UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

Laboratorní úloha "PLC Tecomat FOXTROT - OPC server"

Petr Kynych

Bakalářská práce

2013

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: **2012/2013**

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Petr Kynych**
Osobní číslo:	**I09338**
Studijní program:	**B2646 Informační technologie**
Studijní obor:	**Řízení procesů**
Název tématu:	**Laboratorní úloha "PLC Tecomat FOXTROT - OPC server"**
Zadávající katedra:	**Katedra řízení procesů**

Zásady pro vypracování:

Cílem je vytvořit laboratorní úlohu "PLC Tecomat FOXTROT - OPC server".
Obsah teoretické části:
Teorie logického řízení. Programovatelné logické automaty a inteligentní relé - hardware a programování. OPC server.
Obsah implementační části:
Instalace softwaru pro programování PLC, simulaci a vizualizaci úlohy logického řízení. Vytvoření laboratorní úlohy - jak po hardwarové tak i softwarové stránce.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

L. Šmejkal, M. Matrinásková, PLC a automatizace 1, Základní pojmy, úvod do programování. BEN - technická literatura, Praha 1999
L. Šmejkal, PLC a automatizace 2, Sekvenční logické systémy a základy fuzzy logiky. BEN - technická literatura, Praha 2005
M. Martinásková, Programovací jazyky pro PLC. Automatizace, ročník 47, číslo 6, strana 380, 2004

Vedoucí bakalářské práce: **Ing. Daniel Honc, Ph.D.**
Katedra řízení procesů

Datum zadání bakalářské práce: **9. prosince 2012**
Termín odevzdání bakalářské práce: **10. května 2013**

Univerzita Pardubice
Česká republika

L.S.

prof. Ing. Simeon Karamazov, Dr.
děkan

Ing. Daniel Honc, Ph.D.
vedoucí katedry

V Pardubicích dne 29. března 2013

Prohlášení autora

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 1. 5. 2013 Petr Kynych

Poděkování

Rád bych touto cestou vyjádřil poděkování Ing. Danielovi Honcovi za jeho cenné rady, ochotu a trpělivost při vedení mé bakalářské práce.

Anotace

V práci je popsána teorie logického řízení, programovatelné logické automaty, programovací jazyky programovatelných logických automatů a standard OPC. Cílem práce je vytvořit laboratorní úlohu s automatem TecomatFoxtrot CP-1005 pracující s OPC. K tomu je využit model mísicí jednotky.

Klíčová slova

logické řízení, OPC, PLC

Title

Laboratory task "TECOMAT FOXTROT - OPC Server"

Annotation

The work describes the theory of logic control, programmable logic controllers, programming of logical controllers and OPC standard. The aim is to create a laboratory task realized by Tecomat Foxtrot CP-1005 working with OPC. It's done by using the model of mixing unit.

Keywords

logic control, OPC, PLC

Obsah

Seznam zkratek **9**

Seznam obrázků **10**

Seznam tabulek **11**

1 Úvod **12**

2 Teoretická část **13**

2.1 Logické řízení 13
2.1.1 Logické funkce 13
2.1.2 Pravdivostní tabulka 17
2.1.3 Booleova algebra 18
2.1.4 Disjunktivní a konjunktivní normální forma 19
2.1.5 Karnaughova mapa 20

2.2 Realizace logických obvodů 21
2.2.1 Kombinační logické obvody 21
2.2.2 Sekvenční logické obvody 21

2.3 Programovatelné logické automaty – PLC 24
2.3.1 Rozdělení PLC 25
2.3.2 Průběh programu v PLC 26
2.3.3 Programovací jazyky PLC 27

2.4 OPC 30
2.4.1 Standard OPC 31
2.4.2 Architektura OPC 32

3 Praktická část **34**

3.1 Tecomat Foxtrot 34

3.2 Mosaic 34
3.2.1 Tvorba programu 35
3.2.2 Přiřazení aliasů vstupům a výstupům 36
3.2.3 Vytvoření exportního souboru pro OPC server 36

3.3 Teco OPC 37
3.3.1 Konfigurace DCOM 38

3.4 Promotic 45

3.5 Laboratorní úloha – Mísicí jednotka OPC server 48
3.5.1 EDUtec 48
3.5.2 EDU-mod Mísicí jednotka 49
3.5.3 Laboratorní úloha 50

4 Závěr **55**

Literatura **56**

Seznam zkratek

OLE – objectlinking and embedding

OPC –OLE forprocesscontrol

PLC– programmablelogiccontroller

PC – personalComputer

COM – commonobject model

DCOM – distributed COM

HW – hardware

RAM –randomaccessmemory

CMOS – Complementary Metal–Oxide–Semiconductor

SCADA – supervisorycontrol and data acquisition

CIB –CommonInstallation Bus

Seznam obrázků

Obrázek 1 – Hradlo AND 16
Obrázek 2 – Hradlo OR 16
Obrázek 3 – Hradlo NAND 16
Obrázek 4 – Hradlo NOR 17
Obrázek 5 – Karnaughova mapa 20
Obrázek 6 – Klopný obvod R-S sestavený z hradel NOR 22
Obrázek 7 – klopný obvod D z hradel NAND 23
Obrázek 8 – princip zapojení klopného obvodu J-K z hradel NAND 24
Obrázek 9 – Blokové schéma vnitřní struktury PLC 25
Obrázek 10 – Běh programu v PLC 27
Obrázek 11 – jazyk LD 28
Obrázek 12 – jazyk FBD 28
Obrázek 13 – jazyk IL[5] 29
Obrázek 14 – jazyk ST[5] 29
Obrázek 15 – komunikace bez OPC 30
Obrázek 16 – komunikace s využitím OPC 31
Obrázek 17 – Základní Modul CP – 1005[8] 34
Obrázek 18 – Prázdný program v jazyce LD 35
Obrázek 19 – nastavení vstupů a výstupů 36
Obrázek 20 – nastavení generování exportních souborů 37
Obrázek 21 – Teco OPC server 38
Obrázek 22 – Výchozí vlastnosti 39
Obrázek 23 – přístupová oprávnění 40
Obrázek 24 – Spouštěcí a aktivační oprávnění 40
Obrázek 25 – Oprávnění ke změně konfigurace 41
Obrázek 26 – povolení pravidel 42
Obrázek 27 – přidání nového pravidla 42
Obrázek 28 – Typ nového pravidla 43
Obrázek 29 – Cesta k programu 43
Obrázek 30 – povolit připojení 44
Obrázek 31 – platnost pravidla 44
Obrázek 32 – Dokončení pravidla 45
Obrázek 33 – promotic editor aplikace 46
Obrázek 34 – PmOpcClient 46
Obrázek 35 – PmOpcClientData 47
Obrázek 36 – graficky editor a nastavení vlastností tlačítka 48
Obrázek 37 – EDUtec 49
Obrázek 38 – EDU-mod Mísicí jednotka 49
Obrázek 39 – volba nádrží a omezení násobného spuštění 51
Obrázek 40 – plnění nádrží po zvolenou hladinu 51
Obrázek 41 – ovládání mísicí nádoby 52
Obrázek 42 – Konfigurace TecoOPC 52
Obrázek 43 – PmiSlider 53
Obrázek 44 - PmiBar 54
Obrázek 45 – Výsledná vizualizace 54

Seznam tabulek

Tabulka 1 – Logické proměnné 13
Tabulka 2 – Falsum 13
Tabulka 3 – Negace 14
Tabulka 4 – Aserce 14
Tabulka 5 – Verum 14
Tabulka 6 – Logické funkce dvou proměnných[1] 15
Tabulka 7 – Pravdivostní tabulka 17
Tabulka 8 – Pravdivostní tabulka – příklad 20
Tabulka 9 – Stavy klopného obvodu R-S 22
Tabulka 10 – Stavy klopného obvodu D 23
Tabulka 11 – Stavy klopného obvodu J-K 24
Tabulka 12 – Přiřazení vstupů a výstupů modulu k PLC 50

1 Úvod

Cílem bakalářské práce je vytvořit laboratorní úlohu zaměřenou na standard OPC s použitím PLC automatu Tecomat.

Teoretická část práce popisuje teorii logického řízení, programovatelné automaty a standard OPC. V praktické části je popsán logický modul Tecomat Foxtrot CP – 1005, programovací prostředí Mosaic, OPC server Teco a program Promotic. Dále je zde popsán výukový systém EDUtec a na závěr je vytvořena úloha pracující s modelem mísicí jednotky.

