



SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA CHEMICKÉJ A POTRAVINÁRSKEJ TECHNOLOGIE

KATEDRA INFORMATIZÁCIE A RIADENIA PROCESOV

**RIADENIE ZÁSOBNÍKOV KVAPALINY POMOCOU
RIADIACEHO SYSTÉMU FOXBORO**

Dňa: 20.5.2005

Miesto: Bratislava

Vedúci diplomovej práce: Ing. Magdaléna Ondrovičová

Vypracoval: Bc. Michal Hisira



SLOVENSKÁ TECHNICKÁ UNIVERZITA
Fakulta chemickej a potravinárskej technológie
Radlinského 9, 812 37 Bratislava

Katedra: **informatizácie a riadenia procesov**
Číslo: 5/KIRP/2005

Vec: **Zadanie diplomovej práce**

Meno a priezvisko študenta: **Bc. Michal Hisira**
Meno a priezvisko vedúcej diplomovej práce: **Ing. Magdaléna Ondrovičová**
Meno a priezvisko konzultanta diplomovej práce:
Názov diplomovej práce:

Riadenie zásobníkov kvapaliny pomocou riadiaceho systému FOXBORO

Termín odovzdania diplomovej práce: **20. mája 2005**
Diplomová práca sa odovzdáva v 3 exemplároch vedúcemu katedry.

Bratislava **1. februára 2005**

Doc. Dr. Ing. Miroslav Fikar
vedúci katedry

Prof. Ing. Dušan Bakoš, DrSc.
dekan

Pod'akovanie:

Chcel by som pod'akovať vedúcemu diplomovej práce Ing. Magdaléne Ondrovičovej za pripomienky, cenné rady, ochotu a odborné vedenie. Zároveň by som chcel pod'akovať Ing. Branislavovi Smělíkovi za cenné rady pri problémoch ktoré sa vyskytli pri diplomovej práci.

Abstrakt

Diplomová práca sa zaoberá návrhom riadenia zásobníkov kvapaliny v prostredí riadiaceho systému FOXBORO. Zásobníky kvapaliny sú v práci naprogramované pomocou diferenciálnych rovníc prvého rádu v integrovanom riadiacom systéme. Riadenie výšky hladiny v druhom zásobníku je založené na ovplyvňovaní prietoku. Žiadaná hodnota výšky hladiny a parametre regulátora sú posielané do regulátora pomocou užívateľskej obrazovky vo FoxView. Táto diplomová práca poskytuje základné informácie o riadiacom programe FOXBORO a užívateľskom prostredí FoxDraw.

Abstract

This diploma theses deals with the control of fluid tanks in enviroment of control system FOXBORO. In this work are fluid tanks programmed in the Integrated Control Configurator with the help of the first order differential equation. Tanks level control is based on flow influence. Tank level setpoint and controller parameters are send to controller through user display in FoxView. This diploma theses provides basic information about control system FOXBORO and users setting in FoxDraw.

Obsah

ÚVOD	1
1 VŠEOBECNÝ OPIS RIADIACEHO SYSTÉMU FOXBORO	3
2 KOMPOUNDY	6
2.1 Procesné alarmy komponentu a bloku	7
2.2 Fázovanie komponentu a bloku.....	7
2.3 Vlastnosti komponentu	7
2.4 Sprístupnenie komponentu.....	7
2.5 Staničný komponent/blok	8
3 BLOKY	9
3.1 Priradenie typu bloku	9
3.2 Parametre bloku	9
3.3 Prístup do bloku.....	11
3.4 Typy funkčných blokov	11
4 INTEGROVANÝ RIADIACI KONFIGURÁTOR (ICC).....	12
5 FOXDRAW	20
5.1 Vkládanie objektov do obrazovky pre užívateľa.....	21
5.2 Konfigurácia objektov v obrazovke užívateľa	22
6 VYUŽITIE ICC NA RIADENIE ZÁSOBNÍKOV	25
6.1 Blok CALCA.....	26
6.2 Blok LLAG	26
6.3 Blok PIDA	27
6.4 Blok SWCH.....	30
7 ROZHRANIE VYTVORENÉ VO FOXDRAW	31

7.1	Popis rozhrania	31
7.2	Postup tvorby a konfigurácie základných objektov	33
7.3	Riadenie zásobníkov	41
ZÁVER		42
LITERATÚRA		43
PRÍLOHA 1		44
PRÍLOHA 2		52

Úvod

Riadiace systémy zabezpečujú na základe vhodne zvolených algoritmov vychádzajúcich z konkrétnych požiadaviek danej technológie výroby optimalizáciu, automatizáciu, a riadenie daného výrobného procesu.

Optimalizácia je z matematického hľadiska hľadanie extrémov kriteriálnej funkcie. Jej cieľom je nájsť najvhodnejšie riešenie daného problému.

Automatizáciu si môžeme predstaviť ako zavádzanie, zavedenie a využívanie automatizovaných systémov riadenia v technológiách.

Riadenie sa uskutočňuje pomocou riadiaceho počítača, ktorý vykonáva výpočet riadiaceho signálu pomocou algoritmu riadenia na základe nameraných hodnôt a zabezpečuje prenos signálov riadenia cez prevodníky na ovládacie prvky, ktoré umožňujú vykonávať zmeny riadeného systému. Tieto zmeny sa uskutočňujú za účelom kompenzácie porúch, udržania rovnovážneho stavu riadeného objektu, zabezpečenia optimálnych pracovných podmienok, maximalizácie produkcie, minimalizácie spotreby energie surovín, zabezpečenie najmenšieho možného škodlivého vplyvu na životné prostredie, a tým zabezpečenie maximalizácie zisku.

Počítač okrem riadenia procesu vykonáva aj na zobrazenie súčasného stavu systému pomocou užívateľského rozhrania, najčastejšie ide o monitor, kde sú zobrazené najdôležitejšie informácie o okamžitých hodnotách technologických veličín s možnosťou zobrazenia ich grafického priebehu v čase, pomocou ktorých dokáže operátor zhodnotiť a prípadne odborným zásahom upraviť stav systému.

V súčasnej dobe je na celosvetovom trhu množstvo riadiacich systémov. Každý z nich je určitým spôsobom výnimočný, má vytvorenú inú filozofiu a algoritmy riadenia. Medzi najznámejšie riadiace systémy patria riadiace systémy Genie, Foxboro, Simatic, Yokogawa, Allen Bradley, Mitsubishi.

Cieľom tejto práce je navrhnúť riadenie zásobníkov kvapaliny pomocou programovania v riadiacom programe FOXBORO s použitím grafického editoru FoxDraw, a integrovaného riadiaceho konfiguratúra ICC.

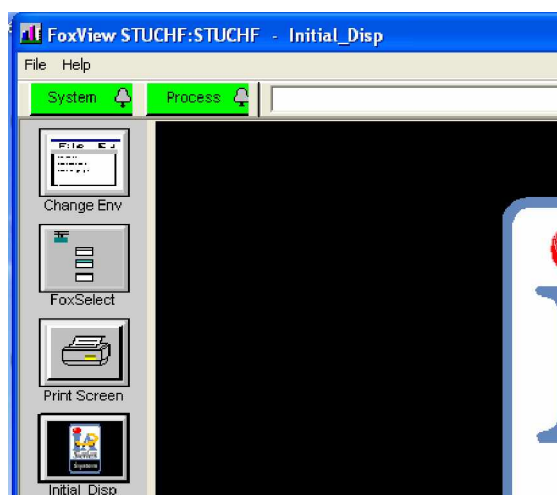
V prvej kapitole je napísaný stručný opis riadiaceho systému FOXBORO. Druhá kapitola sa venuje compoundu, jeho vlastnostiam a parametrom. Tretia kapitola opisuje bloky, ich základné parametre, typy a hodnoty. V štvrtej kapitole je opísaný integrovaný riadiaci konfiguratör, postup vkladania compoundov a blokov a možnosť úpravy ich parametrov. Piata kapitola v stručnosti opisuje grafický zobrazovací editor FoxDraw, pomocou ktorého sa vytvára prístup užívateľa ku informáciám o riadených prebiehajúcich procesoch. Šiesta kapitola opisuje bloky použité v compounde spolu s logikou, ktorá bola použitá pri ich prepájaní. V siedmej kapitole je opísané grafické prostredie vytvorené v grafickom editore FoxDraw spolu s postupom ako nadefinovať podobné prostredie.

1 Všeobecný opis riadiaceho systému FOXBORO

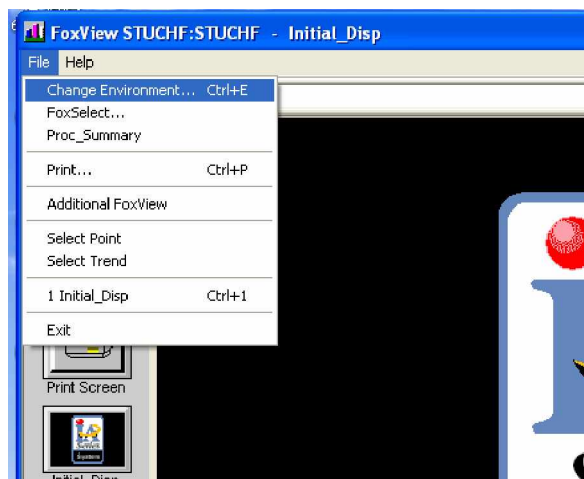
Program I/A Series Foxboro sa inštaluje pod operačným systémom Unix, poprípade pod Windows XP. Pri systéme Windows je pri inštalácii nutná anglická verzia tohto operačného systému, rovnako je pre správnu činnosť potrebné nastavenie anglickej klávesnice.

Na obr. 1.1 je úvodná obrazovka, ktorá sa zobrazí po spustení počítača a načítaní operačného systému spolu s riadiacim systémom.

Prostredie riadiaceho systému Foxboro je z dôvodu zabezpečenia pred neoprávnenými zásahmi rozdelené na viacero úrovní, pričom pre každú z nich sú vymedzené iné prístupové práva. Prechod medzi jednotlivými úrovňami je možný pomocou tlačidla **Change_Environment** na lište vľavo (obr. 1.1), pomocou hlavnej ponuky okna **File – Change_Environment**, alebo klávesovou skratkou Ctrl+E (obr. 1.2).



Obr. 1.1



Obr. 1.2

Najvyššia úroveň oprávnení je pod možnosťou Softw_Eng. Právo na programovanie a zmeny vykonávané v riadiacom systéme v prostredí integrovaného riadiaceho konfiguratéra (Integrated Control Configurator - ICC) vytváraním compoundov, vkladáním blokov, úpravou parametrov v blokoch a ich vzájomným prepájaním je prístupné od úrovne Process_Eng.

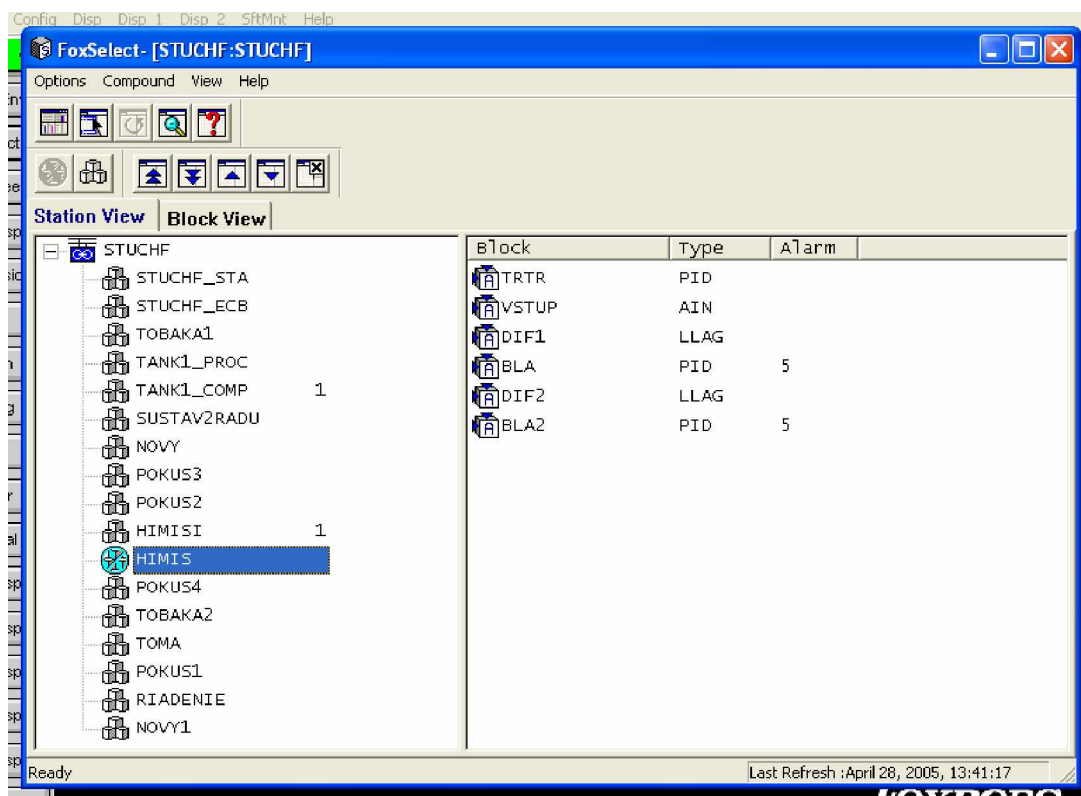
Na každej obrazovke procesu (napríklad obr. 1.1) je tlačidlo **FoxSelect**, ktoré umožňuje zobrazenie zoznamu compoundov a blokov spolu so zobrazením ich stavov.

FoxSelect nahradzuje obrazovku výberu v Display Manager. FoxSelect sa používa na:

- vnesenie podrobného zobrazenia (bloku, compoundu alebo staničného bloku) do FoxView
- zapnutie a vypnutie compoundov a ich priradených blokov
- rozšírenie zoznamu staníc siete a compoundov ktoré odhaľuje štruktúru hierarchie riadiacej databázy
- prehľad zoznamu blokov vo vnútri všetkých pripojených staníc a triedenie zoznamu podľa rôznych kritérií





Obrazovka okna FoxSelect má dve časti (obr. 1.3).

- Station View: zobrazenie stanice a compoundov podľa hierarchie v ľavom paneli a zoznamu blokov vyznačeného compoundu na pravom paneli
- Block View: triedený zoznam blokov vo vnútri všetkých pripojených staníc





Obr. 1.3 Obrázovka FoxSelect

V StationView sú zobrazené stavy compoundov a stavy blokov (obr. 1.4).

Stavy compoundov		Stav bloku	
	Compound je zapnutý (ON)		Automatický
	Compound je vypnutý (OFF)		Manuálny
			Žiaden Auto/Manuál parameter
			Nedefinovaný
			Chyba

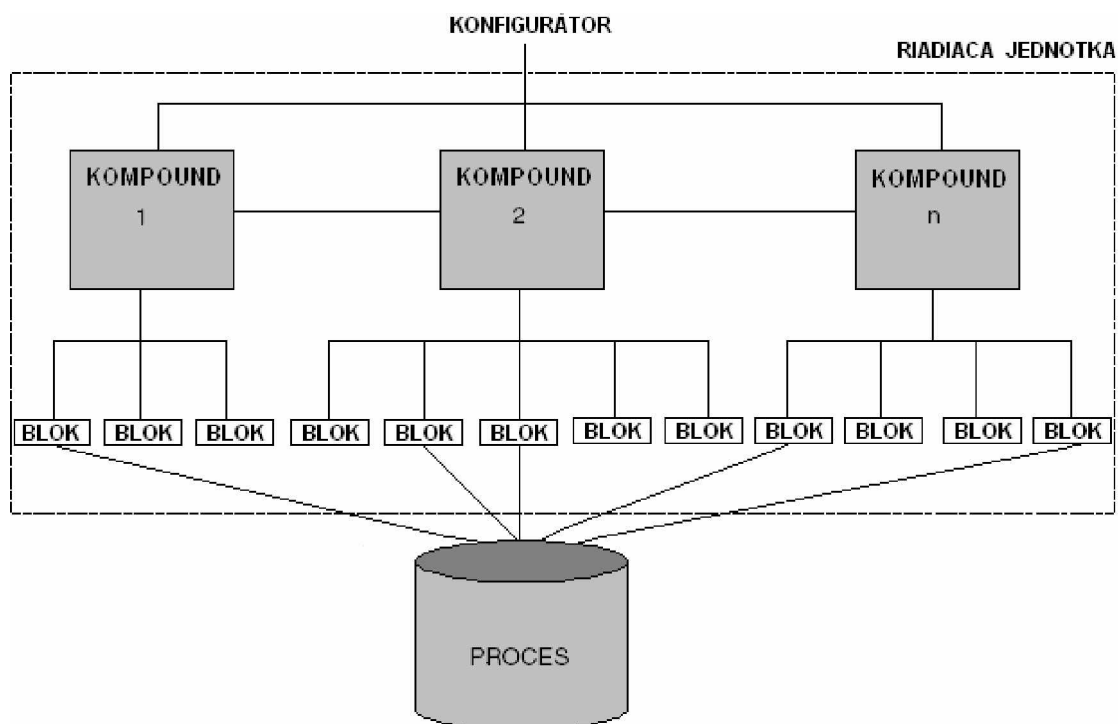
Obr. 1.4 Stavy compoundu a bloku

Compound sa dá zapnúť (tlačidlom ) alebo vypnúť (tlačidlom  – spolu s vypnutím sa zastaví vykonávanie riadiacej logiky bloku) v obrazovke FoxSelect a zároveň sa zmení ikonka pri compounde a obnoví sa obrazovka.

2 Kompondy

Komunikácia medzi zariadeniami a riadenie procesov pomocou systému I/A Series FOXBORO je založené na princípe kompondov a blokov [3].

Kompond je logický súbor blokov, ktoré vykonávajú riadiacu stratégiu. Blok je súčasťou súboru algoritmov, ktoré plnia riadiace úlohy v rámci štruktúry kompondu. Vzťah medzi kompondom a blokmi je vidieť na obr. 2.1.



Obr. 2.1 : Vzťah medzi kompondmi a blokmi

Kompond predstavuje základ pre integráciu:

- Spojitého riadenia,
- Reťazovej logiky,
- Sekvenčného riadenia.

V systéme môže byť blok v ktoromkoľvek komponde spojený s iným blokom v komponde systému. Kompondy a bloky obsahujú parametre, ktoré môžu mať hodnoty rôznych typov (reálne číslo, boolean číslo, celé číslo, alebo reťazec znakov).

Pre kompondy platia nasledujúce pravidlá:

- V rovnakej stanici môžu súčasne pracovať viaceré kompondy,
- Kompond nemôže prejsť hranice stanice,
- Bloky v rôznych kompondoch môžu byť navzájom prepojené
- Každý alarm musí mať jedinečné meno.

2.1 Procesné alarmy kompondu a bloku

Alarmy a správy o alarmoch sú vytvárané po nastavení v blokoch, ktoré ich podporujú. Alarmy majú 5 úrovní priority (1 až 5) (0 = neprítomnosť alarmu, 1 = najvyššia priorita), slúžia na informovanie užívateľa a umožňujú mu sústrediť sa na najdôležitejšie alarmy zariadenia. Alarmy sú sumarizované v nezávislých súhrnoch alarmov pre každý kompond. Súhrn obsahuje prioritu najvyššieho súčasného alarmu v komponde. Alarmy sa spúšťajú pomocou blokov vo vnútri kompondu. Správy o alarme sú potom poslané do skupiny staníc, alebo do aplikácií (do pracovnej stanice, do histórie, do tlačiarne) v závislosti od nastavenia alarmovej skupiny.

