosiahnutí T_{reg} bude výstup vždy nachádzať [13]

Navrhnuté riadenia teploty produktu PID regulátorom sme posúdili s využitím vyššie spomenutých kritérií kvality. Porovnávali sme riadenie bez filtra a s filtrom 1. rádu. Hodnoty ukazovateľov kvality sú uvedené v tabuľke 5. Posledný stĺpec obsahuje hodnotu sumy štvorcov odchýlky teploty od žiadanej hodnoty.

PID regulátor	e_{max} [%]		T_{reg} [s]		Sumy štvorcov odchýlok	
	Bez filtra	S filtrom	Bez filtra	S filtrom	Bez filtra	S filtrom
1.zmena	1,22	0.96	469,5	256,5	6,7923e+005	6.7183e+005
2.zmena	0,72	1.01	338	740	5,8061e+005	8.5690e+005
3.zmena	1,31	0.8	650	308	9,8871e+005	6.1498e+005
4.zmena	1,16	0,94	662,5	191,5	1,7430e+005	4.8124e+004
Priemerné hodnoty	1,1	0,93	530	374	6.0571e+005	5.4796e+005

Tab. 5 Ukazovatele kvality riadenia

Zo získaných výsledkov môžeme usúdiť, že riadenie procesu je jednoznačne lepšie pri zavedení filtra pre signál snímanej teploty produktu. Všetky tri použité kritéria mali v tomto prípade menšiu hodnotu.

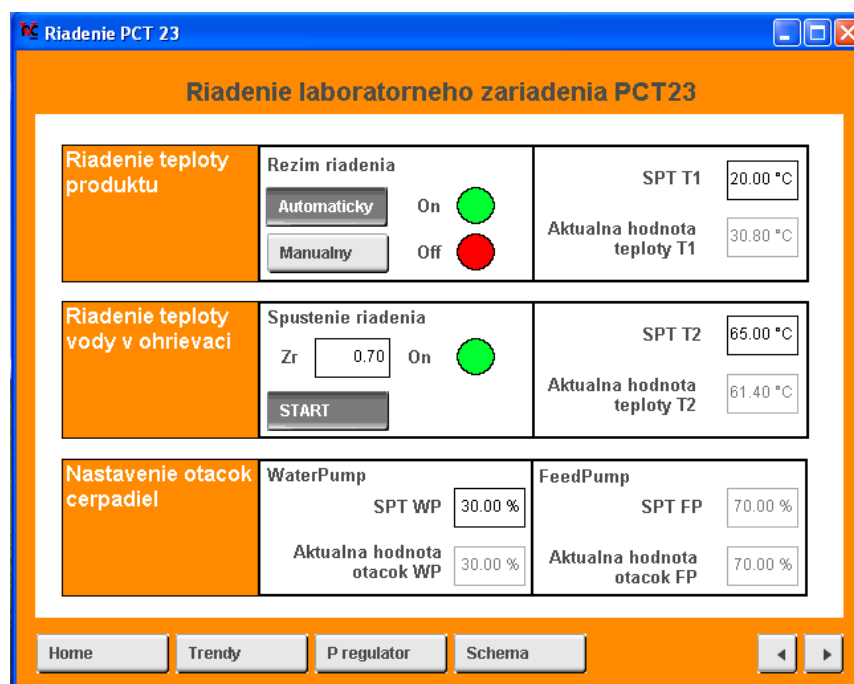
Dosiahnutie ustálenej hodnoty teploty produktu bez zapojeného regulátora trvalo 30 – 40 minút. Pri riadení URO bol čas potrebný na dosiahnutie žiadanej hodnoty päťnásobne menší. Pričom čas regulácie s filtrom bol o 2,5 minúty menší ako pri riadení bez filtra.

Pri riadení zariadenia TRO nemala natrvalo nulovú hodnotu, jej hodnota sa len približovala k nule. Systém dosiahol žiadajú hodnotu na istý čas, no potom sa vychýlila o minimálnu hodnotu, napr. 0.01 °C. Tento jav zapríčinil vysoké hodnoty sumy štvorcov odchýlok od žiadanej hodnoty.

2.1.6 Vytvorenie vizualizačného prostredia

Poslednou časťou riešenia je vytvorenie grafického vizualizačného prostredia (GUI). Prostredníctvom vizualizácie môžeme jednoducho zadávať žiadané hodnoty veličín, nastavovať parametre regulátora, vidieť časové priebehy veličín a ovládať riadenie prostredníctvom tlačidiel. GUI je vytvorené v prostredí AS.

Vizualizáciu sme najprv pridali k hardvérovej konfigurácii projektu vytvoreného v AS. V softvérovej konfigurácii sme v položke „*ArConfig.rtc*“ pridali IP adresu PLC. Na túto adresu sa pripája „*B&R VNC Viewer*“, ktorý zobrazí vytvorenú vizualizáciu. V tejto položke priradíme aj heslo pre spustenie vizualizácie. Projekt v AS „*zbuildujeme*“ a spustíme „*B&R VNC Viewer*“.