2 Teoretická část

2.1 Logické řízení

Logické řízení je činnost, při které se pomocí logického obvodu řídí proces tak, aby se dosáhlo předem stanoveného cíle.

Spojité fyzikální veličiny obecně mohou nabývat nekonečného počtu hodnot. Pro logické řízení je třeba libovolného konečného počtu hodnot, které ze spojité informace získáme například vhodným navzorkováním. Na takto získaná data již můžeme použít pravidla Logické algebry, která jsou určena k popisu vztahu mezi jednotlivými logickými proměnnými. Zvláštním druhem logických proměnných jsou dvouhodnotové proměnné, které nabývají hodnot 0 a 1[1].

Tabulka 1 – Logické proměnné

0	1
L	H
Vypnuto	Zapnuto
Ne	Ano
F(false)	T(true)

Logická algebra založená na stavech 0 a 1 se nazývá Booleova algebra.

2.1.1 Logické funkce

Logické funkce jedné proměnné

Logické funkce jedné proměnné jsou nejjednodušším případem logických funkcí.

Falsum pro libovolné x je y rovno 0.

Tabulka 2 – Falsum

x	**y**
0	0
1	0

Negace výstupní veličina je opakem vstupní.

Tabulka 3 – Negace

x	**y**
0	1
1	0

Aserce výstupní funkce má vždy stejnou hodnotu jako vstupní (opakuje se).

Tabulka 4 – Aserce

x	**y**
0	0
1	1

Verum pro libovolné x je y rovno 1

Tabulka 5 – Verum

x	**y**
0	1
1	1

Prakticky se ovšem z těchto čtyř funkcí využívá jen jedna, **negace.**

Logické funkce dvou proměnných

Logických funkcí dvou proměnných je šestnáct, jak je patrné z tabulky 6.

Tabulka 6 – Logické funkce dvou proměnných[1]

1. falsum		
X1	X2	y
0	0	0
1	0	0
0	1	0
1	1	0

2. konjunkce		
X1	X2	y
0	0	0
1	0	0
0	1	0
1	1	1

3. inhibice		
X1	X2	y
0	0	0
1	0	0
0	1	1
1	1	0

4. aserce		
X1	X2	y
0	0	0
1	0	0
0	1	1
1	1	1

5. inhibice		
X1	X2	y
0	0	0
1	0	1
0	1	0
1	1	0

6. aserce		
X1	X2	y
0	0	0
1	0	1
0	1	0
1	1	1

7. dilema		
X1	X2	y
0	0	0
1	0	1
0	1	1
1	1	0

8. disjunkce		
X1	X2	y
0	0	0
1	0	1
0	1	1
1	1	1

9. negace log. součtu		
X1	X2	y
0	0	1
1	0	0
0	1	0
1	1	0

10. ekvivalence		
X1	X2	y
0	0	1
1	0	0
0	1	0
1	1	1

11. negace		
X1	X2	y
0	0	1
1	0	0
0	1	1
1	1	0

12. implikace		
X1	X2	y
0	0	1
1	0	0
0	1	1
1	1	1

13. negace		
X1	X2	y
0	0	1
1	0	1
0	1	0
1	1	0

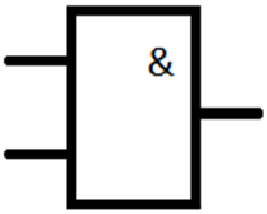
14. implikace		
X1	X2	y
0	0	1
1	0	1
0	1	0
1	1	1

15.neg. log součinu		
X1	X2	y
0	0	1
1	0	1
0	1	1
1	1	0

16.verum		
X1	X2	y
0	0	1
1	0	1
0	1	1
1	1	1

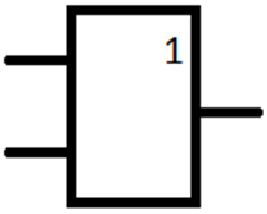
Prakticky se ale používají jen čtyři.

a) Konjunkce, logický součin neboli AND je charakteristický tím, že hodnota výstupu je 1 pouze, když jsou oba vstupy (platí obecně pro libovolný počet vstupů) v hodnotě 1. Jako operátor této funkce se používá znak ●



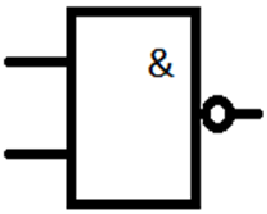
Obrázek 1 – Hradlo AND

b) Disjunkce, logický součet neboli OR nabývá hodnoty 1, když je alespoň jeden vstup v hodnotě 1. Jako operátor této funkce se používá znak + nebo v.



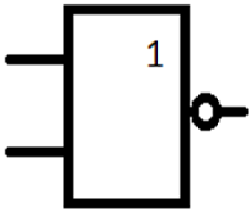
Obrázek 2 – Hradlo OR

c) Negace konjunkce, negovaný logický součin neboli NAND má hodnotu na výstupu 0 pokud jsou hodnoty obou vstupů v hodnotě 1.



Obrázek 3 – Hradlo NAND

d) Negace disjunkce, negovaný logický součet neboli NOR má hodnotu 1 na výstupu, právě když jsou oba vstupy v hodnotě 0.



Obrázek 4 – Hradlo NOR

2.1.2 Pravdivostní tabulka

Pravdivostní tabulka je nejběžnější možnost zápisu logické funkce. Vyjadřuje všechny možné kombinace vstupních proměnných a jím odpovídajících výstupů. Počet sloupců pravdivostní tabulky je dán celkovým počtem proměnných. Počet řádků pravdivostní tabulky je dán vztahem:

$$N=2^n \tag{1}$$

kde N-výsledný počet řádků(kombinací),n-počet vstupních proměnných

Tabulka 7 – Pravdivostní tabulka

	X0	**X1**	**X2**	**y**
I.	0	0	0	0
II.	0	0	1	0
III.	0	1	0	1
IV.	0	1	1	0
V.	1	0	0	1
VI.	1	0	1	0
VII.	1	1	0	X
VIII.	1	1	1	X

V prvním sloupci pravdivostní tabulky vidíme pořadí kombinace. Tento sloupec se běžně vynechává. Sloupce X0,X1 a X2 představují vstupní proměnné, sloupec y představuje výstup. Výstup v hodnotě X je stav, na kterém nezáleží (např. nemůže nastat).

2.1.3 Booleova algebra

Booleova algebra je základní aparát pro práci s logickými výrazy. Je pojmenována podle irského matematika George Boola. Jejím vhodným využitím zjednodušíme logickou funkci tak, že její následná výroba bude jednodušší a levnější, provoz ekonomičtější a spolehlivější.

Zákon vyloučení třetího

$x \vee \bar{x} = 1$ (2)

Příklad: $1 \vee 0 = 1, 0 \vee 1 = 1$

Logický rozpor

$x \cdot \bar{x} = 1$ (3)

Příklad: $1 \cdot 0 = 0, 0 \cdot 1 = 0$

Dvojitá negace

$\bar{\bar{x}} = x$ (4)

Příklad: $\bar{1} = 0 \ldots \bar{\bar{1}} = 1,\ \bar{0} = 1 \ldots \bar{\bar{0}} = 0$

Opakování

$x \vee x = x$ (5)

$x \cdot x = x$ (6)

Příklad: $1 \vee 1 = 1, 1 \cdot 1 = 1$

Komutativní zákony

$x_1 \vee x_2 = x_2 \vee x_1$ (7)

$x_1 \cdot x_2 = x_2 \cdot x_1$ (8)

Asociativní zákony

$x_1 \vee (x_2 \vee x_3) = x_1 \vee x_2 \vee x_3$ (9)

$x_1 \cdot (x_2 \cdot x_3) = x_1 \cdot x_2 \cdot x_3$ (10)

Distributivní zákony

$x_1 \cdot (x_2 \vee x_3) = (x_1 \cdot x_2) \vee (x_1 \cdot x_3)$ (11)

$x_1 \vee (x_2 \cdot x_3) = (x_1 \vee x_2) \cdot (x_1 \vee x_3)$ (12)

Absorpční zákony

$$x_1 \vee (x_1 \cdot x_2) = x_1 \tag{13}$$

$$x_1 \vee (\overline{x_1} \cdot x_2) = x_1 \vee x_2 \tag{14}$$

$$x_1 \cdot (x_1 \vee x_2) = x_1 \tag{15}$$

$$x_1 \cdot (\overline{x_1} \vee x_2) = x_1 \cdot x_2 \tag{16}$$

Neutrálnost 0 a 1

$$0 \vee x = x \tag{17}$$

$$1 \cdot x = x \tag{18}$$

Agresivnost 0 a 1

$$1 \vee x = 1 \tag{19}$$

$$0 \cdot x = 0 \tag{20}$$

De Morganovy zákony

$$\overline{x_1 \vee x_2} = \overline{x_1} \cdot \overline{x_2} \tag{21}$$

$$\overline{x_1 \cdot x_2} = \overline{x_1} \vee \overline{x_2} \tag{22}$$

De Morganovy zákony naleznou své uplatnění především při převádění z Booleovy algebry na NAND nebo NOR logiku.