2.2 Fázovanie kompondu a bloku

Fázovanie dovoľuje posunúť alebo zadržať čas štartu kompondu alebo bloku. Užívateľ môže nastaviť fázu celočíselnou hodnotou, od ktorej závisí priradená perióda.

2.3 Vlastnosti kompondu

Kompond má nasledujúce hodnoty:

- **Meno** Definované užívateľom, jedinečné pre systém, najviac 12 znakov. Súčasťou mena môžu byť číslice (0 až 9), veľké písmená abecedy (A až Z) a (_).
- **Popis** 32-znakové pole určené na identifikáciu.
- **On/Off** Parameter, ktorý umožňuje alebo znemožňuje realizáciu jednotlivých blokov vo vnútri kompondu, (0 = Off, 1 = On).

2.4 Sprístupnenie kompondu

Kompondy aj bloky obsahujú súbor parametrov. Na sprístupnenie parametra kompondu pre jeho použitie inou aplikáciou je treba použiť nasledujúcu syntax:

Kompond.Parameter

v ktorej:

- Compound je názov compoundu (maximálne 12 znakové meno),
- Parameter je názov parametra v compounde (maximálne 6 znakové meno).

2.5 Staničný compound/blok

Staničný compound obsahuje jeden staničný blok pre každú stanicu v systéme. Inštaluje sa automaticky do stanice pri jej načítaní. Tento blok poskytuje ukladanie pamäťových dát pre staničné systémové funkcie. Staničný blok poskytuje informácie o možnostiach stanice. Pre všetky stanice sú staničné bloky totožné. Staničné bloky sú použité pre každý z nasledujúcich typov staníc:

- Aplikácia pracovnej stanice (AWs)
- Ústredná jednotka pracovnej stanice (WPs)
- Aplikácia pracovnej stanice (APs)
- Riadenie pracovnej stanice (CPs)
- Integrátory a Gateways
- Riadiaca stanica.

Obmedzenia platné pre staničný blok a staničný compound:

- Staničný compound nemôže byť zmazaný alebo vypnutý,
- Staničný blok nemôže byť zmazaný,
- Bloky vytvorené užívateľom nemôžu byť pridané do staničného compoundu.

2.5.1 Bezpečnosť databázy

Staničný blok obsahuje v nastavení konfigurácie parameter bezpečnosti CFGOPT. Zmeny databázy bloku alebo ECB sú zablokované ak je parameter CFGOPT nastavený pravda (true).

2.5.2 Čas a dátum

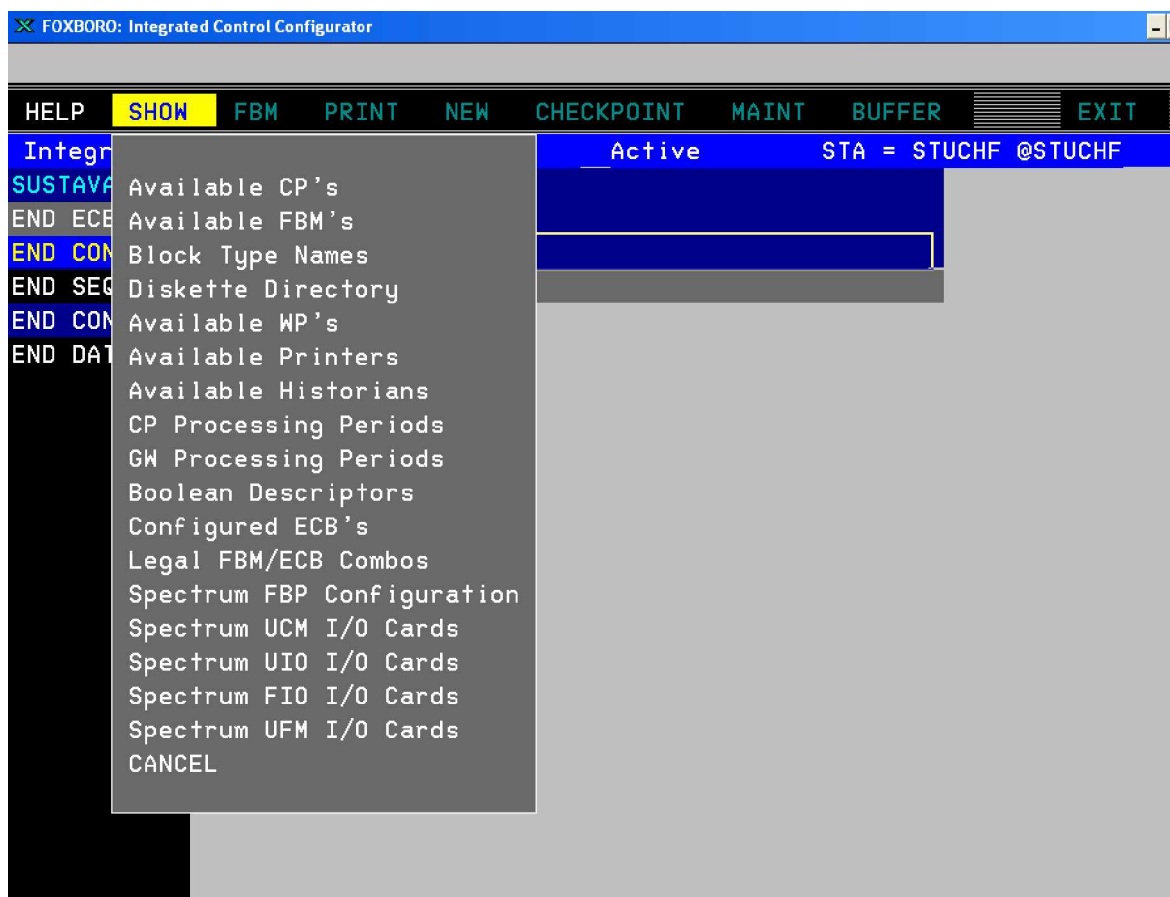
Staničný blok obsahuje päť parametrov, (YEAR, MONTH, DAY, HOUR, SECOND) ktoré slúžia užívateľovi na prístup do času systému. Tieto parametre sú aktualizované pomocou staničného software každých tridsať sekúnd.

3 BLOKY

Blok má jeden alebo viac vstupov (výstupov) a uskutočňuje preddefinované funkcie procesu, ktoré boli vopred stanovené algoritmom. Typy funkčných blokov sú: spojitý, sekvenčný a s reťazovou logikou. Tieto typy blokov sa dajú podľa potrieb kombinovať.

3.1 Priradenie typu bloku

Priradenie bloku (TYPE používané pri definovaní parametrov (Kapitola 3.2)) je možné definovať v integrovanom riadiacom konfiguratore dvoma spôsobmi. Prvý z nich je, že sa typ bloku napíše pomocou klávesnice (napríklad AIN). Druhý spôsob je výber typu bloku zo zoznamu, ktorý sa zobrazí vybraním funkcie SHOW-Block_Type_Names (obr. 3.1).



Obr. 3.1 Ponuka funkcie SHOW

3.2 Parametre bloku

Každý blok obsahuje parametre slúžiace ako vstupy a výstupy jeho vlastných funkcií.

Všeobecné parametre

Všetky bloky, (s výnimkou spomenutých) majú nasledujúce všeobecné parametre:

NAME	Meno je reťazec 12 znakov určený užívateľom, musí byť jedinečný vo vnútri komponentu. Používa sa k prístupu do bloku a jeho parametrov.
TYPE	Typ je meno určené systémom (do 6 znakov), identifikuje algoritmus riadiacej funkcie.
DESCRP	Opis je reťazec definovaný užívateľom (do 32 znakov), opisuje funkciu bloku.
PERIOD	Periódka je vstupný nespojitelný parameter, stanovuje časovú bázu spracovania bloku. <i>Výnimka: bloky dátových premenných (napríklad BOOL, LONG, ..).</i>
MA	Manuál/Auto je pripojiteľný Boolean vstup, riadi operačný stav Manuál/Auto (0 = Manual, 1 = Auto). <i>Výnimka: bloky dátových premenných a BLNALM.</i>
INITMA	Spúšťanie Manuál/Auto označuje žiadaný stav vstupu MA počas inicializácie (0 = Manuál; 1 = Auto; 2 = Bez zmeny, okrem reštartu, používa stav určený v súbore kontrolného bodu). <i>Výnimka: bloky dátových premenných a BLNALM.</i>
LOOPID	Identifikátor slučky je konfigurovateľný reťazec (do 32 znakov), určuje obvod alebo proces s ktorým je blok spojený. <i>Výnimka: bloky dátových premenných.</i>
OWNER	Owner je nastaviteľný reťazec (do 32 ASCII znakov), používa sa na stanovenie riadiacich blokov pre aplikácie. Pokusy poslať OWNER sú úspešné, iba ak je súčasná hodnota OWNER nulový reťazec, reťazec prázdneho poľa, alebo je identická ako hodnota poslanej žiadosti. Inak je žiadosť odmietnutá s chybou „uzamknutý prístup“ (LOCKED_ACCESS). OWNER môže byť vynulovaný aplikáciou alebo nastavením nulovej hodnoty, hodnota je vždy akceptovaná. <i>Výnimka: bloky dátových premenných, AMSSEC, DSI, EVENT, FBTUNE a FFTUNE.</i>
LOCKRQ	Žiadosť uzamknutia (Lock Request) je boolovský vstup, nastavený ako pravda (true) alebo nepravda (false) iba pomocou prepínacieho kľúča LOCK U/L na obrazovke stanice. Ak je LOCKRQ nastavený pravda (true) a sprevádza identifikátor pracovnej stanice, nastavená požiadavka vstupuje do parametra bloku LOCKID. LOCKRQ môže byť poslaný nepravda (false) stanicou na základe prijatia nového LOCKRQ a nového identifikátora

pracovnej stanice zapísaného do LOCKID. *Výnimka: AMSSEC, MSG, PLC a DCI.*

LOCKID Označenie zamknutia je reťazec označujúci pracovnú stanicu, ktorá má uzamknutý prístup do bloku nastavením LOCKRQ. *Výnimka: AMSSEC, MSG, PLC a DCI výstupné bloky.*

Editovanie parametrov

Parametre sa dajú upravovať v integrovanom riadiacom konfigurátore.

3.3 Prístup do bloku

K prístupu do hodnoty parametra bloku mimo príslušného compoundu (obr. 3.2) treba napísať celú cestu:

Kompound:Blok.Parameter



Obr. 3.2 Príklad použitia prístupu do hodnoty definovaný v FoxDraw

Toto pravidlo sa používa medzi blokmi v rozličných compoundoch (aj pri prepojení do inej stanice).

Na pripojenie do hodnoty parametra z iného bloku vo vnútri toho istého compoundu alebo do hodnoty parametra z vnútra toho istého bloku sa môže v ICC použiť aj cesta (obr. 3.3):

:Blok.Parameter

HIMISI	Block Definition:
END ECB***	Name: REG1
REG0	Type: PIDA
REG1	DESCRP <input type="text"/>
POCET1	PERIOD 1
POCET	PHASE 0
SWITCHB	LOOPID
ZASOB_1	MEAS :POCET1.R001

Obr. 3.3 Pripojenie parametra RO01 v ICC

3.4 Typy funkčných blokov

Zoznam uvedený v Prílohe 1 udáva stručný prehľad o všetkých funkčných typoch blokov z celého integrovaného riadiaceho software.

4 Integrovaný riadiaci konfigurátor (ICC)

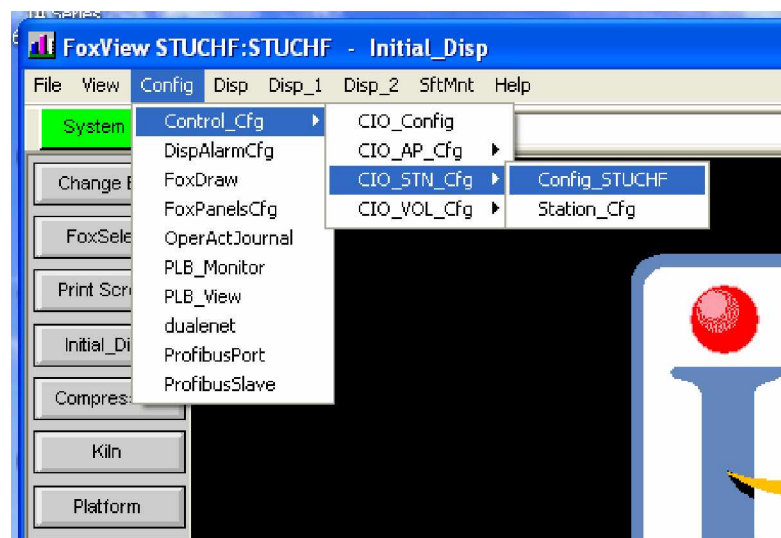
Integrovaný riadiaci konfigurátor (ICC) je prístupný pre použitie od úrovne **Process_Eng** (Kapitola 1), používa sa na vytváranie riadiacich algoritmov. Kompoundy a bloky majú vopred nadstavené hodnoty, preto je možné ich vložiť pred nastavením jednotlivých parametrov.

ICC poskytuje možnosti na vytváranie compoundov a blokov spolu s kompletnou ponukou funkcií na úpravu.

Možnosti úprav:

- tvorba a spájanie spojitých, sekvenčných blokov a blokov s reťazovou logikou v štruktúre jedného compoundu
- tvorba skupín a spájanie compoundov
- zmena, kopírovanie a odstraňovanie compoundov a blokov
- nastavenie a zmena Fieldbus modulov
- priradenie riadiacich schém staniciam v prostredí spracovania
- tvorba a udržiavanie knižníc compoundov
- priame pridávanie ECB (Equipment Control Block) blokov zariadenia.

Na spustenie ICC je potrebné zo stĺpca menu **Config** zvoliť položku **Control_Cfg** – **CIO_STN_Cfg** a položku, ktorá obsahuje **Config_názov_stanice** (obr. 4.1).



Obr. 4.1 Spustenie integrovaného riadiaceho konfigurátora

Na obr. 4.2 je otvorené okno ICC, ktoré obsahuje na hornej lište hlavné menu funkcií. V hlavnom menu sa nachádza položka **EXIT**, ktorá správne vypne ICC s uložením zmien.

Na ľavej strane obrazovky je zobrazený zoznam compoundov ktoré sú v stanici vytvorené (STUCHF_STA je staničný blok).

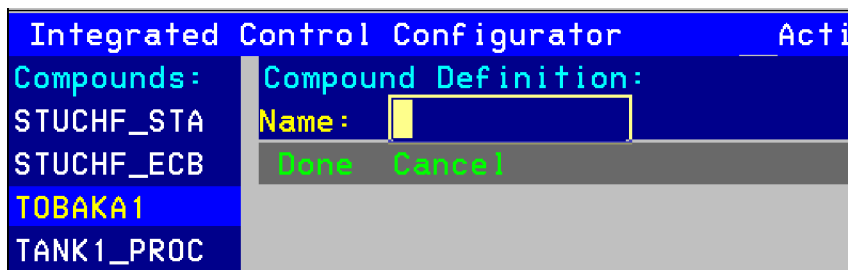
Vloženie nového compoundu

Ak chceme vložiť do stanice nový compound, označíme compound pred ktorý chceme nový compound vložiť (obr. 4.2 – nový compound bude vložený pred TOBAKA1), použijeme funkciu **Insert_New_Compound** z menu **Compound_Functions**.



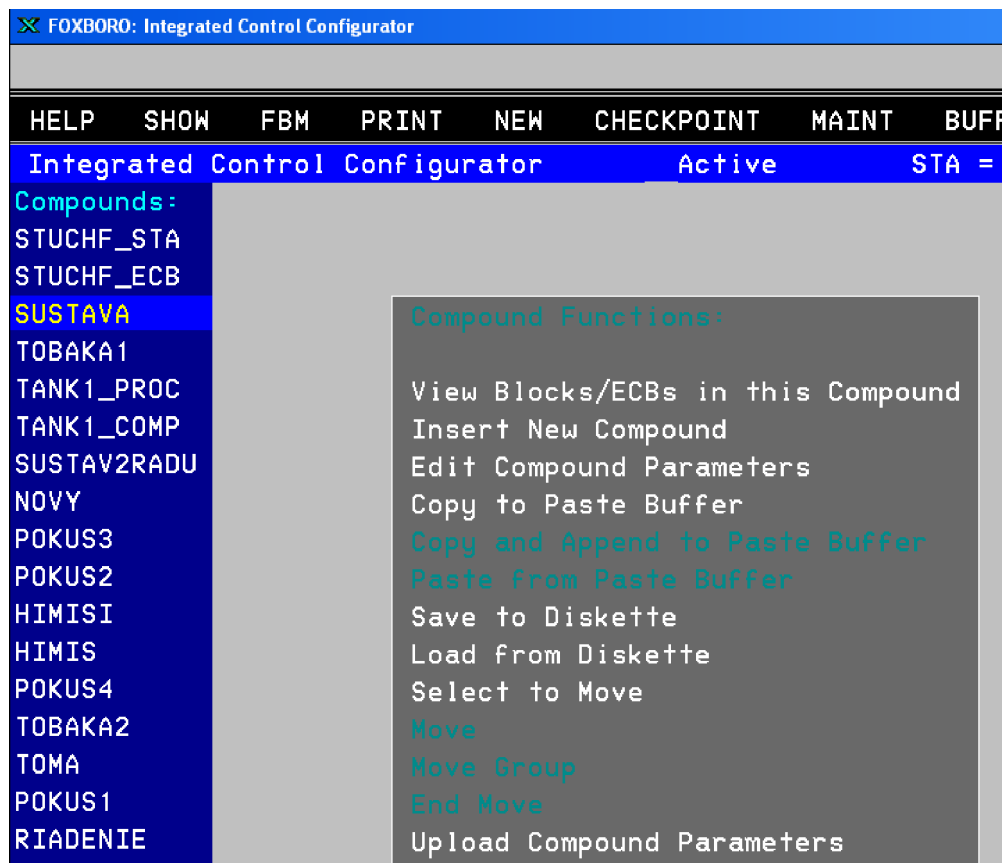
Obr. 4.2 Základné menu ICC

Zobrazí sa okno, v ktorom vložíme názov (obr. 4.3), potvrdíme klávesou Enter.



Obr. 4.3 Definovanie názvu compoundu

Ak je meno prijaté, okno s definíciou compoundu zmizne a compound je vytvorený s predvoleným nastavením hodnôt. Novo vložený compound sa objaví podsvietený v zozname compoundov (obr. 4.4).



Obr. 4.4 Nový vložený compound

Ak meno nie je jedinečné v celej stanici, je hlásená nutnosť zmeny s možnosťou odvolania požiadavky vloženia mena.

Kompondy v zozname (obr. 4.4) je možné presúvať pomocou funkcií **Move** a **Move_Group** v spojení so **Select_to_Move** a **End_Move**.