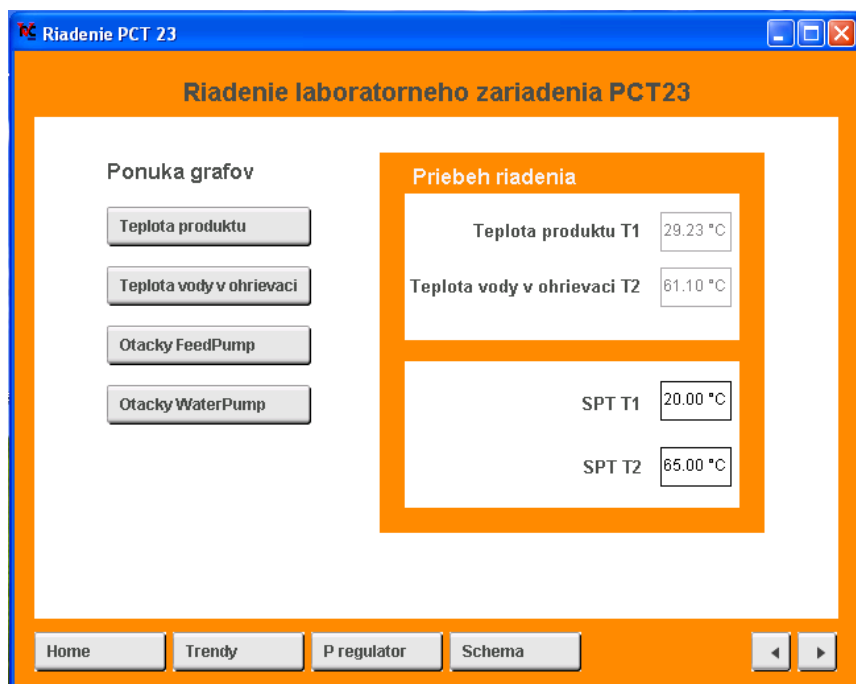


Obr. 38 Hlavná obrazovka vizualizácie

Po otvorení vizualizácie sa zobrazí hlavná obrazovka (obr.38). V tomto okne prostredníctvom tlačidiel „*Automaticky*“ a „*Manualny*“ môžeme prepnúť riadenie systému z automatického riadenia na manuálne. Pri automatickom režime je proces riadený regulátormi, pri manuálnom režime operátor zadáva hodnoty všetkých akčných

vstupov z riadiacej konzoly. V okne sa zobrazujú aktuálne hodnoty teploty produktu, teploty ohrevného média a hodnota otáčok čerpadiel. Z tohto miesta je možné nastaviť žiadané hodnoty riadených veličín, zosilnenie regulátora a tiež hodnotu otáčok čerpadla teplej vody. V dolnej časti obrazovky sú umiestnené tlačidlá, ktoré slúžia na prepínanie medzi obrazovkami vizualizácie.

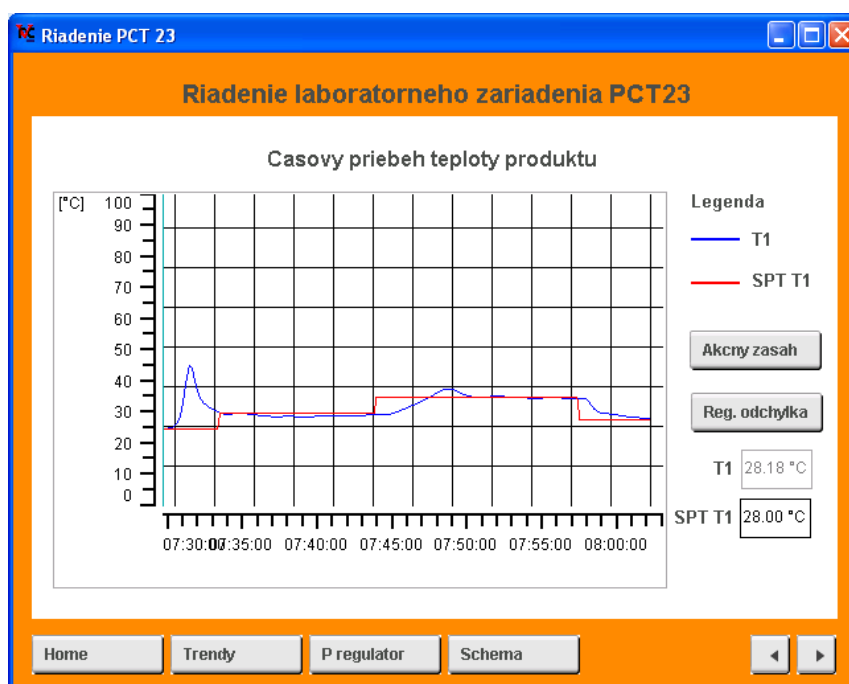
Na obrazovke „Trendy“ (obr.39) sa nachádza ponuka grafov. Vybrať si môžeme zobrazenie grafu teploty produktu, teploty vody v ohrievači a otáčok obidvoch čerpadiel. Stlačením tlačidla sa prepne obrazovka zobrazujúca príslušný grafický priebeh. Aj v tejto obrazovke vidíme aktuálne hodnoty riadených veličín a môžeme nastaviť ich žiadané hodnoty.