2.1.4 Disjunktivní a konjunktivní normální forma

Při převodu z pravdivostí tabulky na logickou funkci máme dvě možnosti.

a) Postupujeme po řádcích, které mají hodnotu výstupu 1, vstupní proměnné mezi sebou násobíme. Pokud má vstupní proměnná hodnotu 1 je zastoupena přímo, pokud má hodnotu 0, je zastoupen svou negací. Jednotlivé řádky mezi sebou sčítáme. Tímto postupem získáme Disjunktivní normální formu (DNF).

b) Postupujeme také po řádcích, ovšem pouze po těch, které mají hodnotu výstupu 0. Jednotlivé vstupní proměnné mezi sebou sčítáme. Pokud má proměnná hodnotu 0, je zastoupena přímo, když má hodnotu 1, je zastoupena svojí negací. Jednotlivé řádky mezi sebou násobíme. Tímto postupem získáme Konjunktivní normální formu (KNF).

Příklad:

Tabulka 8–Pravdivostní tabulka – příklad

X0	X1	X2	y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

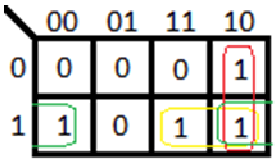
a) $y = \overline{x_0} \cdot x_1 \cdot \overline{x_2} \vee x_0 \cdot \overline{x_1} \cdot \overline{x_2} \vee x_0 \cdot x_1 \cdot \overline{x_2} \vee x_0 \cdot x_1 \cdot x_2$

b) $y = (x_0 \vee x_1 \vee x_2) \cdot (x_0 \vee x_1 \vee \overline{x_2}) \cdot (x_0 \vee \overline{x_1} \vee \overline{x_2}) \cdot (\overline{x_0} \vee x_1 \vee \overline{x_2})$

2.1.5 Karnaughova mapa

Karnaughova mapa je jeden z možných způsobů vyjádření logické funkce. Používá se především k jejich minimalizaci. Mapa má 2^n políček, kde n je počet vstupních proměnných. Každé pole představuje jednu vstupní kombinaci a obsahuje jí příslušející stav výstupu. Každé sousední políčko se liší hodnotou jen jediného argumentu. Teoreticky je možné Karnaughovy mapy použít pro libovolný počet vstupních proměnných ovšem kvůli přehlednosti se většinou tvoří mapy maximálně pro pět proměnných.

Příklad:



Obrázek 5 – Karnaughova mapa

Pro minimalizaci pomocí karnaughovy mapy je třeba uzavřít každou jedničku do smyčky. Uzavírat můžeme vždy 2^n jedniček(1,2,4…). Každá jednička musí být minimálně v jedné smyčce, ale muže být i v několika. Vrchní hrana karnaughovy mapy sousedí s dolní a pravá hrana s levou, tak je možné uzavírat smyčky i tímto způsobem. Za každou smyčku dostaneme část funkce a to tak, že vstupní proměnnou, která svou hodnotu ve smyčce mění, vynecháme,

tu která je zastoupena hodnotou jedna napíšeme přímo, a ta která je zastoupena nulou napíšeme jako negaci.

Příklad : $y = x_1 \cdot \overline{x_2} + x_0 \cdot \overline{x_2} + x_0 \cdot x_1$

Funkci je také možno minimalizovat jen správným použitím Booleovy algebry, dosáhneme tím stejného výsledku. Použití Karnaughovy mapy nám tento postup usnadňuje.

2.2 Realizace logických obvodů

Při realizaci logických máme mnoho možností:

- Relé a stykače
- Hradla a integrované obvody
- Programovatelné logické obvody (PLC)
- Mikroprocesory

Před nástupem polovodičových prvků byly nejčastěji využívanými prvky pro realizaci jednoduchých ovládacích mechanizmů relé a stykače. Pro jejich zakreslení se používaly tzv. liniové diagramy, ze kterých se později vyvinul dodnes používaný jazyk Ladder diagram (LD).

2.2.1 Kombinační logické obvody

U kombinačních logických obvodů záleží výstup pouze na okamžitém stavu vstupu. Reagují tedy okamžitě na změnu vstupních proměnných, pouze se zpožděním způsobeným reakcí součástek, ze kterých je tvořen. Pro realizaci těchto obvodů je nejčastějším postupem sestavení pravdivostní tabulky, z ní například pomocí Karnaughových map je získána minimální funkce, která je realizována pomocí hradel. Výslednou funkci lze upravit do tvaru DNF a použít hradla pouze jednoho typu, čímž se usnadní výsledná realizace obvodu.

2.2.2 Sekvenční logické obvody

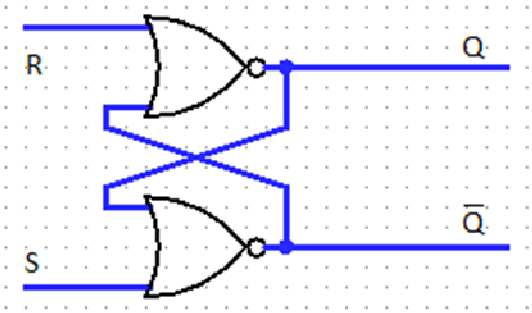
Sekvenční logický obvod vyhodnocuje určitou posloupnost vstupních proměnných a podle té zareaguje výstupy. Hlavním rozdílem oproti kombinačním logickým obvodům je přítomnost paměti, kde je zaznamenán předchozí vývoj systému. Nejjednodušším případem paměti je zpětná vazba. Rozlišujeme dva druhy SLO:

Synchronní: změna výstupních proměnných probíhá v závislosti na taktovacím signálu (hodinový signál)

Asynchronní: změna výstupních proměnných probíhá okamžitě po změně vstupních proměnných.

Klopný obvod R-S

Základní klopný obvod, který je základem i složitějších klopných obvodů. Tento obvod má dva vstupy, R- reset a S- set.



Obrázek 6 – Klopný obvod R-S sestavený z hradel NOR

Tabulka 9 – Stavy klopného obvodu R-S

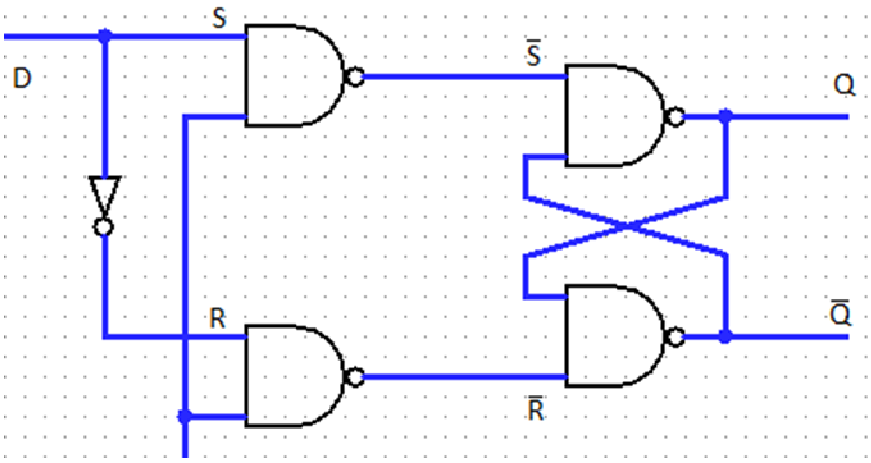
	S	**R**	$\mathbf{Q_n}$	$\overline{\mathbf{Q_n}}$
1.	0	0	Q_{n-1}	$\overline{Q_{n-1}}$
2.	0	1	0	1
3.	1	0	1	0
4.	1	1	0	0

Jak je patrné z tabulky, pokud je na oba vstupy přivedena hodnota logické nuly (řádek 1.), obvod si pamatuje předchozí stav výstupu (Q_{n-1}). Z řádku 2. je patrné že po přivedení logické nuly na vstup S a logické jedničky na vstup R se na výstup Q nastaví hodnota logické nuly. Stejně tak pokud je na vstup S přivedena logická jednička a vstup R je v úrovni logické nuly, je na výstup Q přenesena logická jednička (řádek 3.). Na řádku 4. vidíme zakázaný stav, kdy by na obou výstupech Q i $\overline{Q}$ byla hodnota logické nuly[2].