Úprava parametrov kompondu

Na zmenenie parametrov kompondu treba vybrať z menu **Compound_Functions** položku **Edit_Compound_Parameters** (obr. 4.4). Pomocou nej sa vytvorí kópia aktuálneho zoznamu parametrov vybraného kompondu, v ktorej je možné upravovať hodnoty parametrov. Po potvrdení sú upravené hodnoty uložené.

Delete

Funkcia Delete sa používa na:

- odstránenie kompondu alebo bloku z databázy a obrazovky
- obnovenie pripojenia kompondu alebo bloku

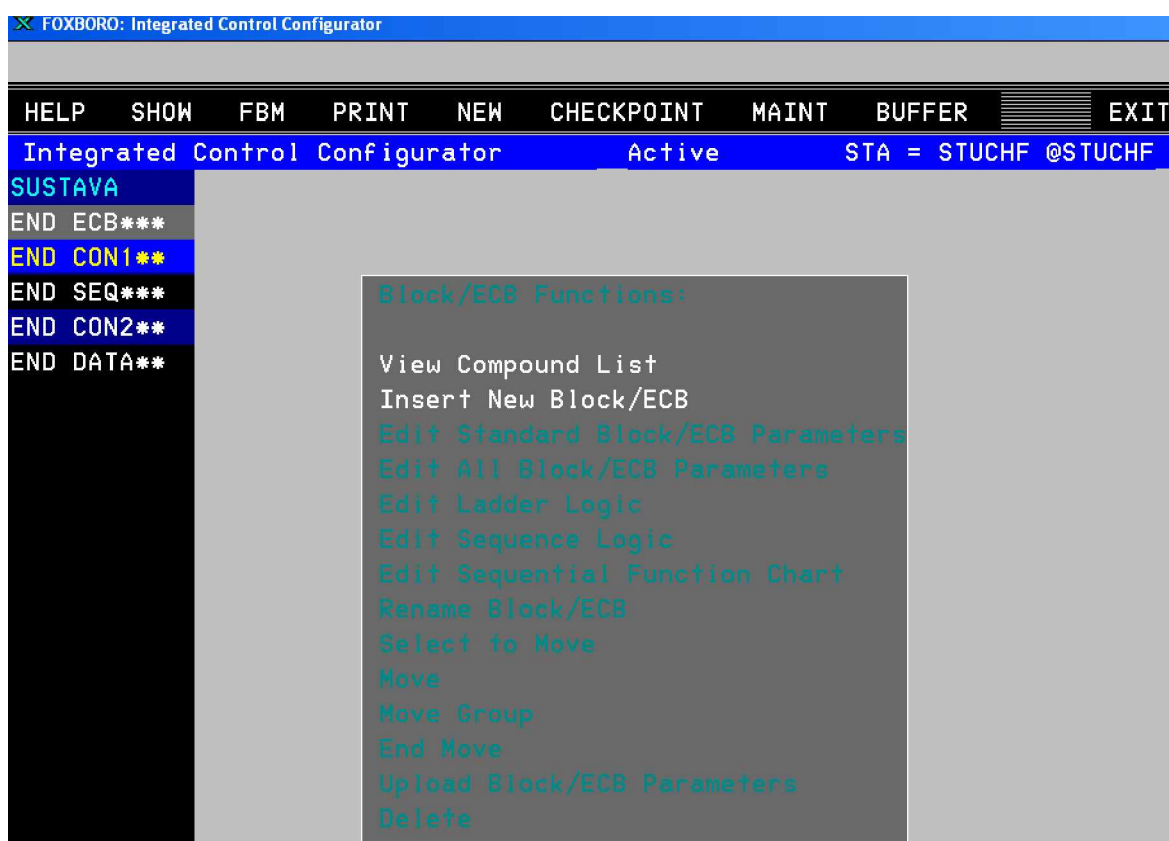
Zvolením funkcie Delete sa zobrazí dialógové okno ktoré obsahuje tri možnosti:

- Delete – odstráni kompond, blok alebo ECB. Nemôže sa ním odstrániť staničný kompond, staničný blok, ECB kompond alebo primárny ECB.

- Delete & Undelete – presunie vybraný compound alebo blok dovnútra a von z dočasného zásobníka na odstránenie. Pozícia compoundu alebo bloku sa v databáze nemení.
- Cancel – preruší operáciu a vráti predchádzajúce menu

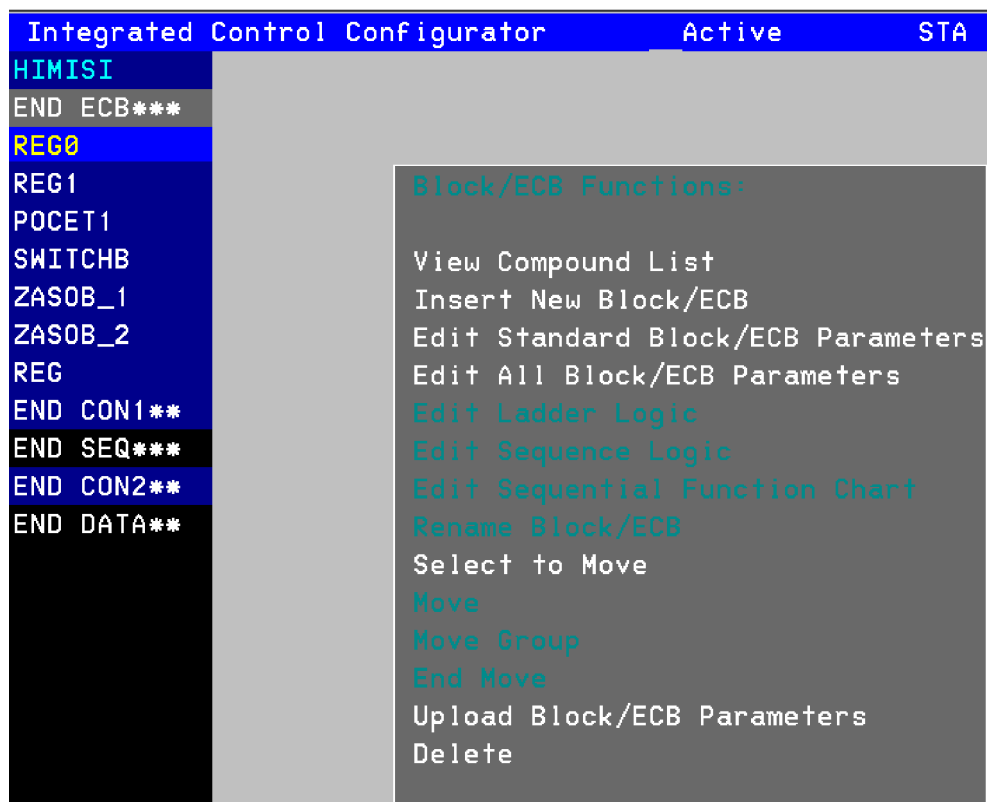
Obrazovka funkcií bloku

Po výbere funkcie **View_Blocks/ECB_in_this_Compound** v ponuke **Compound_Functions** sa zobrazí v ľavej časti ponuky zoznam blokov (ECB), ktoré sa nachádzajú v kompounde a ponuka **Compound_Functions** sa zmení za ponuku **Block/ECB_Functions** (obr. 4.5).



Obr. 4.5 Základná obrazovka funkcií bloku a ECB

Ak už zvolený compound obsahuje nadefinované bloky (ECB) ponuka v strede okna je zmenená (obr. 4.6) a zoznam obsahuje mená všetkých blokov v kompounde.



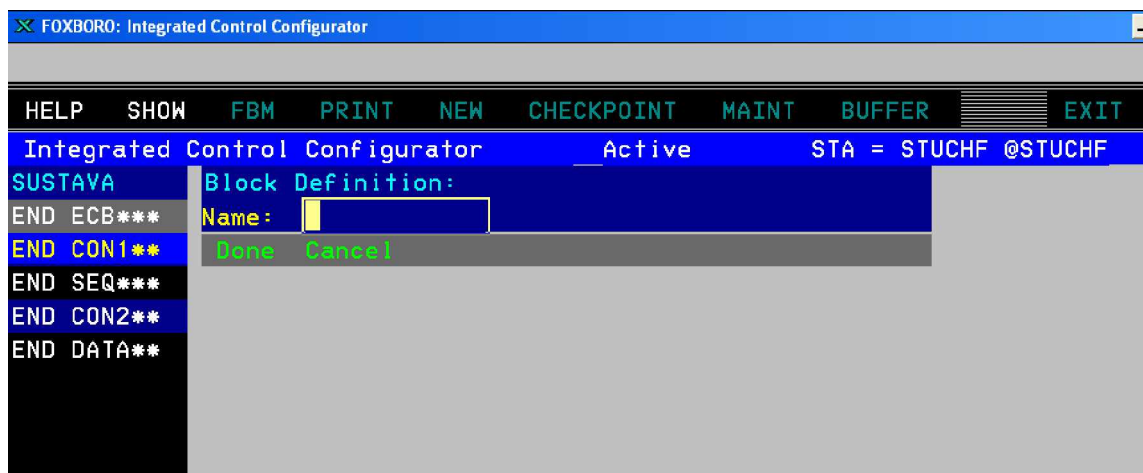
Obr. 4.6 Rozšírená ponuka funkcií bloku a ECB

View Compound List

Táto funkcia je dostupná z menu funkcií bloku a ECB (obr. 4.6). Po jej zvolení sa obrazovka vráti do základnej obrazovky v ICC so zoznamom compoundov (obr.4.2).

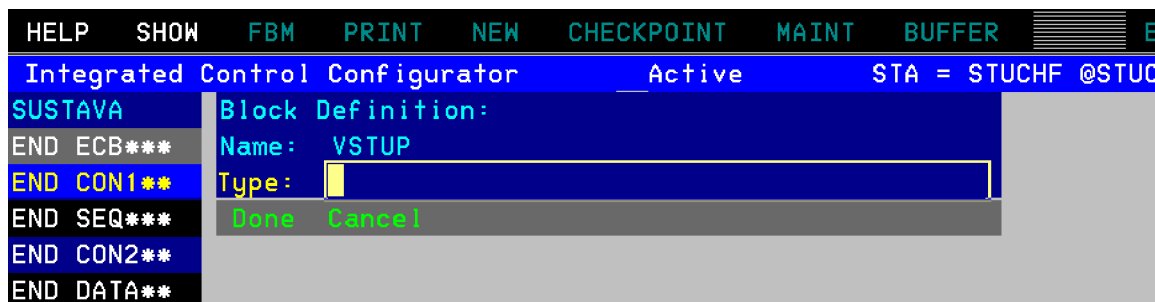
Insert New Block

Ak chceme vložiť nový blok, treba označiť pred ktorý blok v zozname, ho chceme vložiť (pri obr. 4.5 by sa nový blok vložil pred položku END_CON1**). Vyberieme ponuku **Insert New Block/ECB** zo zoznamu **Block/ECB Functions** (obr. 4.6) a zobrazí sa okno na definíciu názvu nového bloku (obr. 4.7).

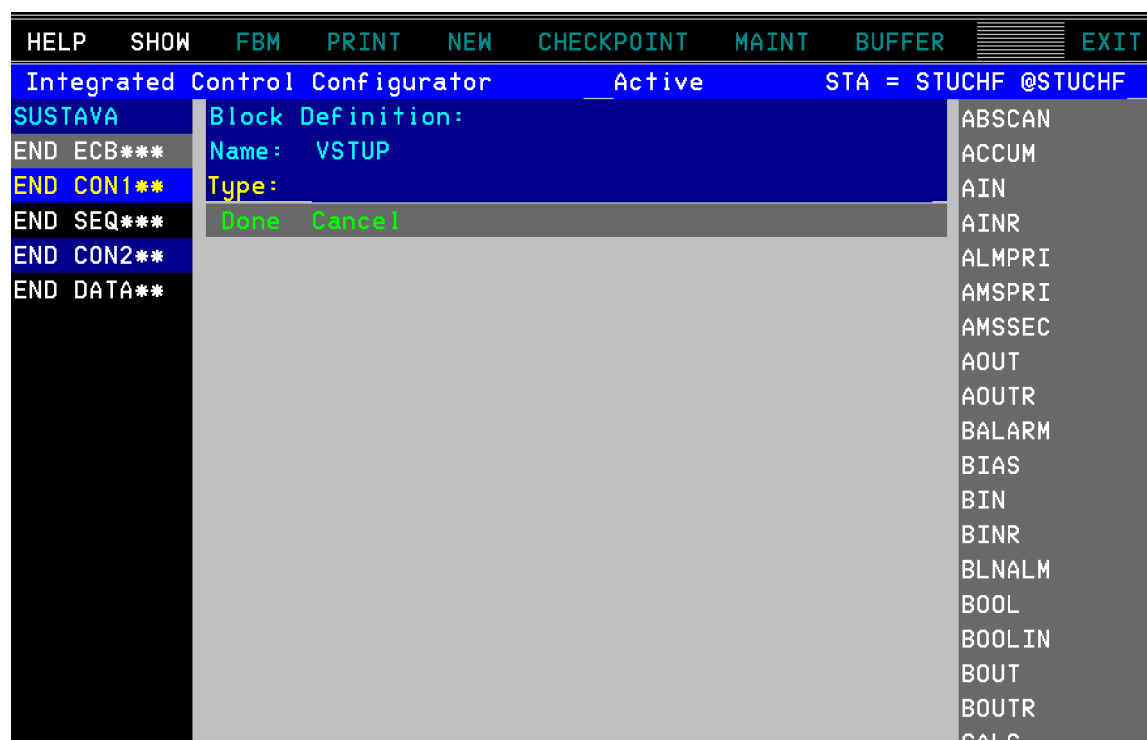


Obr. 4.7 Okno na definovanie názvu bloku

Názov musí byť platné meno, treba ho potvrdiť stlačením klávesy Enter, zobrazí sa položka na definíciu typu (obr. 4.8). Typ musí byť platný, napísaný veľkými písmenami, prípadne je možnosť vybrať typ zo zoznamu, ktorý sa zobrazí vpravo v okne (obr. 4.9) pomocou výberu **Show** zo stĺpca menu a výberom **Block_Type_Names** (obr. 3.1)



Obr. 4.8 Okno na definíciu typu bloku

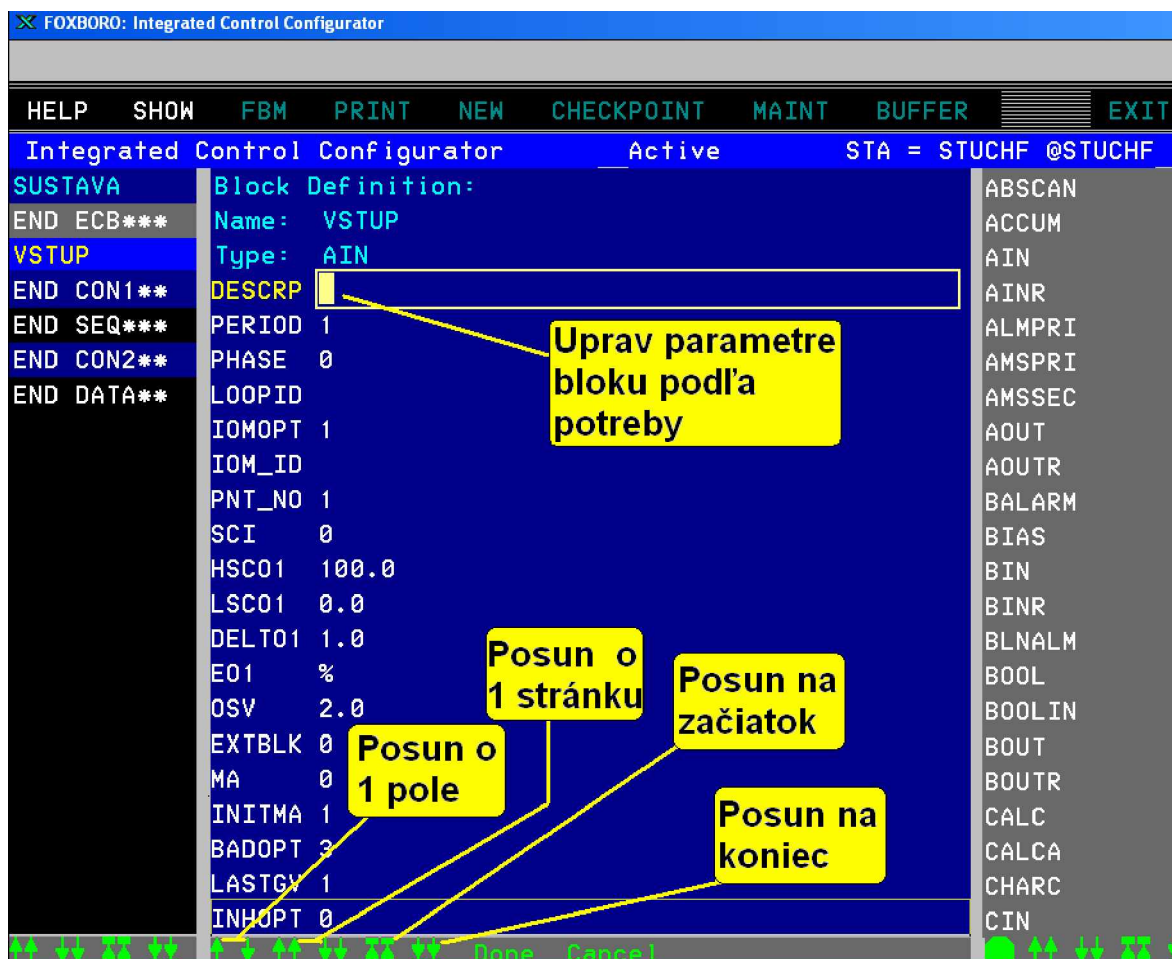


Obr. 4.9 Okno na definíciu typu bloku s časťou zoznamu typov blokov

Po odoslaní údajov, ak sú platné, je nový blok vložený s prednastavenými hodnotami do zoznamu blokov v kompounde

Úprava parametrov bloku

Výberom funkcie **Edit_All_Block/ECB_Parameters** sa zobrazí okno (obr. 4.10) v ktorom sa jednotlivé parametre upravujú prepisovaním v žltom orámovaní a následne sa musia potvrdiť stlačením klávesy Enter. Na potvrdenie uskutočnených zmien treba stlačiť Done, na ich zrušenie a návrat do obrazovky funkcií treba stlačiť Cancel.



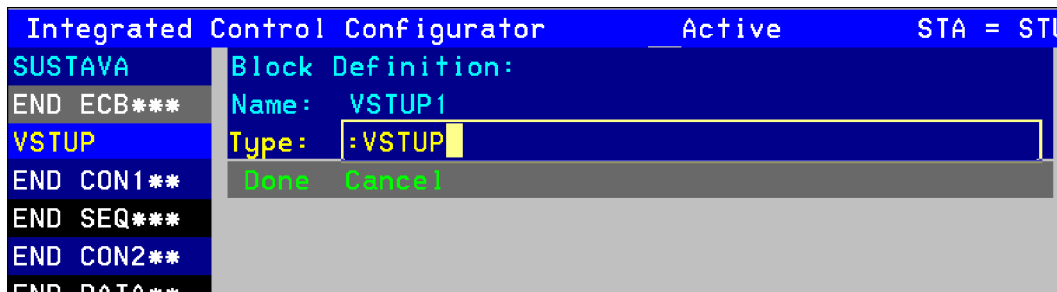
Obr. 4.10 Okno na úpravu parametrov bloku

Kopírovanie bloku

Je výhodné v prípade, ak chceme prekopírovať blok s upravenými parametrami, pretože pri kopírovaní sa prekopírujú všetky parametre.