Obr. 39 Obrazovka pre ponuku grafov

Po zvolení grafu a stlačení príslušného tlačidla môžeme pozorovať časový priebeh veličiny. Výhodou tejto obrazovky je, že z grafického priebehu lepšie vidíme či sa veličina ustálila na žiadanej hodnote a aký bol jej priebeh počas riadenia. Grafické zobrazenie na rozdiel od číselných hodnôt viac hovorí aj o dynamike systému.

Obrazovka zobrazená na obr. 40 vykresľuje žiadanú hodnotu teploty produktu a aktuálnu hodnotu teploty. Tieto hodnoty môžeme pozorovať aj napravo od grafu v číselnej forme. Časový priebeh akčného zásahu PID regulátora a regulačnej odchýlky zobrazíme po stlačení tlačidiel v pravej časti okna. Prepínanie medzi obrazovkami sa uskutočňuje pomocou tlačidiel v dolnom menu. Šípkami v pravom rohu sa dostaneme na predchádzajúcu alebo nasledujúcu obrazovku.

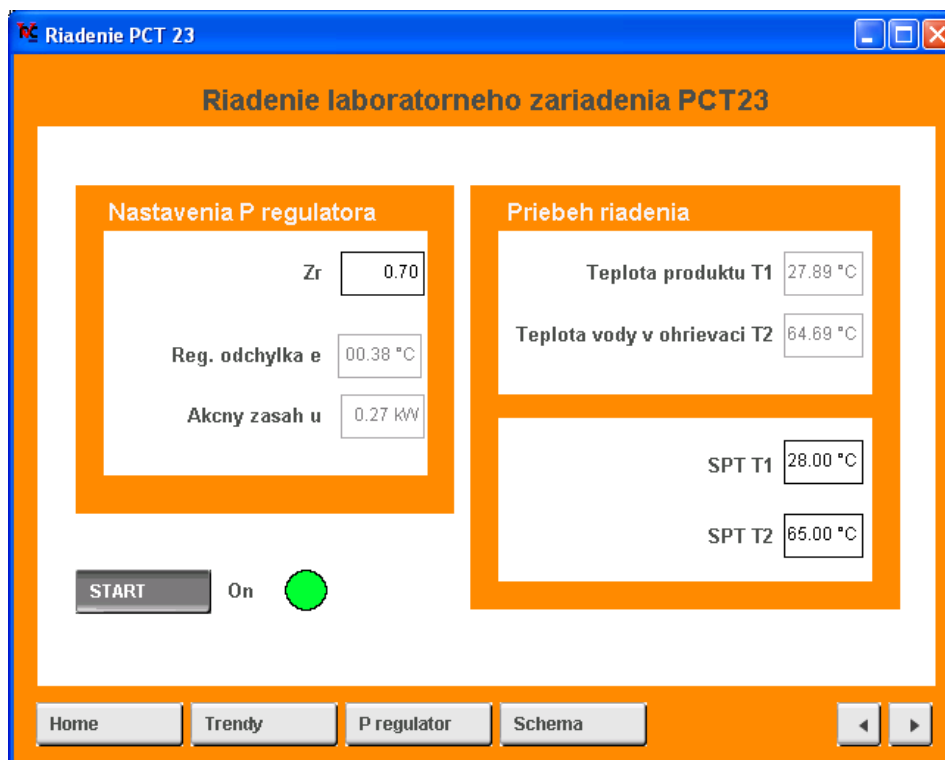


Obr. 40 Graf teploty produktu

Zosilnenie P regulátora nie je pevne dané, ale môžeme ho meniť podľa potrieb riadenia. Preto GUI obsahuje aj obrazovku(Obr.41) pre nastavenie tohto parametra. Po zmene zosilnenia v okne vidíme ako sa menia hodnoty teploty teplej vody aj teploty produktu. Regulačná odchýlka a akčný zásah regulátora sú tiež zobrazené. Riadenie môžeme z tejto obrazovky vypnúť alebo zapnúť. Stav riadenia je farebne odlišený. Zelená farba svieti vedľa tlačidla pri aktívnom riadení, červená pri neaktívnom.