Klopný obvod D

Složitější klopný obvod, který eliminuje zakázaný stav obvodu R-S. Má dva vstupy, D data, na který jsou přiváděna vstupní data, a E enable, který povoluje funkci obvodu. Pokud je

vstup E v logické jedničce, na výstupu bude stejná hodnota jako na vstupu D, změnou hodnoty E na logickou nulu zůstane na výstupu předchozí hodnota výstupu bez závislosti na vstupu D[3].



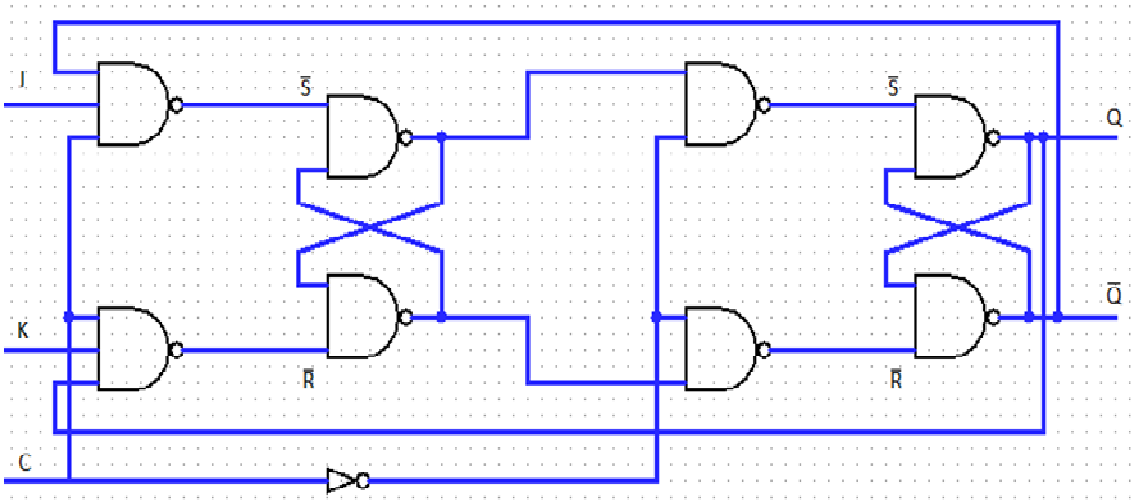
Obrázek 7 – klopný obvod D z hradel NAND

Tabulka10 – Stavy klopného obvodu D

D	E	Q_n	$\overline{Q_n}$
0	1	0	1
1	1	1	0
X	0	Q_{n-1}	$\overline{Q_{n-1}}$

Klopný obvod J-K

Klopný obvod J-K je další rozšíření klopného obvodu R-S, které se především snaží eliminovat jeho zakázaný stav. Tento klopný obvod obsahuje dva datové vstupy označené jako J nastavení a K nulování. Je ovládám hodinovým signálem, který je přiveden na vstup C clock. Obvod je sestaven ze dvou obvodů R-S řízených hranou, první označovaný jako master je ovládán hranou náběžnou, druhý označovaný jako slave reaguje na hranu sestupnou. Funkci klopného obvodu vyjadřuje tabulka 15[4].



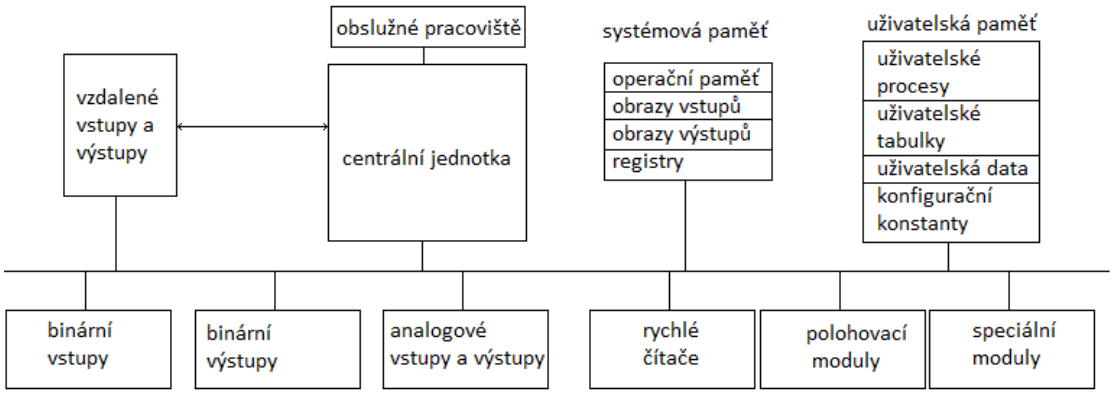
Obrázek 8 – princip zapojení klopného obvodu J-K z hradel NAND

Tabulka 11 – Stavy klopného obvodu J-K

J	K	C	Q_n	$\overline{Q_n}$
0	0	⊓	Q_{n-1}	$\overline{Q_{n-1}}$
0	1	⊓	0	1
1	0	⊓	1	0
X	X	0	Q_{n-1}	$\overline{Q_{n-1}}$
1	1	⊓	$\overline{Q_{n-1}}$	Q_{n-1}

2.3 Programovatelné logické automaty – PLC

Programovatelný logický automat je malý průmyslový počítač používaný pro řízení technologických procesů v reálném čase. Podle předem navrženého programu ovládá technologický a pomocí diagnostických a kontrolních funkcí (watchdog) zabezpečuje, aby bylo dosaženo požadovaného výsledku. Jejich hlavní výhodou je vysoká spolehlivost, zvýšená odolnost proti vlivům prostředí (prach, vlhkost, otřesy…) a výkyvům napájecího napětí.



Obrázek 9 – Blokové schéma vnitřní struktury PLC

Centrální jednotka: obsahuje mikroprocesor, který řídí činnost PLC.

Systémová paměť: paměť, do které jsou ukládány obrazy vstupů a výstupů, slouží jako operační paměť. Většinou se jedná o paměť typu RAM.

Uživatelská paměť: paměť, do které jsou ukládány uživatelská data. Obvykle se také jedná o paměť typu RAM.

Binární vstupy: slouží k přivedení dvouhodnotových signálů, například ze spínače.

Binární vstupy: umožňují dvouhodnotové ovládání připojených zařízení jako je spínání cívky relé. Tyto výstupy bývají realizovány pomocí tranzistorů nebo relé.

Analogové vstupy: slouží k přivedení analogových veličin do PLC. Jejich pomocí je možné k PLC připojit například teplotní čidla (termočlánky).

Analogové výstupy: umožňují ovládat zařízení s analogovým vstupem. Například regulovat otáčky motorů, otevření servomotorů nebo ovládat frekvenční měniče.

Speciální moduly: přídavné moduly PLC, umožňující připojení speciálních periferií.

2.3.1 Rozdělení PLC

Podle složitosti a náročnosti řešené úlohy je volen typ PLC

Logické moduly/smartrelay

nejjednodušší typ zpravidla nahrazující jedno relé, čítač nebo časovač.

MikroPLC

Nejmenší typ PLC hodící se pro nejjednodušší aplikace. Většinou vybaven jedním mikroprocesorem. Obsahuje pevnou sestavou vstupů a výstupů obvykle pouze v binárním provedení. MikroPLC se nedají rozšířit.

Kompaktní PLC

Nabízí pevnou sestavu vstupů a výstupů. Může obsahovat několik mikroprocesorů. Je k němu možné připojit rozšiřující moduly, například další vstupy, ale rozšiřitelnost je omezená.

Modulární PLC

PLC pro náročné aplikace, jednotlivé části jsou na zvláštních modulech, které se vkládají do rámu. Tento typ PLC nabízí nejširší možnosti konfigurace HW PLC, rámy se vyrábějí pro různé množství rozšiřujících modulů a je možné propojit několik rámů navzájem. Na rámech je vždy pevné umístění pro základní moduly, jako je modul zdroje, CPU a paměti. Jednotlivé moduly mohou být umístěny i stovky metrů od sebe.