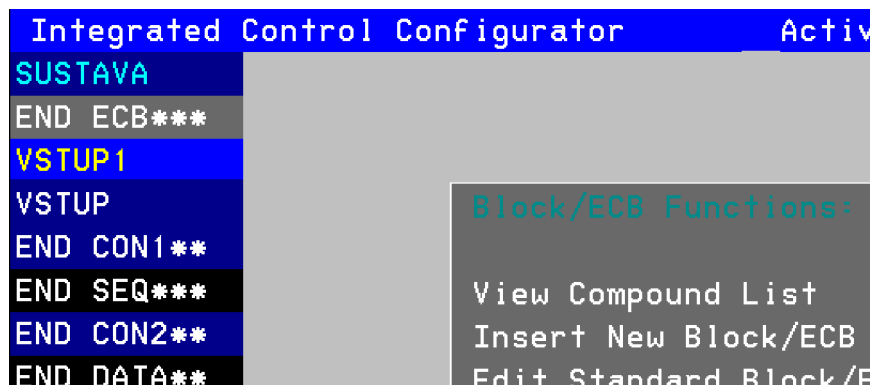
Pri kopírovaní bloku z existujúceho bloku treba v zozname blokov označiť blok, ktorý bude v zozname po prekopírovanom (ak bude posledný v poradí (pred koncom), treba označiť END_CON1**), potom sa vyberie funkcia **Insert_New_Block/ECB** (objaví sa okno pre definíciu bloku (obr. 4.7)). V Name treba vložiť platné meno nového bloku, potvrdiť klávesou Enter, ak je prijaté objaví sa pole na definíciu Type (obr. 4.8). Tu treba vložiť meno bloku, ktorý bude kopírovaný:

- ak ide o blok vzdialený súčasnemu kompoundu, bude meno bloku v tvare **COMPOUND:MENO_BLOKU**
- alebo, ak je blok v súčasnom kompounde (obr. 4.11) bude meno bloku v tvare **:MENO_BLOKU**



Obr. 4.11

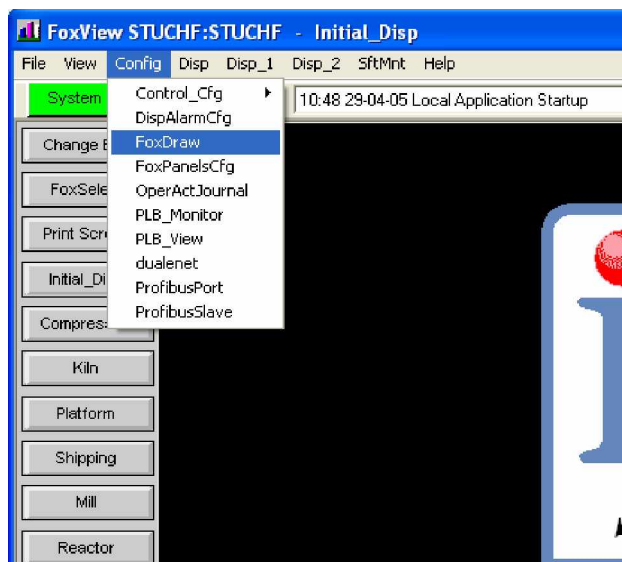
Odošle sa klávesou Enter, ICC skontroluje platnosť, ak je typ bloku správne zapísaný, blok je prekopírovaný do nového umiestnenia a obsahuje to isté nastavenie parametrov, ako predloha. Novo vytvorený blok sa ihneď objaví nad vyznačeným miestom v zozname blokov (obr. 4.12), v prípade potreby je možné upraviť parametre bloku.



Obr. 4.12

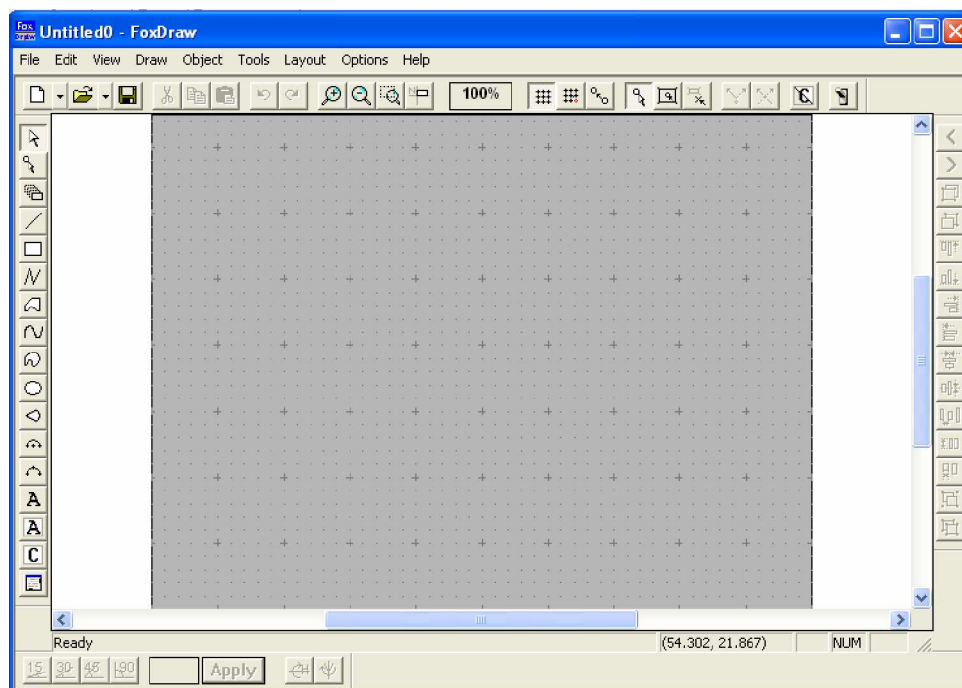
5 FoxDraw

FoxDraw je grafický zobrazovací editor pre vytváranie a údržbu displejov slúžiacich na vizualizáciu dynamiky riadených procesov. Prístup ku základnej obrazovke FoxDraw je zo základného displeja vybratím funkcie **Config – FoxDraw** (obr. 5.1).



Obr. 5.1 Výber ponuky FoxDraw zo základnej obrazovky

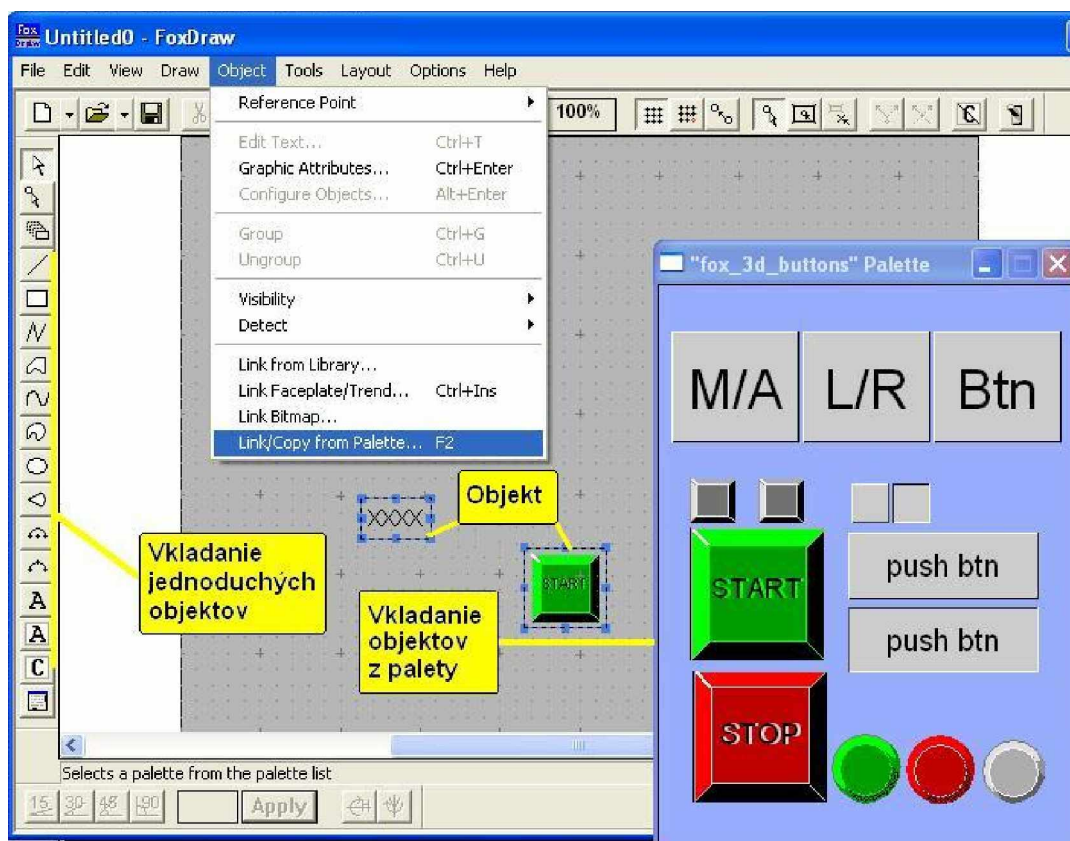
Ako prvá sa otvorí obrazovka, ktorá bola naposledy upravovaná, na výber novej treba vybrať **File – New** alebo klávesovú skratku **Ctrl+N** (obr. 5.2).



Obr. 5.2

Displej je vytvorený a konfigurovaný súbor na pozeranie z obrazovky FoxView alebo Display Managera.

Displej (obr. 5.3) sa skladá z objektov, pričom všetky môžu mať nakonfigurované hodnoty. Pojem „objekt“ zahŕňa jednoduché objekty (čiary, trojuholníky, text, ..), objekty z knižníc a bitové mapy (logo spoločnosti, obrázok prevádzky).



Obr. 5.3 Obrazovka FoxDraw s vloženými objektmi

Displej môže byť nakonfigurovaný na umožnenie zásahov operátora do procesu pomocou pohybu objektov alebo vpísaním vstupov.

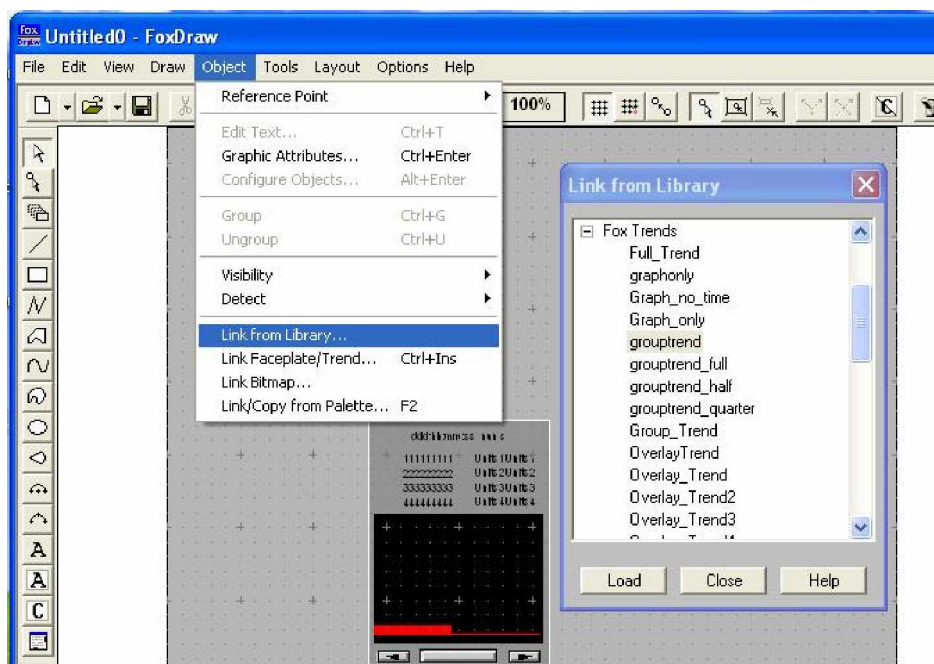
5.1 Vkladanie objektov do obrazovky pre užívateľa

Do obrazovky sa môžu vkresliť objekty použitím lišty s nástrojmi, pomocou príkazov menu, klávesovými skratkami a vkladáním z knižníc.








Najjednoduchšie vytvorenie objektu v obrazovke je jeho skopírovanie z knižnice objektov (**Object – Link from Library** (obr. 5.4) ,**Link/Copy from Palette**(obr. 5.3)) označením vybraného objektu (2x ľavým tlačidlo myši) a prenesením hraníc objektu (ktoré sa zobrazia potom ako kurzor opustí paletu) na vybrané miesto v obrazovke.

Link from Library obsahuje značky, grafy, symboly a tlačidlá.

Link/Copy_from_Palette obsahuje rozsiahly výber symbolov(jednoduché obrázky, zariadenia (ako čerpadlá, motory, ventily, ...)).



Obr. 5.4 Vloženie grafu z knižnice obrázkov

Inou cestou tvorby objektu je jeho vytvorenie z častí získaných použitím lišty nástrojov na ľavej strane obrazovky. Tlačidlo **Select**  umožňuje označiť objekt, **Line**  kresliť rovnú čiaru, **Three-Point Sector**  vytvoriť oblúk, **Text**  pridať alebo zmeniť text, **Background Text**  vytvoriť a zmeniť text pozadia. **Group**  zoskupuje dohromady rozmanité objekty ktoré sa potom chovajú ako jeden z dôvodu jednoduchšej manipulovateľnosti (úprava veľkosti, umiestnenia, otáčanie a úprava grafických atribútov). **Ungroup**  ruší zoskupenie objektov dosiahnuté tlačidlom **Group**.

5.2 Konfigurácia objektov v obrazovke užívateľa

Vzhľad objektu je možné zmeniť (inak ako je pôvodne definovaný). Definované hodnoty poväčšine obsahujú farbu vnútra, farbu hrán, štýl čiar hrán,

Cez komunikačný rámček je možné nastaviť objekt obrazovky. Dajú sa použiť dve metódy a to aj súčasne:

- aplikáciou jednej alebo viacerých dynamických vlastností (napríklad viditeľnosť, výška hladiny) na objekt a pripojením týchto vlastností do premennej procesu alebo do súboru. Pri tomto type nastavenia sa pri zmene hodnoty procesu zmení zobrazenie objektu. Nie je potrebný zásah operátora.

- pripojením objektu na akciu operátora (napríklad otvorenie obrazovky, vykonanie príkazu). Tento typ nastavenia umožní napríklad spúšťať úkon vybratím objektu.

Poznámka –

1. *Aj keď môže mať jeden objekt mnohé dynamické vlastnosti môže mať najviac iba jednu akciu operátora.*
2. *Každý typ objektu má svoj vlastný súbor dynamických vlastností (obr. 5.5).*
3. *Objektu vytvorenému a nastavenému vo FoxDraw je možné priradiť jedinečné meno. I keď mená sú nepovinné, pomáhajú rozlišovať medzi niekoľkými objektmi toho istého typu v jednej obrazovke.*

Dynamické hodnoty (atribúty) menia vzhľad objektov obrazovky ako reakciu na zmeny premenných procesu alebo systémových premenných [6].

Dynamické vlastnosti	Priamka	Nadväzujúce úsečky Krivka	Mnohouh, olník, Uzavretá krivka	Kruh, Výsek kruhu	Text	Text pozadia	Bitová mapa	Skupina	Objekt knižnice
Visibility	X	X	X	X	X	X	X	X	X
Fill Level		X	X	X		X			X
Filled		X	X	X		X		X	X
Fill Color		X	X	X		X		X	X
Fill Blink		X	X	X		X		X	X
Fill Pattern		X	X	X		X		X	X
Edge Style	X	X	X	X		X		X	X
Edge Width	X	X	X	X		X		X	X
Edge Color	X	X	X	X		X		X	X
Edge Blink	X	X	X	X		X		X	X
Text Contents					X	X			X
Text Color					X	X		X	X
Text Blink					X	X		X	X
Text Height					X	X		X	X
Move	X	X	X	X	X	X	X	X	X
Radius				X					
Arc Length				X					
Rotate	X	X	X	X	X	X		X	X
Scale	X	X	X	X		X	X	X	X

Obr. 5.5 Dynamické vlastnosti objektov

Move Horizontal/Vertical Dynamic umožňuje nastaviť objekt na pohyb zvislý, vodorovný alebo oba v pomere hodnoty premennej procesu. Určením hornej a dolnej pozície sa definuje rozsah pohybu. V prípade prekopírovania objektu s uvedenými vlastnosťami má kópia nastavený ten istý rozsah ako originál, aj keď týkajúceho sa jej vlastného referenčného bodu.

Fill Level Dynamic združuje hodnotu procesu s úrovňou naplnenia objektu (objekt musí byť vyplnený farbou) smer naplňania môže byť nahor, nadol, doprava alebo doľava. Minimálna hodnota môže byť 0% a maximálna 100%.

Visibility Dynamic sa používa na zobrazenie alebo skrytie objektu v závislosti od stavu hodnoty procesu (napr. tlačidlo Restart sa môže zobrazovať iba ak je proces v stave Hold).

Fill Color Dynamic spája stav premennej procesu s farbou náplne objektu na obrazovke

Vybraná akcia operátora sa spustí po tom, ako operátor vyberie objekt (kliknutím, vpísaním hodnoty). K dispozícii sú nasledujúce akcie:

Open Display zatvorí súčasnú obrazovku a všetky prekrývajúce obrazovky a otvorí novú základnú obrazovku (musí byť umiestnená v tom istom zariadení v ktorom je spustené FoxView)

Open Overlay otvorí prekrývajúcu obrazovku na vrchu základnej obrazovky alebo ďalšiu prekrývajúcu obrazovku (musí byť umiestnená v tom istom zariadení ako FoxView)

Close Display/Overlay zatvorí súčasnú obrazovku alebo prekrývajúcu obrazovku, po zatvorení základnej obrazovky sa otvorí sa predchádzajúca obrazovka

Display Manager Command uskutoční jeden alebo viac príkazov manažéra

Relative Pick vyberie parameter na ovplyvnenie a vykoná jednoduchý alebo viacnásobný príkaz, toto spojenie sa používa ešte s inou akciou ako napríklad **Momentary Contact, Connect Variable**

Momentary Contact okamžite prepína alebo zasiela Boolean premennú

Connect Variable posielajú hodnotu premennej procesu podľa prevodovej tabuľky

Set Text Contents povoľuje poslať premennú po zapísaní hodnoty do textového poľa (akcia je platná s objektmi typu Text a Background Text)

6 Využitie ICC na riadenie zásobníkov

Počas práce som využíval malú časť riadiaceho programu I/A Series FOXBORO. Signály potrebné na riadenie sa nedali získať z reálneho zariadenia. Preto som v programe I/A Series použil možnosť vložiť blok, ktorý predstavuje model zásobníkov kvapaliny v tvare diferenciálnej rovnice.