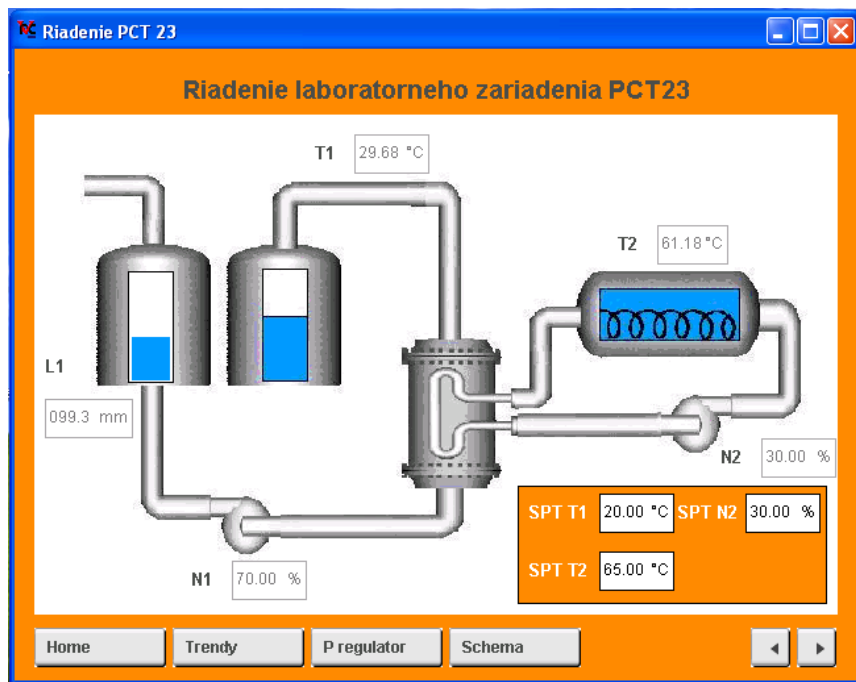
Na poslednej obrazovke(obr.42) vizualizácie sa nachádza schéma zapojenia zariadenia PCT23. Operátor tu opäť môže sledovať hodnoty všetkých veličín. Zobrazená výška hladiny v prvom zásobníku sa dynamicky mení s aktuálnou hodnotou.

Pri preplnení zásobníka môžeme rýchlo reagovať na vzniknuté nebezpečenstvo preliatia a predísť tak možným haváriám.



Obr. 41 Obrazovka pre nastavenie parametrov P regulátora

Pri vytváraní GUI sme sa snažili urobiť vizualizáciu čo možno najprehľadnejšou a najzrozumiteľnejšou. Pri všetkých typoch riadenia je totiž nevyhnutná prítomnosť operátora, ktorý sa pri zle vytvorenom vizualizačnom rozhraní môže dopustiť viacerých chýb a tie môžu viesť k haváriám a následnému prerušeniu výroby v prevádzke. V obrazovkách sme preto používali čo najviac grafických efektov, ktoré zvyšujú pozornosť a nezaplňovali ich zbytočnými prvkami, ktoré nie sú potrebné.



Obr. 42 Obrazovka schémy zapojenia zariadenia PCT23

Vytvorením vizualizácie sme vyriešili poslednú čiastkovú úlohu zadania, čím sme dospeli aj k riešeniu celkovej úlohy.

Záver

Hlavným cieľom diplomovej práce bolo navrhnuť riadenie laboratórneho zariadenia PCT23 v softvérovom prostredí B&R Automation Studio a vytvorené riadenie implementovať na reálny proces s využitím priemyselného riadiaceho systému B&R X20.

Celkovo práca pozostáva z dvoch hlavných kapitol. Obsahom prvej kapitoly sú získané teoretické vedomosti, ktoré sú rozčlenené podľa tém do podkapitol.

Výsledkom prvej podkapitoly je oboznámenie so softvérom B&R AS, jeho všeobecná charakteristika, popis základných funkcií balíka a programovacích jazykov, ktoré program podporuje.

V ďalšej časti sa nachádza popis prepojenia softvéru B&R AS so Simulinkom. Prepojenie umožňuje vytváranie zložitých a komplexných riadiacich systémov, ktoré by inak bolo časovo náročné implementovať. Spojenie týchto dvoch systémov sa využíva aj pri riešení praktickej úlohy.

Teoretickú časť diplomovej práce zakončuje opis laboratórneho zariadenia PCT23, ktoré predstavuje riadený systém. Sú charakterizované jednotlivé súčasti procesu, ich vzájomné zapojenia, špecifické vlastnosti a ochranné prvky. Spomenutá je aj riadiaca konzola, ktorá prepája zariadenie a PLC. Na konci podkapitoly sú uvedené dostupné akčné členy a senzory s príslušnými hodnotami snímacích rozsahov.

Praktické využitie poznatkov nadobudnutých v prvej časti práce je opísané v druhej kapitole. Úlohou bolo riadiť zariadenie PCT23 z prostredia AS. Zadaný problém je rozdelený na menšie čiastkové úlohy, ktorých vyriešením dospejeme ku komplexnému výsledku. Partikulárne časti predstavujú identifikáciu systému, návrhy regulátorov a vyhodnotenie realizovaného riadenia. Záver kapitoly tvorí opísanie vytvoreného vizualizačného prostredia pre riadenie procesu.