SoftPLC

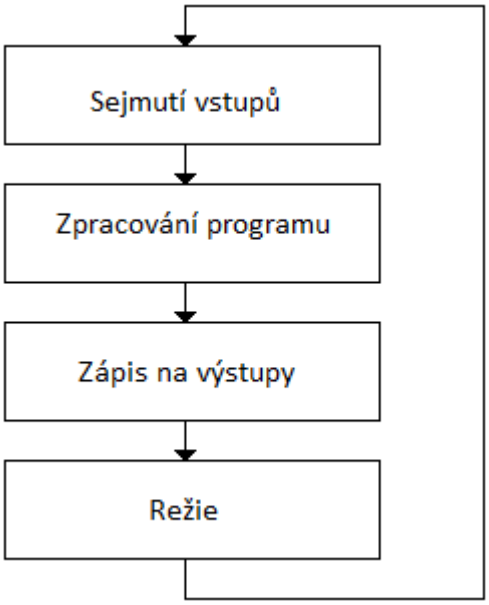
PLC, které je tvořeno PC s realtime operačním systémem a sadou speciálních vstupních a výstupních periferii. Celý systém tak muže využít výhod a funkcí operačního systému nebo programovacích jazyků netypických pro PLC.

PLC pro bezpečné řízení

PLC vhodné pro aplikace, kde je kladen největší důraz na spolehlivost systému, například chemický a jaderný průmysl. Celý systém je většinou zdvojen a vybaven vylepšenou kontrolou.

2.3.2 Průběh programu v PLC

Program v PLC je vykonáván cyklicky. Program nepracuje přímo se vstupy a výstupy, ale pouze s jejich obrazy. Na začátku cyklu jsou sejmuty vstupní veličiny a vytvořen jejich obraz, následně je vykonán program a v závěru jsou zapsány obrazy výstupních veličin na výstupy. Proti zacyklení programu je PLC chráněno pomocí Watchdogu, který hlídá maximální povolený čas jednoho cyklu a při jeho přerušení běh programu zastaví.



Obrázek 10 – Běh programu v PLC

2.3.3 Programovací jazyky PLC

Pro programování PLC se využívají vývojová prostředí. Každý výrobce má vlastní vývojové prostředí, čím dochází k nepřenositelnosti napsaných programů mezi jednotlivými výrobci, někdy dokonce i v rámci jednoho. Pro psaní programů pro PLC si můžeme vybrat ze dvou skupin a několika možností:

- <u>grafické programovací jazyky</u>
  - LD – Jazyk kontaktních schémat (Ladder Diagram)
  - FBD – Jazyk funkčního blokového schématu (FunctionBlock Diagram)
- <u>textové programovací jazyky</u>
  - IL – Jazyk seznamu instrukcí (Instruction List)
  - ST – Jazyk strukturovaného textu

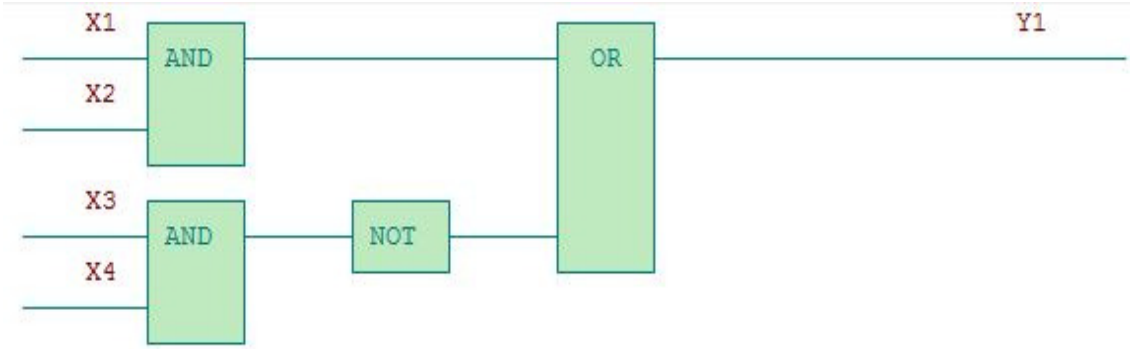
Jazyk kontaktních schémat

Tento jazyk vychází z dříve používaných liniových diagramů využívaných pro zakreslení obvodů s relé a stykači. Využívá spínací a rozpínací kontakty. Lze zde využít i komplikovanějších obvodů jako jsou čítače a časovače. Jazyk je vhodný pro jednodušší úlohy a pro správné pochopení funkce PLC.

Obrázek 11 – jazyk LD

Jazyk funkčního blokového schématu

Jazyk je založen na propojování funkčních bloků a funkcí. Je podobný kreslení logických funkcí, logické funkce a bloky jsou reprezentovány obdélníky. Mezi jednotlivými bloky na rozdíl od jazyka LD nemusí být přenášena hodnota typu boolean ale libovolného datového typu.



Obrázek 12 – jazyk FBD

Jazyk seznamu instrukcí

Je nízkoúrovňový jazyk vychází z assembleru, patří mezi řádkově orientované jazyky. Skládá se z posloupnosti instrukcí, každá instrukce začíná na novém řádku. Lze využít návěští pro provedení skoku v programu.

```
VAR_GLOBAL
 AT %X1.2       : BOOL;
 AT %Y2.0       : BOOL;
END_VAR

PROGRAM Example_IL
 VAR
  tmp1, tmp2   : BOOL;
 END_VAR

Step1:  LD      %X1.2   // load bit from PLC input
    AND     tmp1    (* AND temporary variable *)
    ST      %Y2.0   (* store to PLC output *)
            (* empty instruction *)
Step2:                  (* label *)
    LDN     tmp2
END_PROGRAM
```

Obrázek 13 – jazyk IL[5]

Jazyk strukturovaného textu

Je jazyk definovaný normou IEC 61 131-3. Jedná se o vyšší programovací jazyk s kořeny v jazyku C, je objektově orientován.

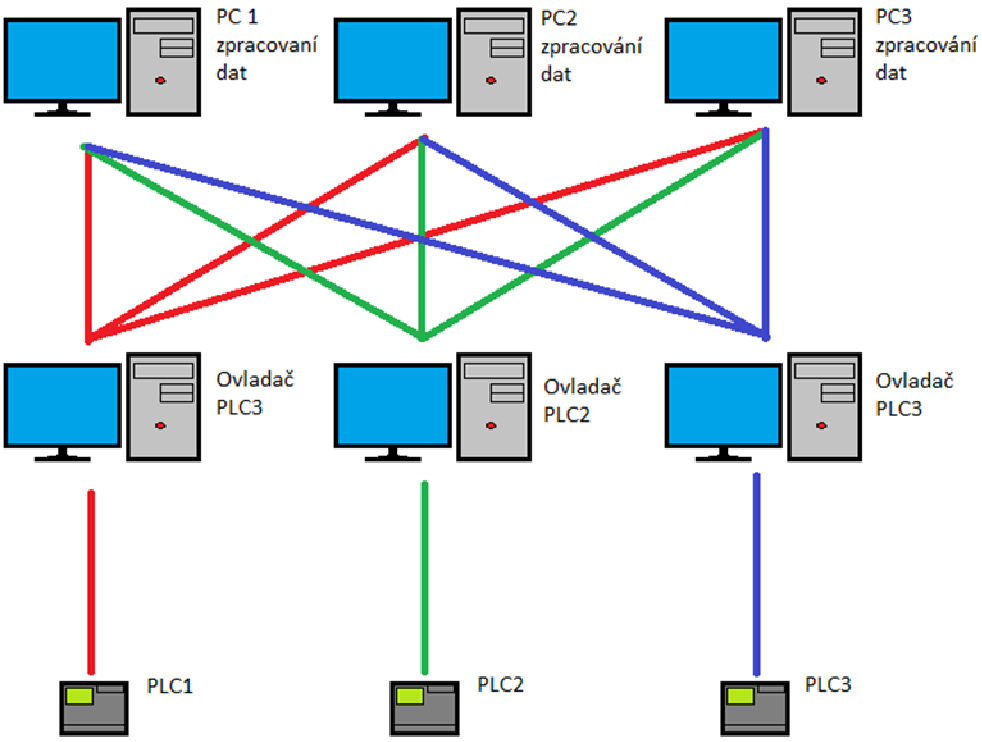
```
PROGRAM PRIKLAD
 VAR                                // lokální proměnné
  A       : INT := 2;
  B       : INT := 4;
  C       : INT := 5;
  D       : INT := 8;
  X,Y     : INT;
  Z       : REAL;
 END_VAR

 X := A + B - C * ABS(D);           // X = -34
 Y := (A + B - C) * ABS(D);         // Y = 8
 Z := INT_TO_REAL( Y);
END_PROGRAM
```

Obrázek 14 – jazyk ST[5]

2.4 OPC

OPC (OLE forProcessControl) je soubor mezinárodních standardů, který definuje způsob předávání informací mezi jednotlivými prvky průmyslové automatizace. Tento standart je spravován sdružením OpcFoundation, na jeho vývoji se ovšem podílí většina dodavatelů automatizačních prostředků a firma Microsoft. Je založen na rozhraní COM/DCOM/OLE firmy Microsoft. Tím je dáno, že OPCfunguje především na operačních systémech firmy Microsoft, (rozhraní DCOM se stalo součástí Windows XP a verzí vyšších, do Windows 95/98 je možné rozhraní doinstalovat) a provoz na Unixových systémech je tím značně zkomplikován.