Na simuláciu dynamických vlastností zásobníkov som použil dve diferenciálne rovnice prvého rádu s nulovými počiatočnými podmienkami. V bloku LLAG je diferenciálna rovnica definovaná v tvare:

$$\begin{aligned} &y + \text{LAGTIM} * dy/dt + \text{LAGTIM} * \text{LAG2} * dy^2/dt^2 = \\ &x + \text{LGAIN} * \text{LAGTIM} * dx/dt \end{aligned}$$

Obr. 6.1

Pomocou bloku PIDA som riadil výstupnú veličinu (výšku hladiny v druhom zásobníku) na žiadanú hodnotu pomocou regulátora ktorého parametre boli získané metódou umiestnenia pólov a Naslinovou metódou. Tieto parametre boli posielané z obrazovky operátora pomocou dynamických vlastností objektov po nadefinovaní ich hodnôt operátorom (poslanie akciou **Set_Text**). Blok PIDA som používal ako blok, ktorý tvoril alarmy (na základe merania vstupnej veličiny). Na definovania alarmov som nastavil hodnoty parametrom MEASHL, MEASLL, HHALIM, LLALIM, HHAOPT (zobrazené v Prílohe 2, bližší popis parametrov v Kapitole 6.3) spolu so stanovením hodnôt priority MEASPR, HHAPRI pre jednotlivé skupiny alarmov. Z viacerých typov regulátorov (Kapitola 6.3, parameter MODOPT) som zvolil riadenie PID regulátorom (MODOPT = 5).

Blok SWCH som používal vybratie jedného z dvoch vstupov na výstup (výstup bloku SWCH bol použitý ako vstup do prvého bloku LLAG) pomocou parametra TOGGLE, ktorý bol posielaný do bloku z obrazovky operátora ako pravda (nepravda). Parameter TOGGLE na základe svojej hodnoty prepínal medzi vstupmi. Jedným zo vstupov do bloku SWCH bol vypočítaný výstup z regulátora a druhý vstup bola nulová hodnota (pri prepnutí výstupu bloku SWCH na nulovú hodnotu bol zablokovaný prenos signálu regulátora na vstup do bloku LLAG).

V bloku CALCA boli vykonávané matematické operácie spočítavania a násobenia, pomocou ktorých som vypočítaval tvar diferenciálnej rovnice druhého rádu z dvoch

diferenciálnych rovníc prvého rádu. Tieto výpočty slúžili ako výstupy do obrazovky operátora.

Detailné nastavenie parametrov je uvedené v Prílohe 2.

6.1 Blok CALCA

poskytuje logické funkcie spolu s možnosťou aritmetických výpočtov v jednom integrovanom prostredí. Poskytuje väčšiu účinnosť vo viacerých matematických a logických príkazoch, čo vyplýva na zmenšenie dĺžky programu viac ako trojnásobne oproti tomu istému výpočtu uskutočnenému v bloku CALC. CALCA nepodporuje blokovanie reálnych výstupov zatiaľ čo CALC ho podporuje.

Vložený program, je vykonávaný vždy keď je vykonávaný blok CALCA.

Definícia parametrov:

M01–M24 24 pamäťových zápisov, poskytujú dočasné uloženie výsledkov operácií

RI01–RI08 8 reálnych vstupov do výpočtov v bloku.

RO01–RO04 4 reálne výstupy z výpočtov v bloku.

STEP01–STEP50 50 programovateľných príkazov (typu reťazca veľkosti maximálne 16 znakov), ktoré vykonávajú funkciu kalkulačky.

6.2 Blok LLAG

dynamicky vyrovnáva zmeny v signáli merania buď počiatočnou vyššou odozvou na vstup (Lead), alebo pozvoľnou zmenou výstupu (Lag), alebo oboma. Výstup má ustálený stav, ktorý sa mení so vstupom ak je blok v Lead/Lag móde. V impulznom móde má blok jednoduchý ustálený stav závislý iba od vstupu BIAS.

Definícia parametrov:

DELTI1,DELTI2 zmena delta pre vstup rozsahu 1 a 2, stanovuje rozlíšenie rozsahu merania (vloženie hodnoty 1 zapríčiní rozpoznanie a reakciu na zmenu 1 percenta z celého rozsahu chyby)

HOLIM je reálny vstup ktorý zavádza hodnotu maximálneho výstupu v jednotkách OUT. Ak sa algoritmus snaží viesť výstup na vyššiu hodnotu, výstup je vnútený na hodnotu HOLIM a indikátor HOLIND je poslaný na true.

HOLIND je boolean výstup, ktorý je poslaný pravda (true), kedykoľvek je výstup vnútený na hodnotu HOLIM

HSCI1,HSCI2 sú reálne hodnoty určujúce horný limit rozsahu merania (definovaný jednotkami EI1 a EI2). Typická hodnota je 100 (percent).

LAG2 je kvadratická časová konštanta v lead, kvadratickom-lag filtri. Je použitá v nasledujúcom vzorci (Obr. 7.1 kde(x = vstupný signál, y = výstupný signál))

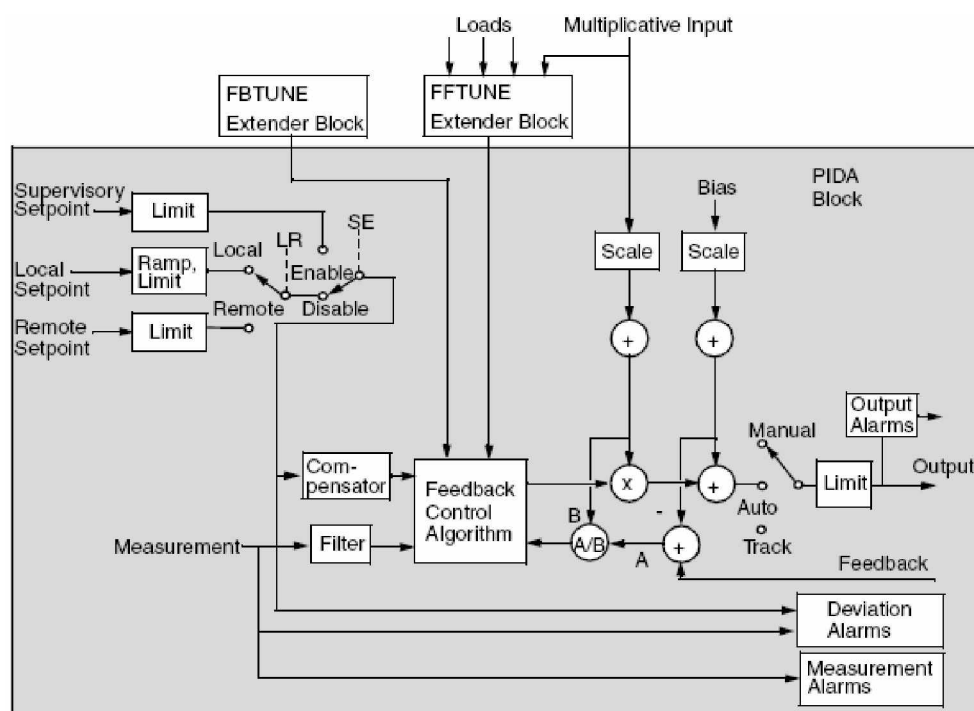
LAGTIM je vstupný parameter, ktorý určuje časovú konštantu prvého rádu (v minútach)

LGAIN je reálny vstupný parameter, ktorý určuje okamžité zosilnenie na výstupe úmerné zmene úmerné zmene kroku vo vstupe merania

MEAS je vstup ktorý určuje zdroj vstupu bloku alebo riadenú premennú.

6.3 Blok PIDA

PIDA regulátor uskutočňuje priebežnú spätnú väzbu PID a súčtové a násobiace riadenie doprednej analógovej slučky (obr. 7.2). Jeho hlavné vstupy, setpointy a merania sú použité na výpočet výstupu, spracované premenné založené na nastavených parametrov – pásmo proporcionality, integračný čas, derivačný čas, doba oneskorenia a relatívne zosilnenie setpointu.



Obr. 6.2 Schéma funkčného bloku PIDA

Kombinácia bloku PIDA a rozširovacích blokov FBTUNE a FFTUNE umožňuje schopnosť adaptívneho riadenia.

Prvých päť módov bloku PIDA (MODOPT=1 až 5) sú tie isté ako pre PID regulátor, okrem toho, že majú pridaný integračno-derivačný kompenzátor setpointu. Kompenzátor setpointu umožňuje aby počet proporcionálnych akcií ktoré sa obracajú na setpoint bol

menší ako tých, ktoré sa obracajú na meranie. Umožňuje, aby bol blok súčasne ladený pre dobré potlačenie odozvy a neprekročenie setpointu.

Definícia parametrov:

??ATXT je textovým reťazcom nadefinovaný užívateľom (do 32 znakov), je výstupom pri vzniku alarmu a slúži na jeho identifikáciu (?? sú rôzne písmená podľa typu alarmu)

DALOPT je vstupom, ktorý umožňuje alebo znemožňuje alarm hornej a dolnej odchýlky (0=žiadan alarm, 1=alarm hornej a dolnej odchýlky, 2=alarm iba hornej odchýlky, 3=alarm iba dolnej odchýlky) DALOPT je možné zmeniť iba pri konfigurácii

DERIV – derivačný čas je reálny vstup v minútach.

DEVADB reálny vstup (v jednotkách MEAS), používa limity odchýlky (horný a dolný)

DEVPRI je vstupom, ktorý nastavuje úroveň priority alarmu (1-5, 1 je najvyššia priorita)

ERROR je reálny vstup rovný hodnote setpoint SPT mínus meranie MEAS

HDAIND (LDAIND) je boolean výstup poslaný pravda, ak je meranie väčšie(menšie) od setpointu o viac ako je limit odchýlky HDALIM (LDALIM), pri vrátení merania späť cez hodnotu DEVADB blok posiela HDAIND (LDAIND) nepravda

HDALIM (LDALIM) je reálny vstup, ktorý stanovuje ako rýchlo ktoré musí meranie prekročiť (klesnúť pod) žiadanú hodnotu na spustenie alarmu a poslanie HDAIND (LDAIND) pravda

HHAIND (LLAIND) je boolean výstup poslaný pravda, ak meranie prevyšuje (klesne pod) HHALIM (LLALIM), ak je poslaný pravda pošle hlásenie nepravda až keď hodnota MEAS klesne pod HHALIM mínus (stúpne nad LLALIM plus) MEASDB

HHALIM (LLALIM) je reálny vstup, ktorý definuje hodnotu MEAS, ktorá spúšťa High-High (Low-Low) alarm

HHAOPT umožňuje alebo znemožňuje High-High a Low-Low absolútny alarm pre alarm merania MEAS. Každý alarm spúšťa indikátor a textovú správu (0=žiadan alarm, 1=High-High a Low-Low alarm, 2=iba High-High alarm, 3=ibaLow-Low alarm)

HHAPRI je vstup, ktorý posiela úroveň priority high-high absolútneho alarmu (1-5, 1 je najvyššia priorita)

HOAIND (LOAIND) boolean vstup poslaný pravda kedykoľvek je výstup väčší ako HOALIM (menší ako LOALIM)

HOALIM (LOALIM) vstup definujúci hodnotu výstupu OUT ktorá spustí alarm vysokého (nízkeho) výstupu

HOLIM (LOLIM) je vstup ktorý definuje maximálnu (minimálnu) hodnotu výstupu, ak sa algoritmus snaží viesť výstup na vyššiu (nižšiu) hodnotu, výstup je vnútený na hodnotu HOLIM (LOLIM) a je poslaný indikátor HOLIND (LOLIND) pravda

HOLIND (LOLIND) boolean výstup, je poslaný pravda vždy pri vnútení výstupu na hodnotu HOLIM (LOLIM)

INT –integračný čas, je reálny vstup v minútach

KD faktor filtra merania, je reálny vstup, nastavuje časovú konštantu filtra merania

MALOPT nastaviteľný vstup umožňuje alebo znemožňuje absolútny alarm High a Low merania (0=žiaden alarm, 1=alarm High a Low merania, 2=iba alarm High merania, 3=iba alarm Low merania). Hodnotu MALOPT je možné zmeniť iba pri rekonfigurácii bloku.

MEAS je vstup, ktorý identifikuje zdroj vstupu merania bloku, riadená premenná.

MEASDB konfigurovateľný vstup (v jednotkách MEAS), týka sa High, Low, High-High a Low-Low alarmových limitov

MEASHI (MEASLI) je boolean výstup, poslaný pravda ak meranie MEAS prevyšuje horný limit alarmu MEASHL (klesne pod dolný limit alarmu MEASLI)

MEASHL (MEASLL) vstup ktorý definuje hodnotu merania, ktorá spustí high (low) absolútny alarm

MEASPR je celočíselný vstup, posiela úroveň priority alarmu merania (1-5, 1=najvyššia priorita)

MODOPT konfigurovateľný parameter (1-8) ktorý nariaďuje regulačný mód ak je blok v stave Auto.

1 = P (len proporcionálny)

2 = I (len integrálny)

3 = PD (proporcionálny plus derivačný)

4 = PI (proporcionálny plus integračný)

5 = PID (proporcionálny, integračný, derivačný)

6 = NPID (neovplyvňujúci PID)

7 = PITAU (proporcionálny, integračný, s dopravným oneskorením, je ovplyvňujúci ak je derivačný člen nenulový)

8 = PIDTAU (neovplyvňujúci proporcionálny, integračný, derivačný s dopravným oneskorením)

PBAND proporcionálne pásmo je vstup vyjadrený v percentách. PBAND je percento zmeny intervalu na vstupe, ktoré spôsobí zmenu plného intervalu na výstupe. $(100/\text{PBAND})$ určí zosilnenie regulátora ak sú MEAS a OUT prevedené na percentá intervalu.

SPT - aktívna požadovaná hodnota. SPT je referenčná premenná, ktorá je porovnávaná so vstupom MEAS na tvorenie signálu ERROR.

6.4 **Blok SWCH**

Tento blok podporuje výber jedného z dvoch reálnych vstupov. V móde Auto prepínacia zmena vstupu riadi, ktorý z dvoch vstupov je vybraný na výstup. V móde Manuál nie je výstup aktualizovaný blokom. Externý program potom môže nastaviť výstup na želanú hodnotu. Po prepnutí bloku do Auto blok okamžite riadi výstup na hodnotu vstupu vybranú vstupom TOGGLE.

Definícia parametrov:

BTIME1 (BTIME2) čas vyváženia pre vstup 1 (2) je reálny vstup ktorý určuje čas pre preklopenie zo vstupu 2 (1) na vstup 1(2). Výstup pristupuje na hodnotu vstupu 1 (2) s reakciou charakteristickou pre oneskorenie prvého rádu ktoré má časovú konštantu BTIME1 (BTIME2) v minútach. Hodnota súčasného vstupu do tohto parametra predstavuje takmer jednu pätinu z času ktorý potrvá výstupu na prenos zo hodnoty vstupu 2 (1) na hodnotu vstupu 1 (2).

INP1, INP2 sú vstupy vyberané na sledovanie výstupu pomocou parametra TOGGLE

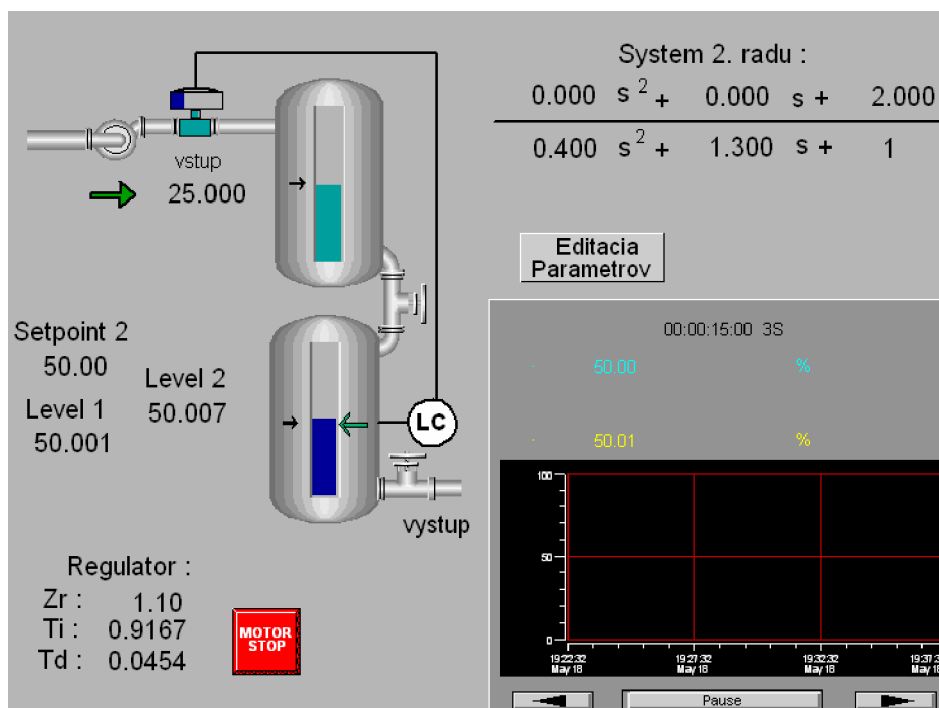
TOGGLE je boolean vstup, ktorý uskutočňuje výber vstupu. Vstup 1 je vybraný, ak je Toggle 0 (false). Vstup 2 je vybraný, ak je Toggle 1 (true).

7 Rozhranie vytvorené vo FoxDraw

7.1 Popis rozhrania

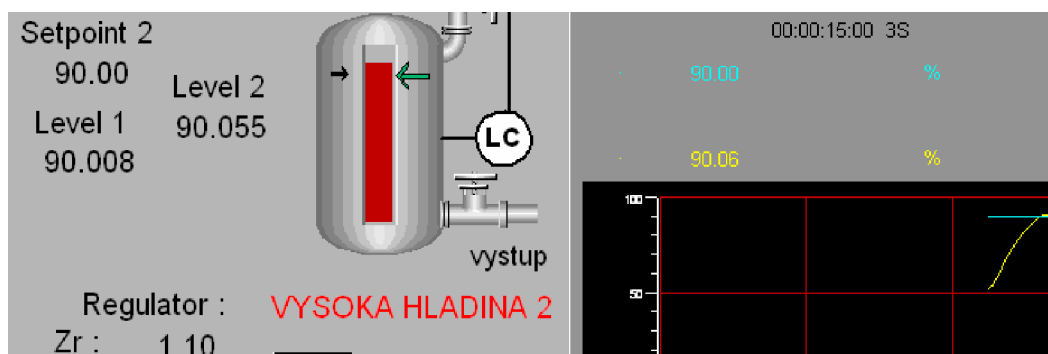
Pre zobrazenie priebehu riadenia zásobníkov kvapaliny boli vytvorené dve obrazovky vo FoxDraw.

Prvá (obr. 7.1) obsahuje obrázok dvoch výmenníkov s dynamickým zobrazením meniacich sa hladín, spolu so zmenou farby náplne v druhom zásobníku pri prekročení alarmových limitov nastavených v bloku PIDA v ICC.