Výstupom z vypracovania diplomovej práce je projekt v AS, ktorý bol reálne implementovaný v riadiacom systéme X20. Riadenie zariadenia PCT23 sa vykonávalo z vizualizácie. Pri riešení úlohy mohli byť navrhnuté aj zložitejšie a možno aj kvalitnejšie typy riadenia. Cieľom práce však nebolo dosiahnuť čo najlepšie riadenie procesu, ale uskutočniť ho cez PLC B&R X20, čo sa nám podarilo úspešne uskutočniť.

Zoznam použitej literatúry

- [1] B&R. *Welcome to B&R Automation Software*. [on-line]. [cit. 2011- 4]. Dostupné na internete <http://www.br-automation.com/downloads_br_productcatalogue/BRP4440000000000000000032574/Welcome_in_AS.pdf>
- [2] B&R. *Automation Studio*. [on-line]. [cit. 2011-04]. Dostupné na internete <http://www.br-automation.com/cps/rde/xchg/br-productcatalogue/hs.xsl/cookies_allowed.htm?caller=products_5417_ENG_HTML.htm>
- [3] B&R. *Project management*. [on-line]. [cit. 2011-04]. Dostupné na internete <http://www.br-automation.com/cps/rde/xchg/br-productcatalogue/hs.xsl/products_151699_ENG_HTML.htm>
- [4] B&R. *Programming*. [on-line]. [cit. 2011-04]. Dostupné na internete <http://www.br-automation.com/cps/rde/xchg/br-productcatalogue/hs.xsl/cookies_allowed.htm?caller=products_151728_ENG_HTML.htm>
- [5] B&R. *Visualization*. [on-line]. [cit. 2011-04]. Dostupné na internete <http://www.br-automation.com/cps/rde/xchg/br-productcatalogue/hs.xsl/products_151762_ENG_HTML.htm>
- [6] Wikipedia. *WYSIWYG*. [on-line]. 03. 04. 2011[cit. 2011-04]. Dostupné na internete <<http://sk.wikipedia.org/wiki/WYSIWYG>>
- [7] WALLNER, Philipp. *B&R Automation Studio Target for Simulink*. [Manuál].
- [8] Armfield. *PCT 23Process Plnat Trainer*. [Manuál].
- [9] B&R. *Automation Studio - B&R Help Explorer*. [Manuál]. 2009.
- [10] FIKAR, Miroslav – MIKLEŠ, Ján. 1999. *Identifikácia systémov*. 1.vyd. Bratislava: Vydavateľstvo STU. 114 s. ISBN 80 – 227 – 1177 – 2.
- [11] BAKOŠOVÁ, M. – FIKAR, M. – Čirka, Ľ. 2003. *Základy automatizácie*. 1.vyd. Bratislava: Vydavateľstvo STU. 153 s. ISBN 80 – 227 – 1831 – 9.
- [12] FIKAR, M. – BAKOŠOVÁ, M. 2008. *Riadenie procesov*. 1.vyd. Bratislava: Vydavateľstvo STU. 193 s. ISBN 978 – 80 – 227 – 2841 – 6.
- [13] MIKLEŠ, J. – FIKAR, M. 2004. *Modelovanie, identifikácia a riadenie procesov 2: Identifikácia a optimálne riadenie*. 1.vyd. Bratislava: vydavateľstvo STU. 260 s. ISBN 80 – 227 – 2134 - 4

Prílohy

- Príloha A: Kód pre obmedzenie rozsahu vstupných a výstupných signálov riadiacej konzoly v jazyku Automation Basic
- Príloha B: Kód na prepočet premenných na fyzikálne jednotky v jazyku ANSI C
- Príloha C: Rebríkový diagram pre riadenie teploty vody v ohrievači P regulátorom
- Príloha D: M-file použitý pri identifikácii procesu 2.riadiacej slučky
- Príloha E: Vygenerovaný kód pre riadenie teploty produktu PID regulátorom
- Príloha F: CD médium – práca v elektronickej podobe

Príloha A

Kód pre obmedzenie rozsahu vstupných a výstupných signálov riadiacej konzoly v jazyku Automation Basic.

```
*****
* Program: NewProgram
* File: NewProgramCyclic.ab
* Author: Administrator
* Created: March 29, 2011
*****
* Implementation of program NewProgram
*****)

PROGRAM _CYCLIC
;Ak je hodnota vystupu vacsia ako 5V nastav 5V co zodpoveda integeru 16384
;Ak je hodnota vystupu mensia ako 0V nastav 0V co zodpoveda integeru 0
;to vykona pre vsetky 4 analogove vystupy
if(nic>16384) then
nic=16384
endif
if(nic<0) then
    nic=0
endif
;Ak je hodnota vystupu vacsia ako 1kW(odporucan hodnota vykonu spirali)
;nastav 1kW co zodpoveda integeru 8620
if(WaterHeater>8620) then
WaterHeater=8620
endif
if(WaterHeater<0) then
    WaterHeater=0
endif
if(FeedPump>16384) then
FeedPump=16384
endif
if(FeedPump<0) then
    FeedPump=0
endif
if(WaterPump>16384) then
WaterPump=16384
endif
if(WaterPump<0) then
    WaterPump=0
endif
(* TODO : Add your code here *)

END_PROGRAM
```