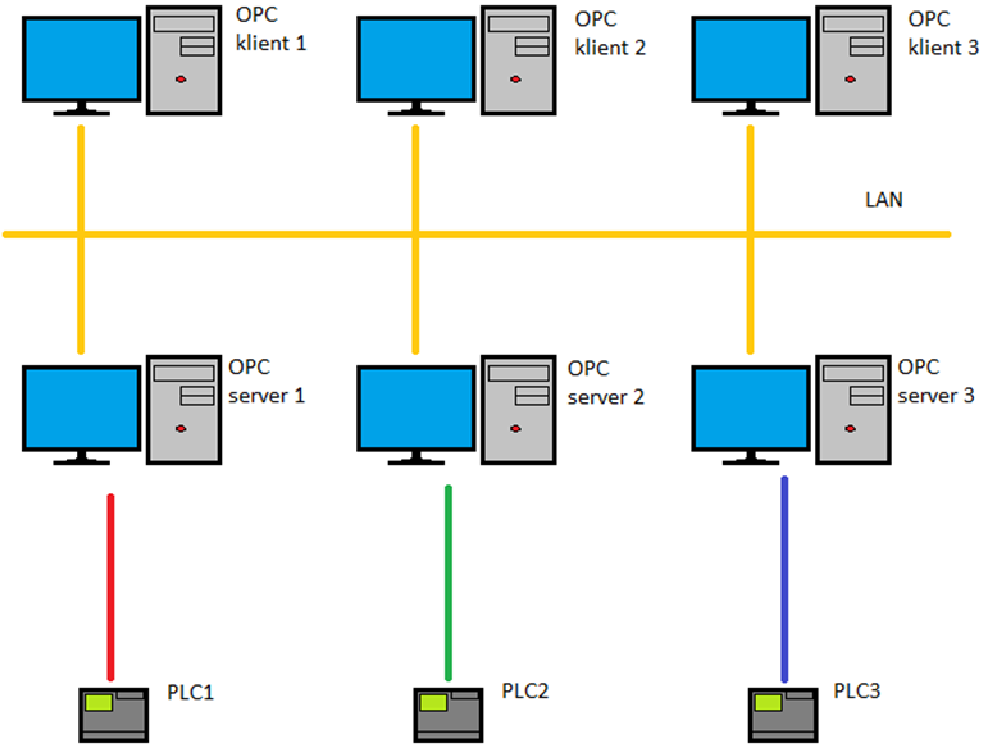
Hlavní výhodou OPC je získání jednotného rozhraní pro komunikaci zařízení různých výrobců.



Obrázek 15 – komunikace bez OPC

Při komunikaci bez použití OPC musí být každé PLC připojeno k počítači kde je nainstalován příslušný ovladač, jak je patrné z obrázku 10. Při použití PLC různých výrobců připojených k jednomu PC může dojít ke kolizím. Přenos dat ke stanicím, na kterých probíhá zpracování,

musí být proveden zvlášť. Pokud je nahrazen nějaký HW například PLC1, musí dojít i instalaci nových ovladačů daného HW, ale zároveň i k úpravě programů, které data dál zpracovávají.



Obrázek 16 – komunikace s využitím OPC

Při použití standartu OPC je PLC připojeno k PC, na kterém je nainstalován příslušný OPC server. Komunikace s PC, na kterých dochází ke zpracování dat (OPC klienti) probíhá pomocí sítě LAN. Pokud dojde výměně PLC stačí nastavit příslušný OPC server a nemusí dojít ke změně řídícího programu[6].

2.4.1 Standard OPC

Mezinárodní dobrovolné sdružení OPC Foundation, které má standard na starosti, sdružuje v současnosti přes 400 členů z řad producentů monitorovacích, řídících, vizualizačních a jiných aplikací v oblasti řízení a sledování technologických procesů.

Standard OPC je tvořen prostřednictvím specifikacíOPC. Specifikace OPC je volně přístupná technická dokumentace stanovující pravidla, podle které se řídí chování a konfigurace rozhraní OPC.

OPC Data Access – v současnosti ve verzi 3.00. Originální specifikace OPC umožňující předávání dat mezi zařízeními v reálném čase. Jedná se o nejčastěji používanou specifikaci.

OPC Alarms&Events – umožňuje předávání alarmů a upozornění na události na požádání. Patří sem operátorské akce, informační zprávy, procesní alarmy a sledovací zprávy.

OPC Batch – podobně jako OPC Data Access, je však určen pro dávkové systémy nikoli spojité.

OPC Data eXchange – tato specifikace umožňuje komunikaci server- server přes sítě typu Fieldbus.

OPC Historical Data Access – umožňuje přístup k již uloženým datům.

OPC Security – Chrání data proti nesprávnému aktualizování a proti neoprávněným změnám parametrů.

OPC XML-DA – poskytuje pravidla a formáty dat pro přenos pomocí jazyka XML.

OPC Complex Data – kombinace OPC Data Access a OPC XML-DA která umožňuje předávání složitějších datových typů, jako jsou binární struktury a XML dokumenty.

OPC Commands – slouží k monitorování, identifikaci a odesílání příkazů prováděných na zařízení.

OPC UnifiedArchitecture – nová sada specifikací která není založena na COM, stává se tím více multiplatformní.

2.4.2 Architektura OPC

Standart OPC využívá schéma klient-server. K jednomu serveru se muže připojit několik klientů[7].

OPC Server

Je ve skutečnosti ovladač, který zajišťuje komunikaci s připojeným PLC a převádí získaná data do formátu OPC. Je třeba vněm nadefinovat vstupní a výstupní veličiny a jejich vazbu na

proměnné v paměti PLC. V závislosti na připojeném HW a typu připojení musí být zvolen správný OPC server.

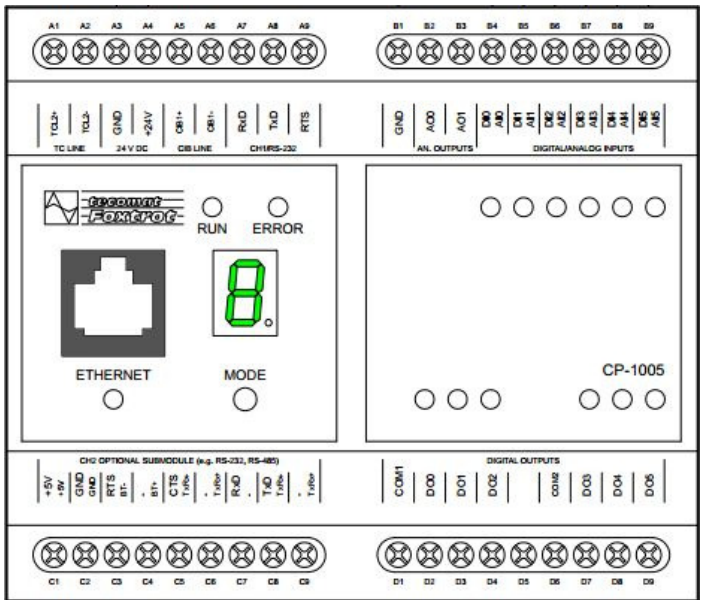
OPC Klient

Je program, který zpracovává přijatá OPC data a provádí s nimi další operace, například vizualizaci nebo samotné řízení procesu. Při přenášení dat se podle standardu OPC přenášejí tři složky hodnota, kvalita a čas. Kvalita muže být dobrá, špatná nebo neznámá.