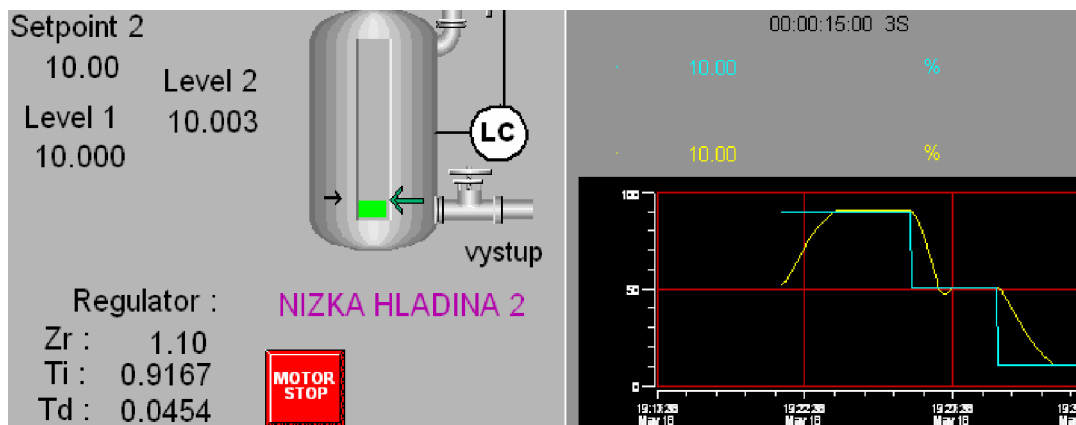


Obr. 7.1 Základná obrazovka

Farba náplne sa zmení inak pri poklese pod dolný a stúpnutí nad horný alarmový limit (nastavený v bloku PIDA v parametroch MEASHL, MEASLL spolu s nastavením priority alarmu), inak sa zmení (znázornené na obr. 7.2 a 7.3) pri prekročení najvyššieho a poklese pod najnižší alarmový limit (HHALIM a LLALIM parametre bloku PIDA s nutným nastavením priority alarmu a umožnením alarmovania zmenou parametra HHAOPT).



Obr. 7.2 Obrazovka užívateľa pri prekročení limitu HHALIM

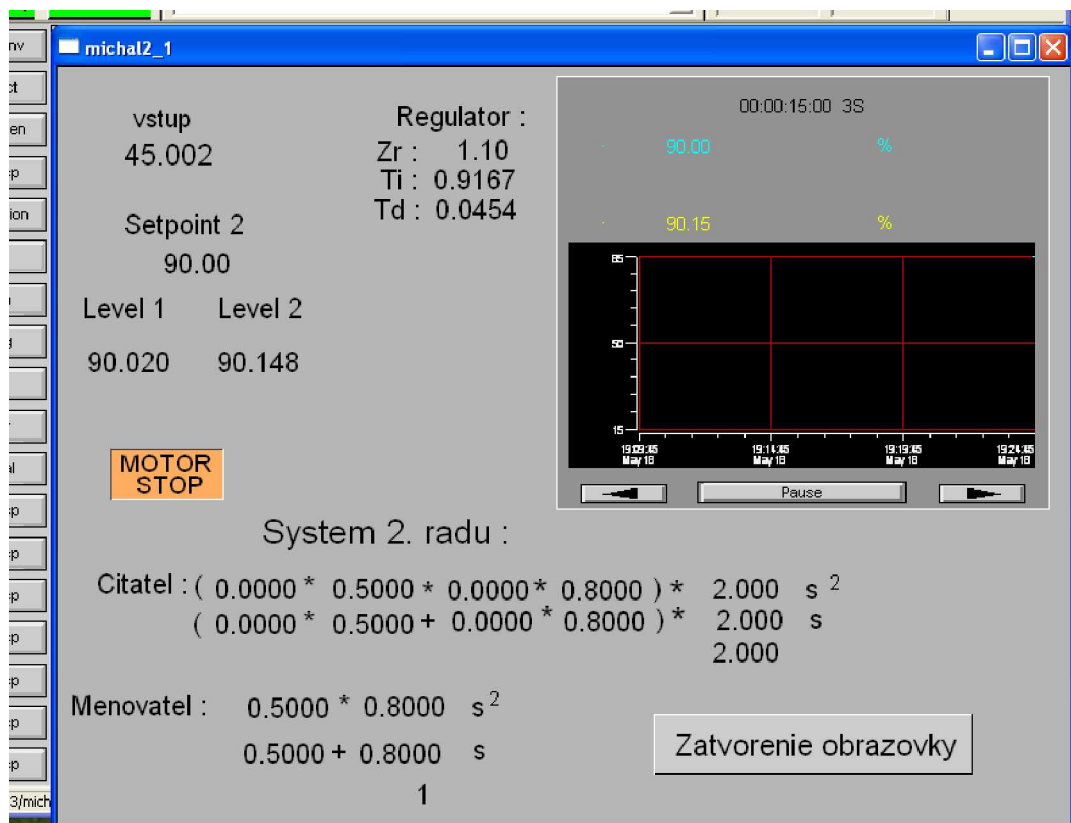


Obr. 7.3 Obrazovka užívateľa pri poklese pod limit LLALIM

Ako vidieť na obr. 7.1, obr. 7.2, obr. 7.3 pod zásobníkmi sú umiestnené textové polia, ktoré informujú užívateľa o aktívnych alarmoch správou ktorá sa v nich zobrazí. Polia sú viditeľné iba pri stave aktívneho alarmu – ak je stav parametrov HHAIND, LLAIND pravda a je aktívny parameter HHAOPT (bližšie popísané v Kapitole 6.3). Na obrazovke sú prístupné dve užívateľské tlačidlá, jedno prepína medzi vpustením výstupnej hodnoty regulátora „MOTOR STOP“ na vstup do zásobníka a druhé „Editacia parametrov“ (obr. 7.1) má za úlohu otvoriť prekrývajúcu obrazovku. Sú tu polia, ktoré zobrazujú parametre regulátora načítavané z bloku PIDA (pri zmene parametrov v bloku sa automaticky prepíšu parametre týchto polí na aktuálne hodnoty), prenos sústavy dvoch zásobníkov (obe polia sú bez možnosti zmeny v tejto základnej obrazovke) a pole, ktoré podporuje zmenu nastavenia setpointu. Okrem toho je tu graf zobrazujúci priebeh zvolených hodnôt od času a obrázok, ktorý sa mení podľa toho, či do zásobníka vstupuje výstup regulátora alebo nulová hodnota.

Druhá obrazovka (obr. 7.4) zobrazuje súčasné hodnoty regulátora a prenosu sústavy spolu s výškami hladín v zásobníkoch. V tejto obrazovke je možnosť úpravy parametrov regulátora a prenosu sústavy. Po zmene parametrov v tejto obrazovke sú automaticky


aktualizované hodnoty na základnej obrazovke. Je tu graf, ktorý zobrazuje priebeh zvolených hodnôt v závislosti od času a dve tlačidlá, jedno zatvára túto obrazovku „Zatvorenie obrazovky“ a druhé „MOTOR STOP“ prepína medzi poslaním výstupu regulátora a medzi poslaním nulovej hodnoty na vstup do zásobníka.

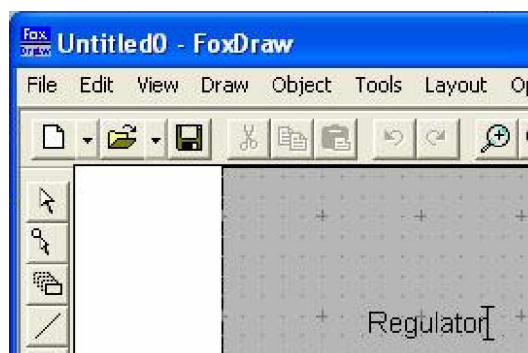


Obr. 7.4

7.2 Postup tvorby a konfigurácie základných objektov


Pole textu


Použil som ho napríklad na napísanie „Regulator“ (obr. 7.4), vybral som ponuku Text z lišty nástrojov , presunul kurzor na zvolené miesto obrazovky, potvrdil miesto stlačením ľavého tlačidla myši a vpísal zvolený text (obr. 7.5).



Obr. 7.5

Presun objektu

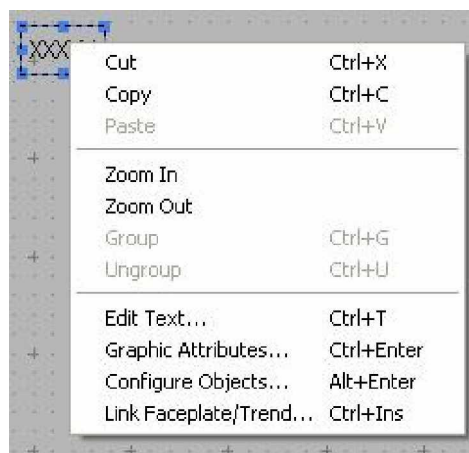
Na úpravu užívateľskej obrazovky bolo potrebné premiestňovať viaceré vložené objekty. Vybral som ponuku Select z lišty nástrojov , označil celý objekt stlačením ľavého

tlačidla myši nad objektom  a presun objektu na vybrané miesto stlačeným ľavým tlačidlom.

Úprava vzhľadu objektu a možnosť zviditeľnenia

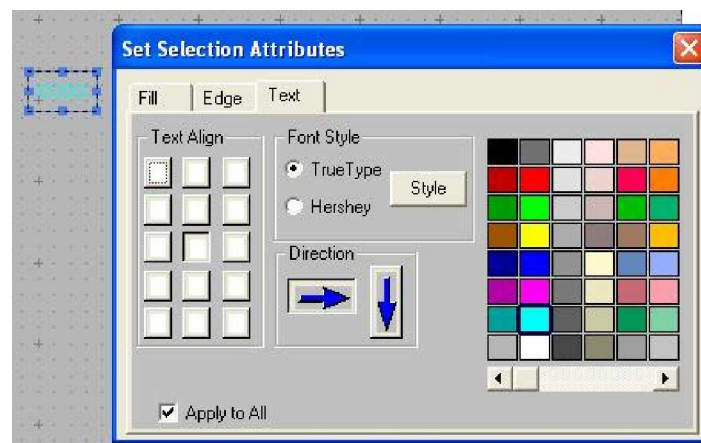
Použil som to napríklad na zobrazenie alarmu pre operátora „VYSOKA HLADINA 2“, ktorý sa zobrazí pri poslaní parametra MEASHI pravda (pri stúpnutí nad výšku hladiny definovanú v bloku PIDA ako MEASHL)

Vytvoril som textové pole, označil ho kliknutím pravým tlačidlom a z výberu (obr. 7.6) som zvolil možnosť **Graphic_Attributes**



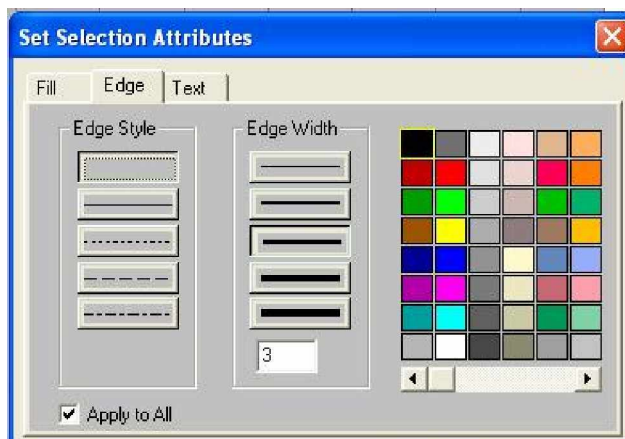
Obr. 7.6 Ponuka po výbere možností pravým tlačidlom myši

Z možností možností výplň (Fill), okraje (Edge), textové pole (Text) som zvolil Text a vybral som farbu ktorá sa aplikovala na text (obr. 7.7)

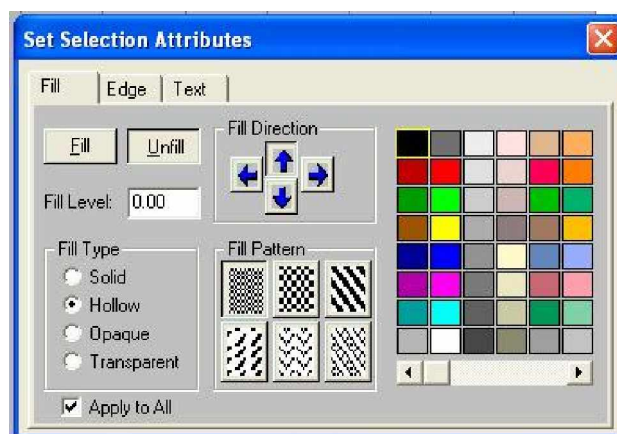


Obr. 7.7 Úprava farby textu

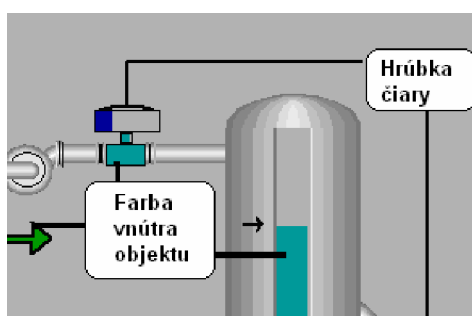
Okrem úpravy textu je v ponuke po výbere **Graphic_Attributes** možnosť úpravy čiar (obr. 7.8), a možnosť grafického nastavenia vnútra objektu (obr. 7.9). Možnosť úpravy hrúbky čiar a nastavenie farby vnútra objektu je použité na obr. 7.1, bližšie zobrazenie na obr. 7.10.



Obr. 7.8



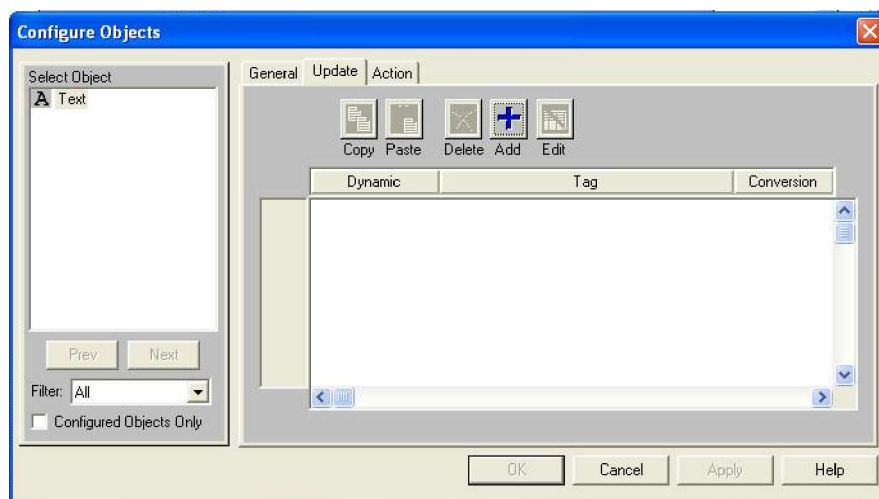
Obr. 7.9



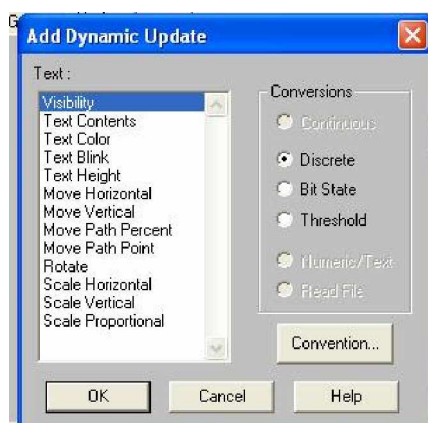
Obr. 7.10

Z výberu na obr. 7.6 som zvolil ponuku **Configure_Objects**, zobrazila sa ponuka (obr. 7.11), pri ktorej som zvolil položku **Update** a z ponuky **Add_Dynamic_Update** (obr. 7.12) zobrazenej po zvolení tlačidla **Add** som vybral položku **Visibility**.

Tu som v poli TAG definoval umiestnenie parametra, ktorý slúži ako vstupná hodnota (v tvare Compound:Blok.Parammeter) vybral počet stavov, ktoré parameter nadobúda a označil, pri ktorom stave chcem objekt vidieť (obr. 7.13).



Obr. 7.11



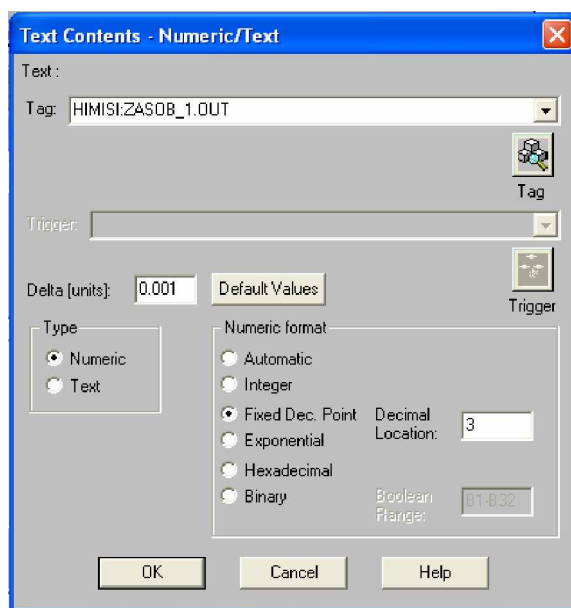
Obr. 7.12 Možnosti dynamického priradenia



Obr. 7.13 Príklad nadefinovania možnosti aktualizácie viditeľnosti

Aktualizácia textového poľa načítaním údajov

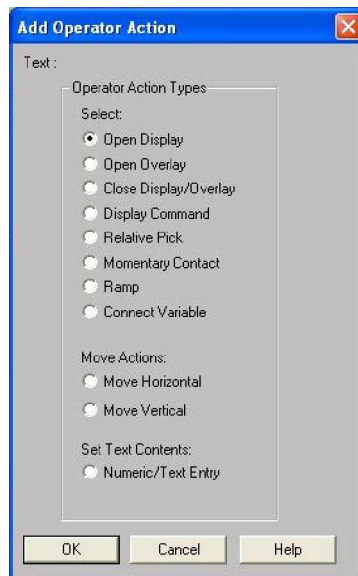
Aktualizácia je využitá na obr. 7.1 pri načítavaní parametrov regulátora a prenosu. Pri nadefinovaní aktualizácie výšky hladiny v prvom zásobníku som vo FoxDraw vytvoril textové pole (v tvare **XXX.XXX**), označil ho pravým tlačidlom myši, a z ponuky (obr. 7.6) som zvolil ponuku **Configure_Objects**, zobrazila sa ponuka (obr. 7.11), pri ktorej som zvolil položku **Update** a z ponuky **Add_Dynamic_Update** (obr. 7.12) zobrazenej po zvolení tlačidla **Add** som vybral položku **Text_Contents** (obr. 7.14). Je nutné definovať v Tag miesto, odkiaľ budú hodnoty textu načítavané, typ hodnôt (číselný alebo textový),



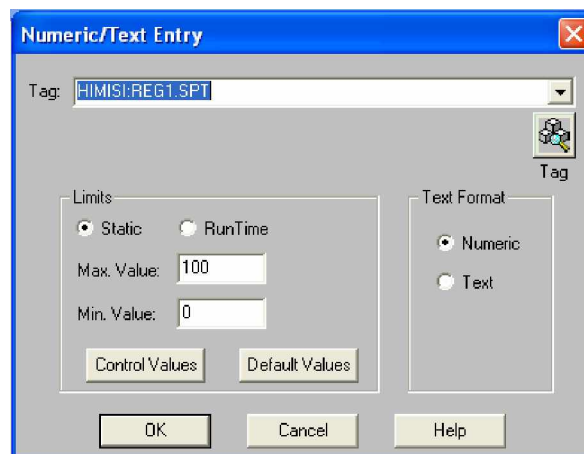
Obr. 7.14 Príklad nadefinovania možnosti aktualizácie textového poľa

Odosielanie údajov z textového poľa po ich zmene užívateľom

Odosielanie je použité (obr. 7.4) na nastavovanie parametrov regulátora a prenosu. Po vytvorení textového poľa, jeho označení pravým tlačidlom myši, som z ponuky (obr. 7.6) som zvolil **Configure_Objects**, zobrazila sa ponuka (obr. 7.11), pri ktorej som zvolil položku pridania akcie **Action** a z ponuky (obr. 7.15) som zvolil **Numeric/Text_Entry**. Zobrazilo sa okno (obr. 7.16) v ktorom som vybral formát zadávaného textu (čísla), definoval miesto, kam sa budú čísla zadané v tomto poli posielat', a obmedzil som ich rozsah.