Príloha B

Kód na prepočet premenných na fyzikálne jednotky v jazyku ANSI C.

```
#include <asmath.h>          /* prototypes for AsMath-Library */

#ifdef _DEFAULT_INCLUDES
#include <AsDefault.h>
#endif

void _INIT prepoctyINIT( void )
{
    /* TODO: Add code here */
}

void _CYCLIC prepoctyCYCLIC( void )
{
    /* TODO: Add code here */
    /*Prepocet teploty na stupne Celzia(ohrievac)*/
    T2_stC=((REAL)T2*150.0)/16384.0;

    /*Prepocet teploty na stupne Celzia(produkt)*/
    T1_stC=((REAL)T1*150.0)/16384.0;

    /*Prepocet hladiny na mm*/
    L1_mm=((REAL)L1*250.0)/16384.0;

    /*Prepocet prikonu na kW*/
    WH_kW=((REAL)WaterHeater*2.0)/16384.0;

    /*Prepocet cerpadla na %(zo zasobnika)*/
    FP_per=((REAL)FeedPump*100.0)/16384.0;

    /*Prepocet cerpadla na %(z ohrievaca)*/
    WP_per=((REAL)WaterPump*100.0)/16384.0;

    /*prepocet  % (zo zasobnika) na INT*/
    FeedPump=(INT)ceil((SPT_FP*16384.0)/100.0);

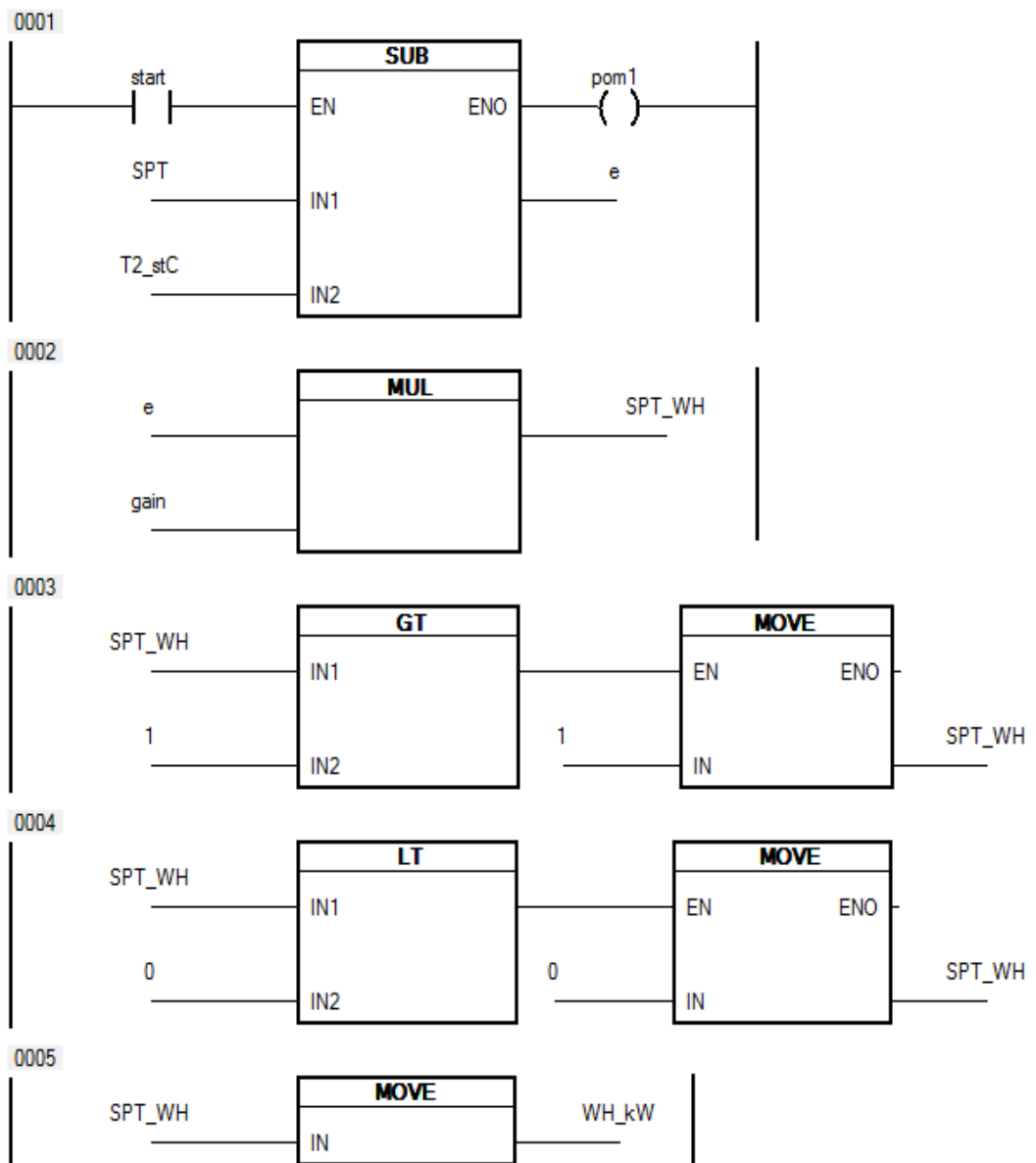
    /*prepocet  % (z ohrievaca) na INT*/
    WaterPump=(INT)ceil((SPT_WF*16384.0)/100.0);

    /*prepocet kW na INT*/
    WaterHeater=(INT)ceil((SPT_WH*16384.0)/2.0);

    /*pripadenie premennych k INT*/
}
```