3 Praktická část

3.1 Tecomat Foxtrot

Tecomat Foxtrot CP-1005 je kompaktní PLC z produkce Kolínské firmy Tecoa.s. Je vybaven 6 víceúčelovými vstupy, použitelnými analogově nebo binárně (24 V). Modul obsahuje dva analogové výstupy 10 V a 6 binárních, reléových. O výpočty se stará CPU řady K, který je určen pro aplikace s vysokými nároky na výkon. Uživatelská data jsou uložena na paměti CMOS RAM. Dále je modul vybaven pamětí Flash pro zálohování programu, slotem na paměťové karty (do 2 GB), obvodem reálného času, dvěma sériovými kanály a systémovým rozhraním TCL2 pro připojení rozšiřovacích modulů. Modul neobsahuje zdroj.



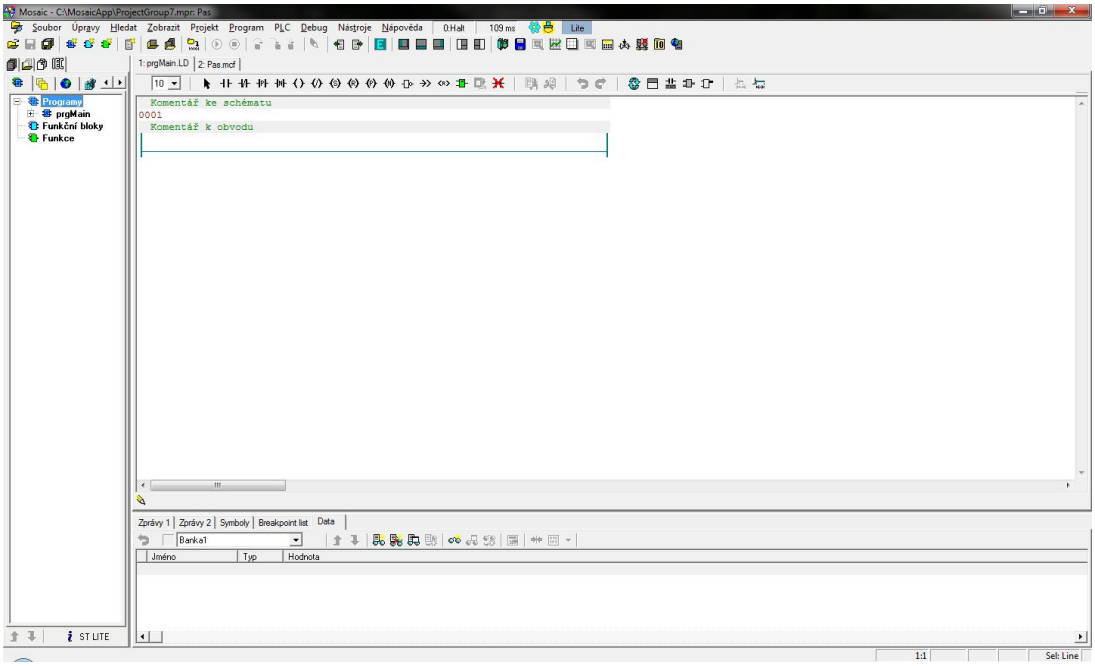
Obrázek 17 – Základní Modul CP – 1005[8]

3.2 Mosaic

Mosaic je vývojové prostředí pro PLC automaty firmy Teco. Program je dodáván ve formě „All in one“, obsahuje tedy všechny v tu chvíli dostupné prostředky. Aktualizace programu vychází několikrát ročně, je ale vždy zajištěna zpětná kompatibilita. Aktuálně je program dostupný ve verzi v2013.1. Pokud není po instalaci přítomný HW klíč, funguje program ve verzi Lite, která umožňuje plné využití programu s několika omezeními.

3.2.1 Tvorba programu

Po spuštění programu je zobrazena informace o verzi programu a přítomnosti HW klíče. Následující okno umožňuje založit novou skupinu projektů nebo otevřít skupinu již existující. Pokud je zvolena nová skupina projektů, je uživatel nejprve vyzván k zadání jména této skupiny projektů. Následně je možné importovat již existující projekt do této skupiny nebo založit projekt nový. Další okno slouží k výběru použitého PLC. Posledním krokem je založení samotného programu, kde je možné zvolit jeho jméno a programovací jazyk, ve kterém bude napsán.



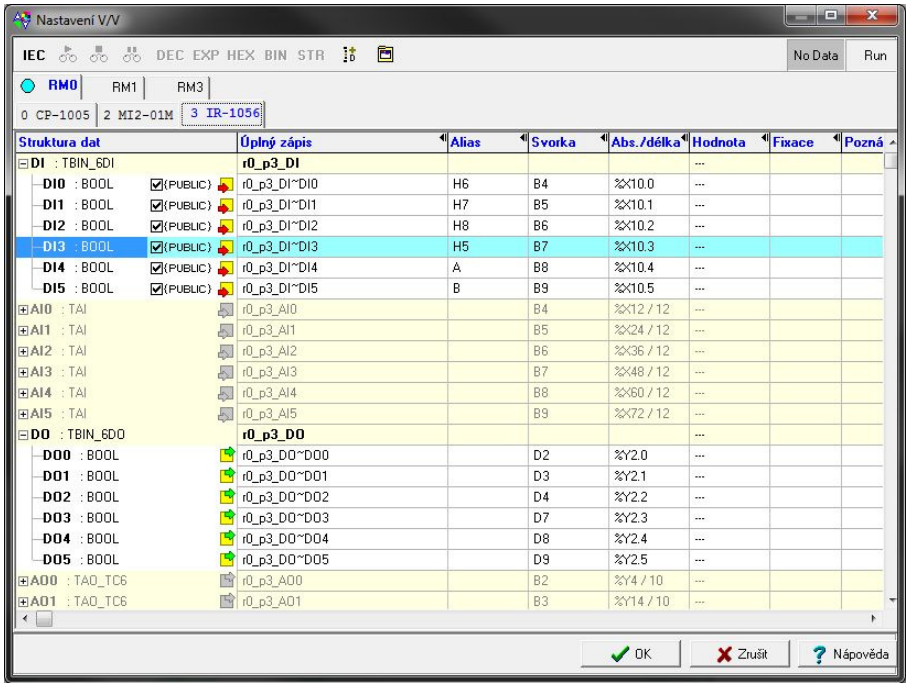
Obrázek 18 – Prázdný program v jazyce LD

Podle jazyka programu, který byl zvolen, se změní hlavní okno a nástrojová lišta. Jednotlivé komponenty jsou u grafických jazyků vkládány tak, že nejprve je na nástrojové liště vybrán požadovaný prvek, který je následně vložen do schématu na místo červeného čtverečku pro sériové řazení, či modrého čtverečku pro řazení paralelní. Po vložení komponenty je třeba zadat její vazbu na proměnnou, pokud tato proměnná ještě neexistuje, je otevřeno okno založení nové proměnné. Aby bylo možné k proměnným přistupovat pomocí OPC, je třeba, aby byly typu „VAR_GLOBAL“ (globální), a aby byly veřejné (zatrženo „{PUBLIC}“). Po dokončení programování je třeba program přeložit. To je možné provést buď příslušným tlačítkem na nástrojové liště, nebo pomocí klávesy F9. Posledním krokem je vyslání a spuštění programu to provedeme pomocí tlačítka „Run“ (Ctrl + F9). Program je takto spuštěn v simulovaném PLC programem Mosaic. Pokud je třeba program vyslat do reálného PLC je

nutno nakonfigurovat jeho připojení. To lze provést v „Manažeru projektů“, který najdeme na nástrojové liště.