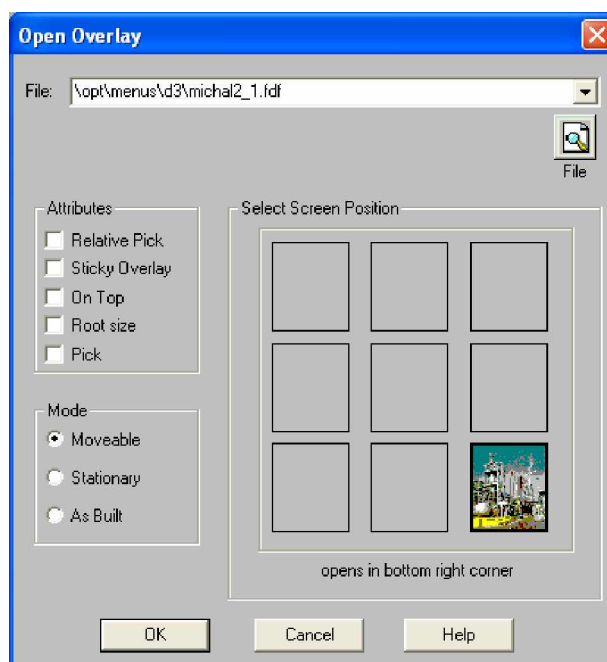
Obr. 7.15



Obr. 7.16 Príklad nadefinovania cieľa odosielania údajov

Otvorenie novej prekrývajúcej obrazovky

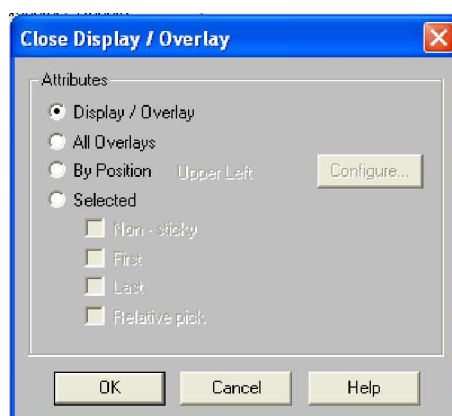
Je použité na obr. 7.1 (tlačidlo „Editacia parametrov“) a nadefinované na otvorenie okna s možnosťou zmenenia parametrov regulátora, rovnice (na obr. 7.4). Vytvorené tlačidlo som označil, zvolil ponuky **Configure_Objects – Action – Open_Overlay** zobrazilo sa okno na definovanie celej cesty umiestnenia vytvoreného a umiestneného okna, ktoré chceme otvoriť pri stlačení tlačidla (obr. 7.17), je možné vybrať miesto na obrazovke, kde bude nová obrazovka otvorená.



Obr. 7.17 Príklad definovania otvorenia prekryvajúcej obrazovky

Zatvorenie obrazovky

Táto akcia je použitá na zavretie prekryvajúcej obrazovky (na obr. 7.4 je táto akcia priradená tlačidlu „Zatvorenie obrazovky“). Akciu som definoval pre tlačidlo vybratím ponúk **Configure_Objects – Action – Close Display/Overlay**. Otvorila sa obrazovka (obr. 7.18).

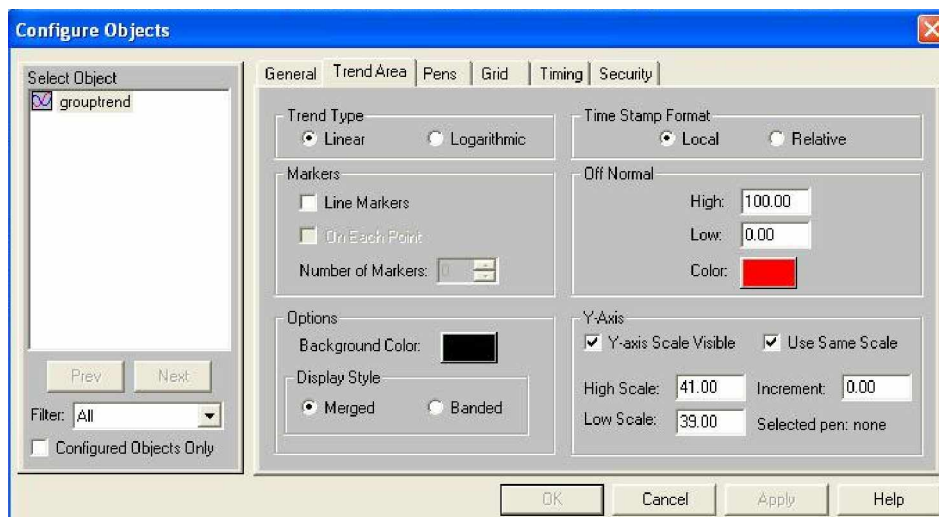


Obr. 7.18 Definovanie zatvorenia obrazovky

Úprava grafu:

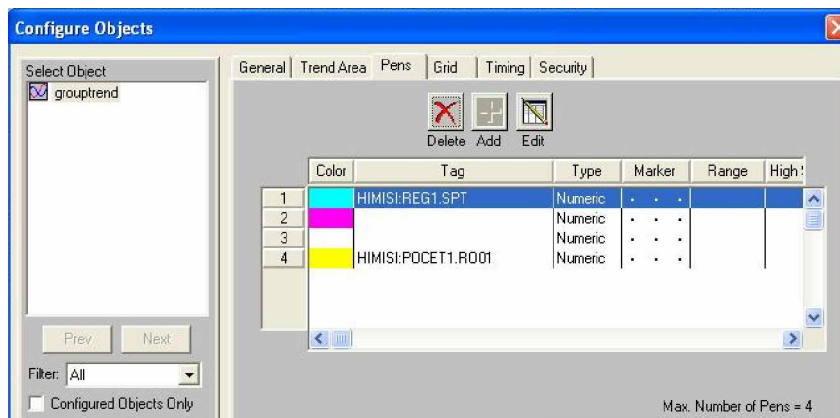
Po vybratí grafu z knižnice sa graf upravuje aby vyhovoval požiadavkám užívateľa.

- vo všeobecnej ponuke (General) je možné priradiť grafu meno
- v ponuke Mriežka (Grid) sa určuje viditeľnosť, počet a farba súradníc
- v ponuke Oblasť grafu (Trend Area) je možné určiť rozsah grafu, rozsah zobrazenej časti grafu (obr. 7.19)



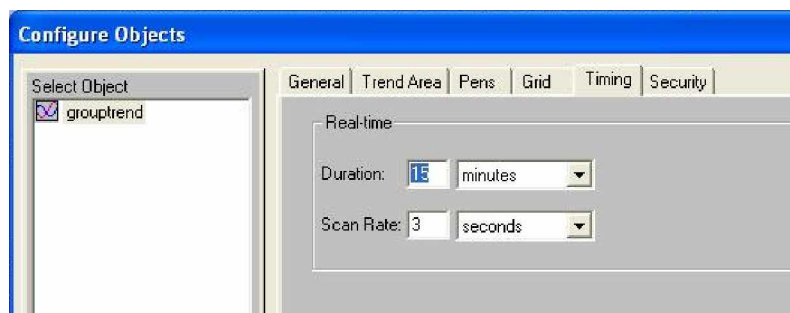
Obr. 7.19

- v ponuke Perá (Pens) sa vyberajú farby a definujú cesty ku umiestneniu parametrov (obr. 7.20)



Obr. 7.20 Príklad určenia farieb a umiestnenia parametrov

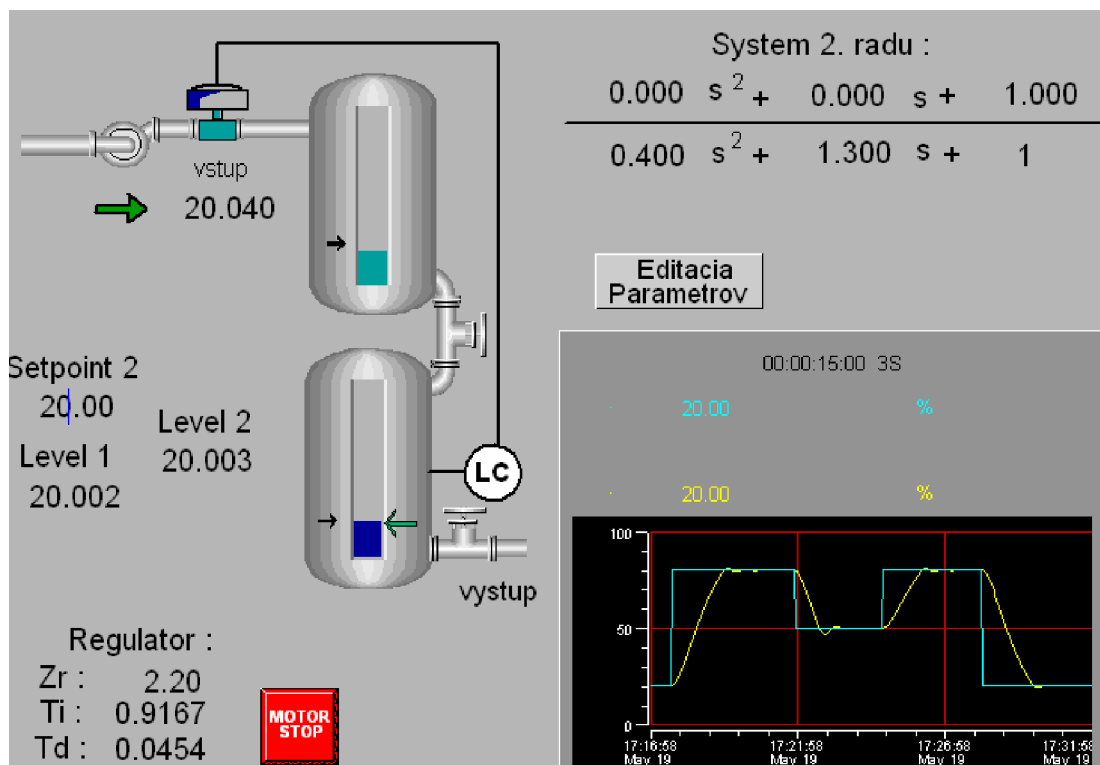
- v ponuke Časovanie (Timing) sa určuje rýchlosť snímania premenných a dĺžka časového úseku zobrazeného v grafe (obr. 7.21)



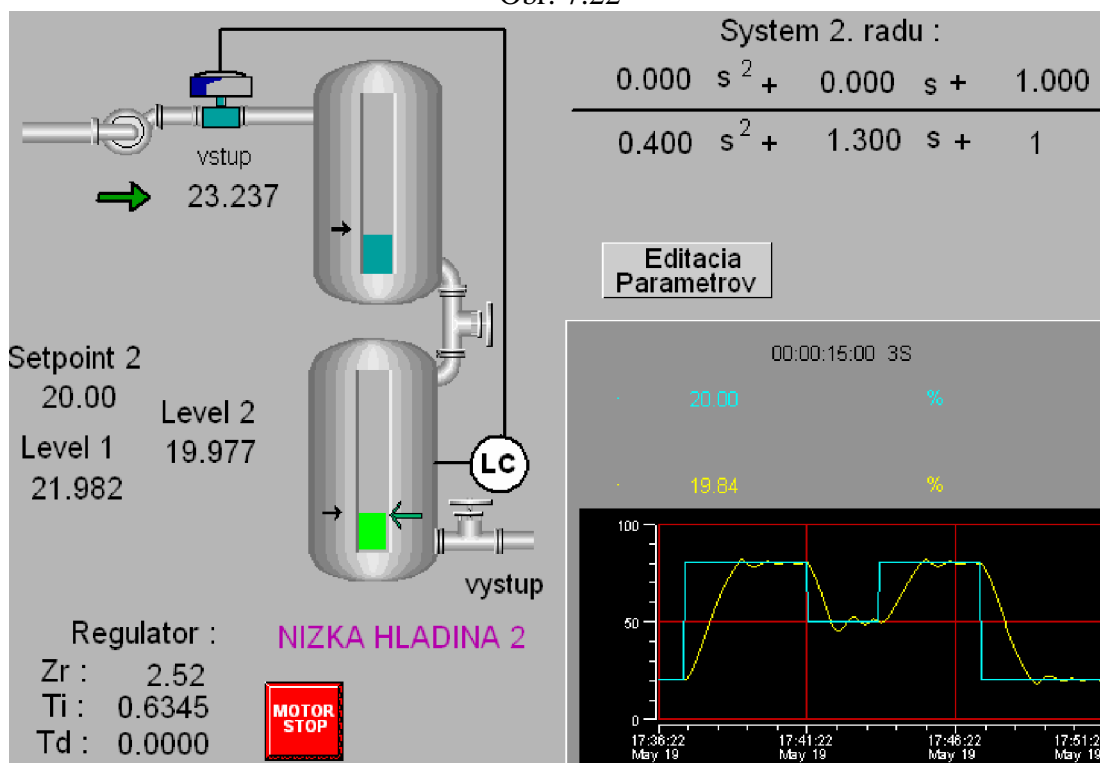
Obr. 7.21 Nastavenie časovania

7.3 Riadenie zásobníkov

Na riadenie zásobníkov kvapaliny som používal blok PIDA do ktorého som posielal parametre regulátora vypočítané pomocou metódy umiestnenia pólov (je zobrazené na obr. 7.22) a pomocou Naslinovej metódy (zobrazené na obr. 7.23).



Obr. 7.22



Obr. 7.23

Záver

Úlohou tejto diplomovej práce bolo vytvoriť simuláciu zásobníkov kvapaliny a riadenie týchto zásobníkov pomocou riadiaceho systému I/A Series FOXBORO.

V integrovanom riadiacom konfigurátore ICC som vytvoril compound do ktorého som vložil bloky LLAG, PIDA, SWCH CALCA a zadefinoval medzi nimi komunikáciu. Blok LLAG bol využitý na vytváranie signálov simulujúcich chovanie zásobníku kvapaliny v závislosti od nastavenia parametrov diferenciálnej rovnice s nulovými počiatočnými podmienkami. V bloku PIDA som nastavoval parametre regulátora navrhnuté metódou umiestnenia pólov. Blok PIDA som použil na vytváranie alarmov po definovaní úrovni alarmov a výšok hladiny v druhom zásobníku pri ktorých sa alarmy spúšťali. Blok SWCH bol použitý na prepínanie posielania výstupu regulátora a nulovej hodnoty do bloku LLAG. Prepínanie zabezpečoval parameter TOGGLE ktorý bol posielaný z obrazovky operátora.

V programe FoxView som vytvoril dve užívateľské obrazovky obsahujúce grafický priebeh výšky hladiny v druhom zásobníku a žiadanej výšky hladiny. Medzi obrazovkami som nadefinoval prepojenie pomocou ktorého sa otvárala nová obrazovka ktorá prekryvala pôvodnú. V oboch obrazovkách boli vytvorené dynamické objekty, ktoré čerpali údaje z parametrov blokov. Medzi ne patrili parametre regulátora, parametre diferenciálnych rovníc, výšky hladiny v zásobníkoch zobrazené číselne a graficky. Pri grafickom zobrazení som použil grafy a obrázky zásobníkov, v ktorých boli dynamicky zobrazené výšky hladín. V druhom zásobníku som nadefinoval zmenu farby hladiny ak sa vyskytol alarm definovaný v bloku PIDA.

V nasledujúcich prácach by bolo možné použiť všetky možnosti vizualizácie pomocou dynamických atribútov meniacich vzhľad obrazovky, možnosti systémových alarmov a vytvoriť adaptívne riadenie v riadiacom systéme FOXBORO.

Literatúra

- [1] www.alya.sk
- [2] I/A Series Electronic Documentation V6.5, Group Display Configurator
- [3] I/A Series Electronic Documentation V6.5, Integrated Control Configurator
- [4] I/A Series Electronic Documentation V6.5, Integrated Control Software Concepts
- [5] I/A Series Electronic Documentation V6.5, Integrated Control Block Descriptions
- [6] I/A Series Electronic Documentation V6.5, FoxDraw
- [7] I/A Series Electronic Documentation V6.5, FoxView

Príloha 1

Zoznam všetkých blokových funkčných typov z celého integrovaného riadiaceho software.