Príloha C

Rebrikový diagram pre riadenie teploty vody v ohrievači P regulátorom.



Príloha D

M-file použitý pri identifikácii procesu 2.riadiacej slučky.

```
clear all
load identifikacia2.txt
t=identifikacia2(:,1);
FP=identifikacia2(:,2);
T1=identifikacia2(:,4);
T2=identifikacia2(:,6);
WP=identifikacia2(:,8);

figure
subplot(2,1,1),plot(t,FP,'b','LineWidth',2),
title('Skokové zmeny FP'),ylim([35 60]),
xlabel('Čas[s]'),ylabel('FP[%]'),grid,legend('FP')
subplot(2,1,2),plot(t,T1,'b'),title('Odozva T1 na skokové zmeny'),
xlabel('Čas[s]'),ylabel('T1[°C]'),grid on,legend('T1')
% figure
% plot(t,T2)
% figure
% plot(t,WP)

%filtrovanie udajov
b=fir1(30,0.005);
% yf=filter(b,1,T1);
%yf2=filtic(b,1,T1);

T1_f=filtfilt(b,1,T1);
T2_f=filtfilt(b,1,T2);

%rozdelenie FP na jednotlivé skoky
poz1=find(FP>=50);
poz2=find(FP>=55);
poz3=find(t>=2648);

FP_step1=FP(poz1(1)-1:poz2(1)-1);
t_step1=t(poz1(1)-1:poz2(1)-1);
t_step1=t_step1-t_step1(1);

% figure
% plot(t_step1,FP_step1)

FP_step2=FP(poz2(1)-1:poz3(1));
t_step2=t(poz2(1)-1:poz3(1));
t_step2=t_step2-t_step2(1);

% figure
% plot(t_step2,FP_step2)

FP_step3=FP(poz3(1)+65:end);
t_step3=t(poz3(1)+65:end);
t_step3=t_step3-t_step3(1);
```

```

%upravim vektory na rovnaku dlzku
FP_step1=FP_step1(1:length(FP_step2));
FP_step3=FP_step3(1:length(FP_step2));

%rozdelenie T1
T1_step1=T1_f(poz1(1)-1:poz2(1)-1);
T1_step2=T1_f(poz2(1)-1:poz3(1));
T1_step3=T1_f(poz3(1)+65:end);

T1_step1=T1_step1(1:length(FP_step2));
T1_step3=T1_step3(1:length(FP_step2));

%vykreslenie skokovych zmien a odoziev
figure
subplot(2,1,1),plot(t_step2,FP_step1,'r','LineWidth',3),axis([0 700 35 55])
title('1.skoková zmena FP'),xlabel('Čas[s]'),ylabel('FP[°]'),grid on,
legend('u1'),subplot(2,1,2),plot(t_step2,T1_step1,'r','LineWidth',1.5)
,
title('Odozva T1 na 1.skokovú
zmenu'),xlabel('Čas[s]'),ylabel('T1[°C]'),
grid on,legend('y1'),xlim([0 700])

figure
subplot(2,1,1),plot(t_step2,FP_step2,'b','LineWidth',3),axis([0 700 45 60])
title('2.skoková zmena FP'),xlabel('Čas[s]'),ylabel('FP[°]'),grid on,
legend('u2'),subplot(2,1,2),plot(t_step2,T1_step2,'b','LineWidth',1.5)
,
title('Odozva T1 na 2.skokovú
zmenu'),xlabel('Čas[s]'),ylabel('T1[°C]'),
grid on,legend('y2'),xlim([0 700])

figure
subplot(2,1,1),plot(t_step2,FP_step3,'g','LineWidth',3.5),axis([0 700 35 60])
title('3.skoková zmena FP'),xlabel('Čas[s]'),ylabel('FP[°]'),
grid on,legend('u3'),
subplot(2,1,2),plot(t_step2,T1_step3,'g','LineWidth',2),
title('Odozva T1 na 3.skokovú
zmenu'),xlabel('Čas[s]'),ylabel('T1[°C]'),
grid on,legend('y3'),xlim([0 700])

%normovanie vystupov na jednotkový skok
T1_n1=(T1_step1-T1_step1(1))/10;
T1_n2=(T1_step2-T1_step2(1))/5;
T1_n3=(T1_step3-T1_step3(1))/(-15);

figure
plot(t_step2,T1_n1,'r',t_step2,T1_n2,'b',t_step2,T1_n3,'g','LineWidth'
,1.5),
title('Normované prechodové charakteristiky'),
xlabel('Čas[s]'),ylabel('T1[°C]'),legend('y1','y2','y3'),grid
on,xlim([0 700])

figure
T1_n=(T1_n1+T1_n3)/2;
%T1_n=(T1_n1+T1_n2+T1_n3)/3;
plot(t_step2,T1_n,'LineWidth',1.5),
title('Stredná hodnota prechodovej charakteristiky'),

```

```

xlabel('Čas[s]'),ylabel('T1[°C]'),legend('Vysledna PCH'), grid on,
xlim([0 700]),ylim([-0.5 0.1])

D=30;
K=T1_n(end)+0.01;

% K =
%
% -0.4486

x=0.63*K;
i=find(T1_n<=x);

T=t_step2(i(1));

% T =
%
% 102.6000

T=60;
figure
plot(t_step2,T1_n,'g'),legend('y pov')
s=tf('s');
y_iden=(K*exp(-D*s))/(T*s+1)
hold on
step(y_iden,700),title('Porovnanie identifikovanej a pôvodnej PCH'),
xlabel('Čas'),ylabel('T1[°C]')
grid on,xlim([0 700]),ylim([-0.5 0.1])

%verifikacia
t_ver=ymodel(:,1);
y_ver=ymodel(:,2);

figure
plot(t,T1,t_ver,y_ver),legend('pov','iden')

```

Príloha E

Vygenerovaný kód pre riadenie teploty produktu PID regulátorom.

```
/*
#include "uloha2.h"
#include "uloha2_private.h"

/* Defines */

/* Data Types */

/***** GLOBAL DATA *****/
/* Definitions */

/* Block signals (auto storage) */
BlockIO_uloha2 uloha2_B;

/* Block states (auto storage) */
D_Work_uloha2 uloha2_DWork;

/* Declarations */

/***** FILE SCOPE DATA *****/

/***** FUNCTIONS *****/

/* Model step function */
void uloha2_step(void)
{
    /* local block i/o variables */
    real32_T rtb_DiscreteTransferFcn;
    real32_T rtb_Product;
    boolean_T rtb_LogicalOperator;
    uloha2_B.Input = SPTm_T1;
    uloha2_B.Input2 = T1m_stC;
    T1m_stC=T1_stC;
    SPTm_T1=SPT_T1;

    /* Sum: '<Root>/Sum' */
    uloha2_B.Sum = uloha2_B.Input - uloha2_B.Input2;

    /* DiscreteTransferFcn: '<Root>/Discrete Transfer Fcn' */
    rtb_DiscreteTransferFcn = (-1.602812042E+002F)*uloha2_B.Sum;
    rtb_DiscreteTransferFcn += 1.583161926E+002F*
        uloha2_DWork.DiscreteTransferFcn_DSTATE[0] + (-1.583186951E+002F)*
        uloha2_DWork.DiscreteTransferFcn_DSTATE[1];
    uloha2_B.Input1 = start1;

    /* Product: '<Root>/Product' */
    rtb_Product = (int32_T)uloha2_B.Input1 ? rtb_DiscreteTransferFcn : 0.0F;

    /* Logic: '<Root>/Logical Operator' */
    rtb_LogicalOperator = !uloha2_B.Input1;
    uloha2_B.Input3 = manual;
}
```

```

/* Sum: '<Root>/Sum3' incorporates:
 * Product: '<Root>/Product1'
 */
uloha2_B.Sum3 = rtb_Product + (real32_T)((int32_T)rtb_LogicalOperator ?
(uint16_T)uloha2_B.Input3 : 0U);
FPm_per = uloha2_B.Sum3;
e = uloha2_B.Sum;
if (FPm_per>=70.0)
{
    FPm_per=70.0;
}

if (FPm_per<=0.0)
{
    FPm_per=0.0;
}

SPT_FP=FPm_per;
/* DiscreteTransferFcn Block: '<Root>/Discrete Transfer Fcn' */
{
    real32_T xtmp = uloha2_B.Sum;
    xtmp += (1.0F)*uloha2_DWork.DiscreteTransferFcn_DSTATE[0] + (-0.0F)*
        uloha2_DWork.DiscreteTransferFcn_DSTATE[1];
    uloha2_DWork.DiscreteTransferFcn_DSTATE[1] =
        uloha2_DWork.DiscreteTransferFcn_DSTATE[0];
    uloha2_DWork.DiscreteTransferFcn_DSTATE[0] = xtmp;
}
}

/* Model initialize function */
void uloha2_initialize(boolean_T firstTime)
{
    (void)firstTime;

    /* (no initialization code required) */
}

/* Model terminate function */
void uloha2_terminate(void)
{
    /* (no terminate code required) */
}

```