ACCUM	(Accumulator) zaraďuje reálny vstupný signál a kalibruje ho na tvorbu reálneho výstupného počtu celkového spracovania.
AIN	(Analog Input) podporuje jediný vstup z analógového typu FBM. Podmienky sú na stav, meranie, vnútenie stavu a filtráciu vstupu, alarmovanie stavu hardware a výstupnej hodnoty.
AINR	(Redundant Analog Input) podporuje jediný vstup z dvoch analógových typov FBM. Podmienky sú na stav, meranie, vnútenie stavu, filtráciu stavu a alarmovanie stavu hardware a výstupnej hodnoty.
ALMPRI	(Alarm Priority Change) je použitý na dynamický prevod priority stupňa alarmu.
AMSSEC	(Gas Chromatograph Secondary Block) je použitý v spolupráci s ECB plynového chromatografu („MSPRI“). Poskytuje prídavný vstup do GC a obsahuje premenné pre individuálne aplikácie.
AOUT	(Analog Output) poskytuje funkciu Auto/Manual s posunutím. Posúva, vnucuje stav a vstupné podmienky a vedie jediný výstupný bod vo vnútri analógového typu FBM.
AOUTR	(Redundant Analog Output) poskytuje funkciu Auto/Manual s posunutím. Posúva, vnucuje stav a podmienky výstupu a vedie jediný bod výstupu cez dvojité pár statických analógových typov FBM.
BIAS	Vytvára výstup, ktorý je sumou dvoch vstupných hodnôt, MEAS a BIAS, každá z nich môže byť nezávisle kalibrovaná. Blok podporuje meranie alarmových správ a kaskádne spúšťanie. Nepodporuje výstup alarmových správ.
BIN	(Binary Input) poskytuje riadiacu stratégiu so schopnosťou čítať jednoduché binárne hodnoty, má dva stavy ako ON a OFF alebo START a STOP z adresy v asociovanom Fieldbus Modul (FBM) alebo označenom bode. Je použitý v Distributed Control Interface

	(DCI) (DCI bloky podporujú pripojiteľnosť riadiacej stanice I/A Series do rozličných zariadení cez hlavné účelové rozhranie) a v PLC rozhraní.
BINR	(Binary Input Redundand) získava jednu binárnu hodnotu z externého zariadenia. Zdroj tejto hodnoty môže byť určený dvoma alebo troma statickými vstupmi z toho istého, alebo iných zariadení. Výber algoritmu blokom rozhoduje, ktorá z hodnôt vstupu je zobrazená do riadiacej stratégie ako výstup bloku. BINR je použitý v DCI.
BLNALM	(Boolean Alarm) poskytuje nezávislú zmenu stavu alarmových správ pre každý z ôsmich vstupov Boolovského typu.
BOOL	(Boolean Data Variable) poskytuje schopnosť vytvorenia nastaviteľnej a konfigurovateľnej booleovskej hodnoty na použitie iným riadiacim blokom.
BOUT	(Binary Output) posiela jednu binárnu hodnotu na adresu externého zariadenia. Nepretržite hlási I/A systému každú zmenu urobenú zariadením na tejto adrese. Je použitý v DCI a PLC rozhraní.
CALC	(Calculator) poskytuje do 50 aritmetických a logických operácií následne vykonaných. Má schopnosť programovateľnej vedeckej kalkulačky.
CALCA	(Advanced Calculation Block) pridáva dvojité operačnú výkonnosť mnohým matematickým a logickým výpočtovým operáciám.
CHARC	(Characterized) mení reálny vstup na reálny výstup použitím vyhľadávania po častiach lineárnych úsekov v tabuľke. Môže byť použitých 20 úsekov.
CIN	(Contact Input) podporuje jediný vstupný bod z digitálneho vstupného typu FBM. Poskytuje teda nastavenie prevrátenia vstupu.
COUT	(Contact Out) podporuje jediný vstup do digitálneho typu FBM. Podporuje výstupné nastavenie pulzovania.
DEP	(Dependent Sequence) obsahuje užívateľom naprogramované inštrukcie, ktoré ovplyvňujú compound alebo parametre bloku, alebo spoločné premenné. Môže aktivovať ďalšie bloky spolu s poslaním správ na obrazovku a do histórie. Vykonanie závislého sekvenčného bloku je automaticky spomalené, kým je spustený hocijaký vykonávajúci sekvenčný blok v tom istom compounde.

DGAP	(Diferencial Gap) dáva na výstup dve výstupné diskkrétne hodnoty, môžu byť použité na ON/OFF riadenie dvoj- alebo troj-stavového riadeného hnacieho zariadenia. Výstup je závislý od rozdielu medzi meraním, setpointom a nastaviteľnou chybou GAP. Tento blok nepodporuje kaskádnu inicializáciu.
DPIDA	(Distributed Controller) pripája riadiaci procesor do algoritmu PIDA typu realizovanom v FBM(04, 17, 37 alebo 22).
DSI	(Display Station Interface) poskytuje prepojenie medzi panelom monitora stanice a komponentmi a blokmi ktoré uskutočňujú riadiacu stratégiu.
DTIME	(Deadtime) oneskoruje vstup a špecifickú dĺžku času pred využitím na výstupe. Je poväčšine použitý na simuláciu dopravného oneskorenia a na kompenzáciu dopredných signálov.
EVENT	(Sequence of Event) poskytuje oznamovanie správ pre postupnosť zmeny stavu udalosti zistenej v kontaktnom vstupe FBM.
EXC	(Exception Sequence) obsahuje užívateľom programované inštrukcie, ktoré môžu narábať s parametrami komponentu, bloku, alebo so zdieľanými premennými. Môže aktivovať ďalšie sekvenčné a sledujúce bloky spolu s poslaním správy do obrazovky a histórie.
FBTUNE	(Feedback Self-Tuner) je použitý na adaptívne nastavenie proporcionálneho zosilnenia, integračného času, derivačného času, oneskorenia, filtra setpointu PIDA bloku.
FFTUNE	(Feedforward Self-Tuner) je použitý na adaptívne nastavenie dopredného kompenzátora pre PIDA blok.
GDEV	(General Device) poskytuje riadenie otvorenia/uzavretia ventilu (ovládaného motorom alebo vzduchom) a riadenie štart/stop motorového okruhu.
IIN	(Integer Input) je DCI blok. Môže získať jednu celočíselnú hodnotu z externého zariadenia. Aktuálne prijatie a spracovanie tejto hodnoty je zavedené vzhľadom na podmienky voľby simulácie a módu auto/manuál bloku IIN.
IND	(Independent Sequence) obsahuje príkaz naprogramovaný užívateľom, ktorý môže narábať s parametrami bloku, komponentu, alebo so zdieľanými premennými. Môže spustiť iné sekvenčné a sledujúce

bloky spolu so zaslaním správy na obrazovku a do histórie. Nezávislý sekvenčný blok nepôsobí na vykonávanie iných sekvenčných blokov, ani nepôsobí vykonávanie iných blokových operácií nezávislých sekvenčných blokov.

IOUT	(Integer Output) je DCI blok. Môže poslať jednu celočíselnú hodnotu do poľa zariadenia. Kontinuálne hlási hocijaké zmeny urobené zariadením. Tieto hlásenia sú urobené pre hodnotu na tej istej adrese.
LIM	(Limiter) poskytuje umiestnenie a rýchlostný obmedzovač funkcie pre reálny vstupný signál.
LLAG	(Lead-Lag) vykonáva dynamickú kompenzáciu signálu dynamickým vedením výstupu alebo zaostávaním za vstupom.
LOGIC	poskytuje 15 následne splnených logických funkcií.
LONG	(Long Integer Data Variable) poskytuje schopnosť vytvorenia nastaviteľnej a konfigurovateľnej dátovej hodnoty pre použitie iným riadiacim blokom.
MAIN	(Multiple Analog Input) podporuje do 8 vstupov z analógového typu FBM. Je poskytnutý vnútorný kanál pre teplotný odkaz senzora.
MATH	(Mathematics) poskytuje súbor matematických funkcií potrebný pre špecializované riadenie.
MCIN	podporuje do 16 digitálnych výstupov do digitálneho typu FBM.
MDACT	(Motor Driven Actuator Controller) poskytuje schopnosť pre vysokú rozlíšiteľnosť riadenia procesov riadenia trojstavovými zariadeniami. Blok podporuje kaskádnu inicializáciu.
MEALM	(Measure Alarm) slúži ako signalizátor alarmu na aktiváciu alarmového mechanizmu I/A Series do objavenia podmienky alarmu externým zariadením.
MON	(Monitor Sequence) sleduje do 16 podmienok procesu. Sleduje hodnoty parametrov a boolean výrazy a spúšťa výnimky, závislé alebo nezávislé bloky.
MOVLV	(Motor Operated Valve) ovláda dve súvisiace výstupné spojenia ktoré otvárajú alebo zatvárajú klapku na základe prírastku. Podporuje voliteľnú spätnú väzbu z jedného alebo dvoch spojení pre nezhodu alarmovania.

MRIN	(Multiple Real Input) podporuje riadiacu stratégiu so schopnosťou čítania ôsmych analógových hodnôt zo súvisiacich adries v programovateľnom logickom regulátore (PLC). Tento blok je použitý v PLC prepojení.
MROUT	(Multiple Real Output) podporuje schopnosť zapisovania ôsmych analógových hodnôt do súvisiacich adries v programovateľnom logickom regulátore (PLC). Tento blok je použitý ako PLC prepojenie.
MSG	(Message Generator) tvorí stavové zmeny správ do zmeny jeho boolean vstupov.
MTR	(Motor Controller) podporuje obe dvoj- a troj- linkových riadiacich funkcií motora.
MVC	(MultiVariable Controller) pôsobí ako rozhranie do riadiaceho procesora vložení MVC ako predchádzajúceho mechanizmu pre odborné riadiace modely a konfiguračné parametre.
MVL	(MultiVariable Loop) získava riadiace premenné (CV) meraním z blokov ako AIN a zapisuje spracované premenné (MV) kontrolné setpointy do iných regulačných blokov (PIDA, DPIDA alebo RATIO).
OUTSEL	(Output Select) riadi stratégie, ktoré žiadajú vyšší alebo nižší z dvoch vstupných signálov na vybratie ako konečného výstupného signálu do procesu, pokiaľ poskytnutie vhodných dát na zabránenie integrálnej akcie pred zrušením v bloku ktorý obsahuje nezvolený signál. Blok podporuje rozdelenú kaskádnu inicializáciu, každú pre dva predchádzajúce bloky.
PACK	(Packed Long Data Variable) poskytuje schopnosť vytvorenia nastaviteľnej a konfigurovateľnej dátovej premennej pre použitie iným regulačným blokom.
PACKIN	(Packed Input) načítava do 32 bitov diskretných dát zo skupiny adries v programovateľnom logickom regulátore (PLC). Každý bit reprezentuje binárnu hodnotu, ktorá má protichodné stavy ako ON a OFF alebo START a STOP. Tento blok je použitý v PLC pripojení.
PAKOUT	(Packed Output) zapisuje do 32 bitov diskretných dát do skupiny adries v programovateľnom logickom regulátore (PLC). Každý bit

	reprezentuje binárnu hodnotu, ktorá má protichodné stavy ako ON a OFF alebo START a STOP. Blok je použitý v PLC pripojení.
PATALM	(Pattern Alarm) porovnáva vzťah do 16 boolean vstupov podľa užívateľom určenej schémy.
PATT	(Pattern Match) poskytuje schopnosť upravenia modelu pre 16 bitové modely.
PID	(Proportional, Integral, Derivate) poskytuje funkcie tradičného interagujúceho trojprvkového regulátora. Blok podporuje kaskádnu inicializáciu.
PIDA	(Proportional, Integral, Derivate, Advanced) realizuje kontinuálny PID alebo spätnú väzbu s dopravným oneskorením a súčtové a násobiace dopredné riadenie analógovej slučky, v prípade pokročilejších vlastností mimo tých od PID a PIDX blokov.
PIDE	(PID with EXACT™) podporuje tú istú schopnosť ako PID blok s doplnkom presného samo nastavujúceho algoritmu.
PIDX	(PID Extended) poskytuje tú istú schopnosť ako PID blok s voliteľnou schopnosťou pre nelineárnu kompenzáciu zosilnenia, snímacieho módu a predbežným riadiacim predpätím.
PIDXE	(PID Extended with EXACT) pridáva algoritmus presnosti do bloku PIDX.
PLB	(Programmable Logic Block) je funkčne zložený z násobiacich PLC blokov, spojených s jediným PLC. Je obzvlášť vhodný keď je veľký počet bodov v PLC s rôznymi typmi dát pripojený do stanice I/A Series.
PLSOUT	(Pulse Output) umožňuje riadiacu stratégiu alebo operátor na výstup ON a OFF alebo START a STOP príkazov cez prechodne impulzové výstupy na dvoch oddelených líniiach, jednej pre každý stav. Blok je použitý v PLC pripojení.
PTC	(Proportional Time Controller) uskutočňuje funkciu proporcionálneho ON/OFF regulátora. Nepodporuje kaskádnu inicializáciu.
RAMP	uskutočňuje univerzálnu rampovú funkciu.
RATIO	počíta výstup, ktorý je kalibrovaný násobením meraného vstupu so vstupným pomerom setpointu. Blok podporuje kaskádnu inicializáciu.

REAL	(Real Data Variable) poskytuje schopnosť vytvorenia nastaviteľnej a konfigurovateľnej dátovej hodnoty pre použitie iným riadiacim blokom.
REALM	(Real Alarm) voliteľne podporuje tri typy alarmov: <ul style="list-style-type: none"> • Vysoký a nízky absolútny alarm merania • Rýchlosť zmeny alarmu merania • Vysokú a nízku odchýlku alarmu rozdielu medzi meraním a setpointom
RIN	(Real Input) prijíma jednu analógovú hodnotu z externého zariadenia. Poskytuje túto hodnotu po vstupnom spracovaní v parametri RINP. RIN je použitý v DCI a v PLC pripojení.
RINR	(Real Input, Redundant) získava jednu binárnu hodnotu z externého zariadenia. Zdroj hodnoty môže byť označený buď dvoma alebo tromi statickými vstupmi z toho istého alebo z rozdielnych zariadení. Každý zo statických vstupov je nezávisle kalibrovaný, limitovaný a prevedený na inžinierske jednotky predtým ako je vyvolaný výberový algoritmus bloku na určenie, ktorý z dvoch alebo troch vstupov je nastavený do parametra RINP. RINP je použitý v bloku DCI.
ROUT	(Real Output) posíla jednu analógovú hodnotu do adresy v externom zariadení. Spojito oznamuje každú zmenu urobenú zariadením na hodnote na tejto adrese. Je použitý v DCI a v PLC pripojení.
SIGSEL	(Signal Selector) skúša do osem vstupov a vytvára výstup závislý na relatívnej voľbe výberu.
STALM	(State Alarm) slúži ako signalizátor alarmu na aktiváciu alarmového mechanizmu I/A Series na podmienkach alarmu objavených externým zdrojom ako HTG.
STATE	Blok State vystupuje vybraných 16-bitových patterns.
STRIN	(String Input) blok je blokom v Distributed Control Interface (DCI). STRIN získava jednu hodnotu reťazca z externého zariadenia.
STRING	(String Data Variable) poskytuje schopnosť vytvorenia nastaviteľnej a konfigurovateľnej dátovej hodnoty pre použitie iným riadiacim blokom.

SWCH	(Switch Position Selector) vyberá jeden z dvoch nezávislých vstupov. Každý vstup môže byť reálne číslo, celé číslo alebo boolean premenná. Podporuje kaskádnú inicializáciu.
TIM	(Timer Sequence) sleduje čas pokiaľ sú vykonávané riadiace stratégie.
VLV	(On-Off Valve Control) je typicky použitý na vykonávanie dvoch spojených výstupných spojení, ktoré otvárajú alebo zatvárajú ventil na základe ON/OFF.

Príloha 2

Detailné nastavenie parametrov v blokoch, ktoré som upravil v ICC : LLAG, PIDA, SWCH a CALCA.

SWCH

Block Definition:			
Name:	SWITCHB	HSCI2	100.0
Type:	SWCH	LSCI2	0.0
DESCRP		DELT12	1.0
PERIOD	1	EI2	%
PHASE	0	TOGGLE	0
LOOPID		HSC01	100.0
PROPT	0	LSC01	0.0
EROPT	1	DELT01	1.0
INP1	0.0	E01	%
HSCI1	100.0	MA	0
LSCI1	0.0	INITMA	1
DELT11	1.0	BTIME1	0.0
EI1	%	BTIME2	0.0
INP2	:REG1.OUT	PRIBLK	0
		INITI	0
		BCALCI	0.0

LLAG

Block Definition:			
Name:	ZASOB_2	HSC01	100.0
Type:	LLAG	LSC01	0.0
DESCRP		DELT01	0.001
PERIOD	1	E01	%
PHASE	0	HOLIM	100.0
LOOPID		LOLIM	0.0
LLOPT	0	BIAS	0.0
MEAS	HIMISI:ZASOB_1.OUT	HSCI2	100.0
HSCI1	100.0	LSCI2	0.0
LSCI1	0.0	DELT12	0.001
DELT11	0.001	EI2	%
EI1	%	MA	1
PROPT	0	INITMA	1
LGAIN	0.0	MCLOPT	0
LAGTIM	0.8	FOLLOW	0
LAG2	0.0		

CALCA

Block Definition:		STEP01 MUL RI01 RI02
Name: POCET1		STEP02 OUT R001
Type: CALCA		STEP03 MUL RI03 RI04
DESCRP		STEP04 OUT R002
PERIOD	1	STEP05 ADD RI03 RI04
PHASE	0	STEP06 OUT R003
LOOPID		STEP07 MUL RI05 RI03
RI01	:ZASOB_2.OUT	STEP08 OUT M01
RI02	0.0	STEP09 MUL RI06 RI04
RI03	:ZASOB_1.LAGTIM	STEP10 OUT M02
RI04	:ZASOB_2.LAGTIM	STEP11 ADD M02 M01
RI05	:ZASOB_1.LGAIN	STEP12 OUT M03
RI06	:ZASOB_2.LGAIN	STEP13 MUL M02 M01
RI07	0.0	STEP14 OUT M04
RI08	0.0	STEP15 MUL M03 RI02
BI01	0	STEP16 OUT R004
		STEP17 MUL M04 RI02
BI15	0	STEP18 OUT M05
BI16	0	STEP19 END
II01	0	STEP20
II02	0	STEP21
LI01	0	STEP22
LI02	0	STEP23
MA	1	STEP24
INITMA	1	STEP25
TIMINI	0	STEP26
M01	0.0	STEP27
M02	0.0	STEP28

PIDA

Block Definition:	
Name:	REG1
Type:	PIDA
DESCRP	
PERIOD	1
PHASE	0
LOOPID	
MEAS	:ZASOB_2.OUT
HSCI1	100.0
LSCI1	0.0
DELT11	0.1
EI1	%
NLNBLK	0
SPT	0.0
SPCLMP	0
SPHLIM	100.0
SPLLIM	0.0
STRKOP	0
SPROPT	0
SPRATE	0.0
SPTARG	0.0
SPRAMP	0
RSP	0.0
LR	0
INITLR	2
LOCSP	0
LOCSW	0
REMSW	0
MODOPT	5
INCOPT	0
PBAND	1000.0
INT	100.0
DERIV	0.0
KD	10.0
SPLLAG	1.0
DTIME	0.0
FILTER	0.0
NONLOP	0
HZONE	100.0
LZONE	100.0

KZONE	1.0	DELTO1	1.0
SPLCOP	0	EO1	%
SPLRDY	0	OSV	2.0
TSAMPL	0.0	HOLIM	100.0
BIAS	0.0	LOLIM	0.0
HSCI2	100.0	LIMOPT	1
LSCI2	0.0	MCLOPT	0
DELT12	1.0	BATCHO	0
EI2	%	PRLOAD	0.0
BBIAS	0.0	TRACK	0.0
KBIAS	1.0	TRKENL	0
BTRKOP	0	HOLD	0
MULTIN	100.0	PRIBLK	0
HSCIN	100.0	INITI	0
LSCIN	0.0	BCALCI	0.0
EIN	%	FBK	0.0
HSC01	100.0	MA	1
LSC01	0.0	INITMA	1

MANFS	0	MEASGR	1
MBADOP	0	DALOPT	0
CEOPT	1	DEVTIM	0
PROPT	0	HDALIM	20.00
MANSW	0	HDATA	
AUTSW	0	LDALIM	-20.00
MANALM	1	LDATA	
INHOPT	0	DEVADB	0.0
INHIB	0	DEVPRI	2
INHALM	0	DEVGRP	1
MEASNM		HHAOPT	1
MALOPT	1	HHALIM	90.0
MEASHL	85.0	HHATA	
MEASHT	MEASHL.A	LLALIM	10.0
MEASLL	20.0	LLATA	
MEASLT	MEASLL.A	HHAPRI	1
MEASDB	0.0	HHAGRP	1
MEASPR	2	OALOPT	1

OUTNM	
HOALIM	100.0
HOATA	
LOALIM	0.0
LOATA	
OUTADB	0.0
OUTPRI	3
OUTGRP	1
FLBOPT	0
INITSE	0
SUPGRP	1
SUPOPT	0