3.2.2 Přiřazení aliasů vstupům a výstupům

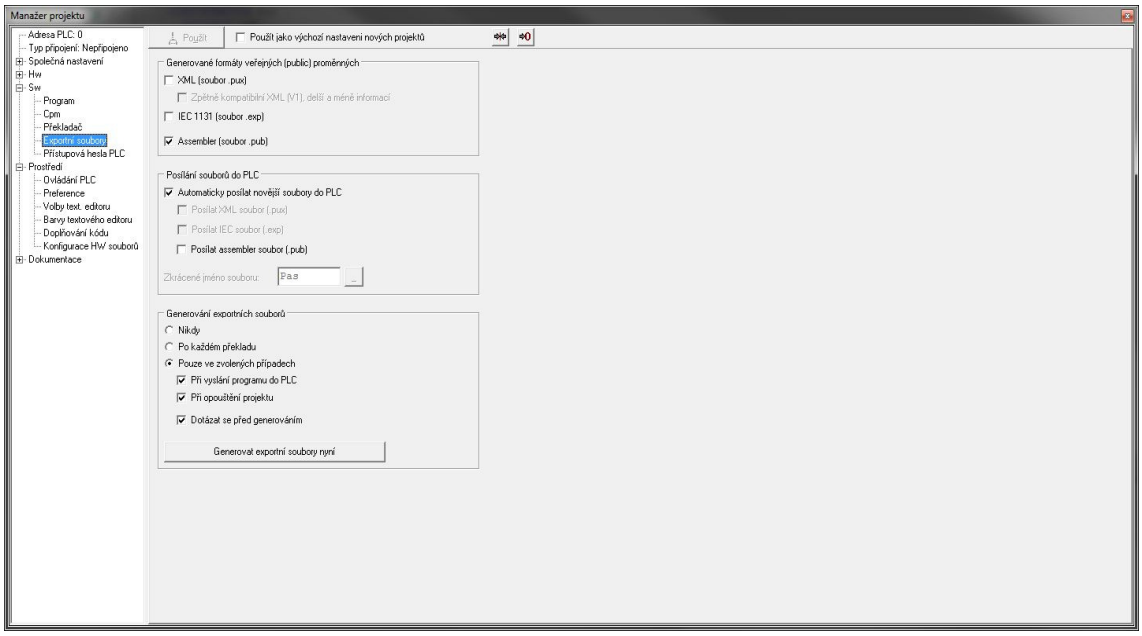
Pro jednodušší práci s vstupy a výstupy PLC automatu program Mosaic umožňuje jejich symbolické pojmenování. Po Stisknutí tlačítka „Nastavení V/V“ je možné do řádku s požadovaným vstupem či výstupem do pole „Alias“ napsat jeho symbolické označení. V programu poté stačí místo celého názvu použít tento Alias.



Obrázek 19 – nastavení vstupů a výstupů

3.2.3 Vytvoření exportního souboru pro OPC server

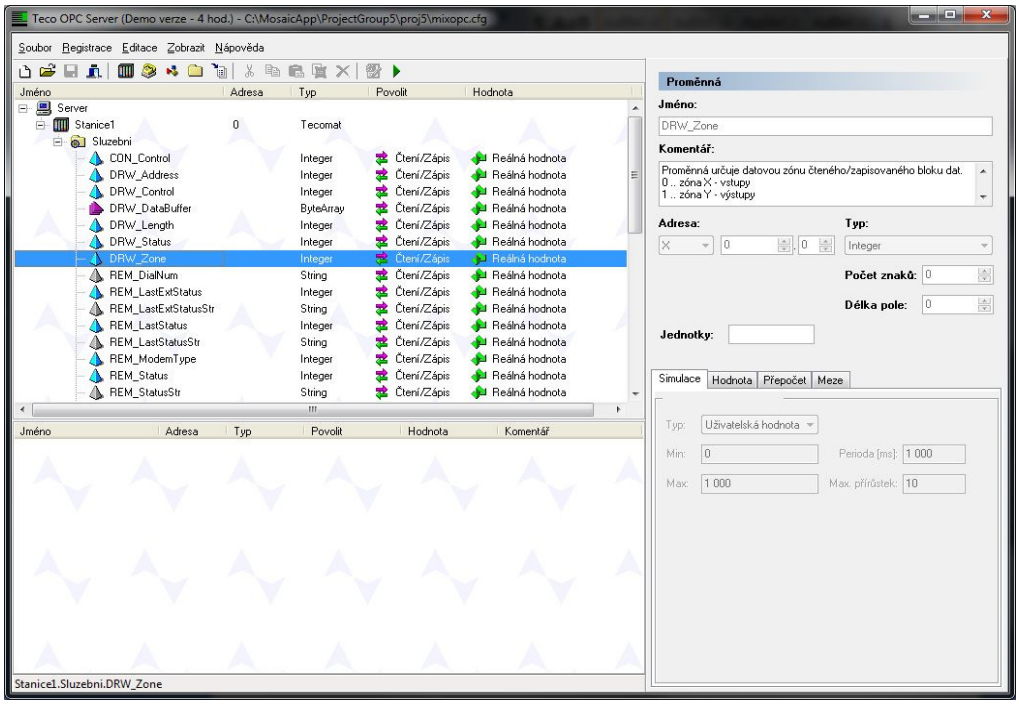
V manažeru projektů v kartě „SW- Exportní soubory“ je třeba nastavit generování exportního souboru zatrhnutím „Assembler(soubor . pub)“ a nastavením případů, kdy se generování provede. Je zde možnost pomocí tlačítka „Generovat exportní soubory nyní“ vygenerovat aktuální soubor.



Obrázek 20 – nastavení generování exportních souborů

3.3 Teco OPC

Teco OPC je OPC server pro řídicí systémy firmy Teco. Využívá specifikace OPC Data access 2.0. Licence Teco OPC serveru je chráněna HW klíčem, bez tohoto klíče je umožněn zkušební běh (trial verze) po dobu 4 hodin. Okno programu je rozděleno na tři podokna. Hlavní, které je vlevo nahoře a zobrazuje celou konfiguraci OPC serveru. Vedlejší, které je vlevo dole a zobrazuje obsah jednotlivých zvolených komponent OPC serveru z okna hlavního a okno konfigurační, které je napravo.



Obrázek 21 – Teco OPC server

Nová konfigurace je založena stisknutím tlačítka „Nový“ na nástrojové liště. Poté je nutno pomocí tlačítka „Nová stanice“ přidat novou stanici. V konfiguračním okně je možné změnit typ jejího připojení a její název. Jednotlivé proměnné, které budou předávány OPC serverem, je možné nakonfigurovat buď jednotlivě pomocí tlačítka „Nová proměnná“ nebo efektivněji pomocí tlačítka „import“ a exportního souboru vytvořeného programem Mosaic. Před spuštěním OPC serveru je ještě třeba v panelu „Registrace“ pomocí tlačítka „Registrovat server“ a „Registrovat konfiguraci“ server přidat do registrů systému. Server je možné pomocí tlačítka „Runtime“ spustit.

3.3.1 Konfigurace DCOM

Jedná se o stěžejní bod konfigurace komunikace s použitím OPC. Server by při konfiguraci měl být vypnutý. Konfigurace se liší podle verze operačního systému a je velice problematická. Aby bylo možné vše správně nastavit, je třeba nainstalovat balík „OPC CoreComponents“, který lze stáhnout ze stránek OpcFoundation. První je třeba povolit použití COM na počítači, následně povolit jeho použití jednotlivým uživatelům. Důležité je také přidat výjimku pro OPC server a komponentu „OPCEnum“, která je součástí „OPC CoreComponents“, do firewallu systému Windows.

Následující postup popisuje nastavení rozhraní na Operačním systému Windows 7 x86 ServicePack 1.

1) Konfigurace OPC serveru
2) Nastavení Firewalu

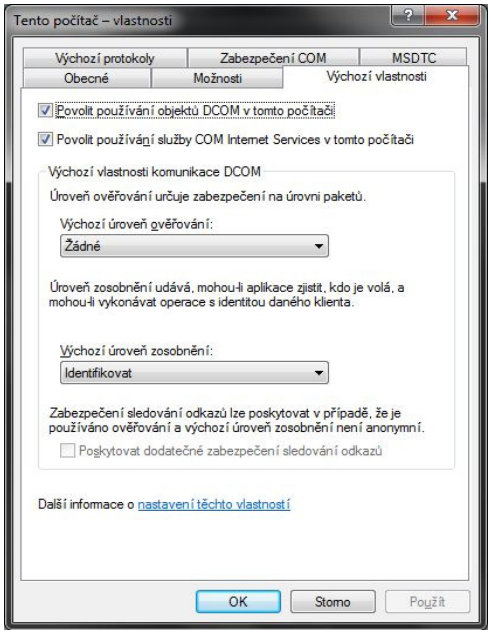
Konfigurace OPC serveru

Nejprve je třeba se ujistit, že je nainstalován balík „OPC CoreComponents“, který může být získán ze stránek www.opcfoundation.org v kategorii „Download->Redistributables“.

Je třeba otevřít „Služby komponent“. To lze provést dvěma způsoby:

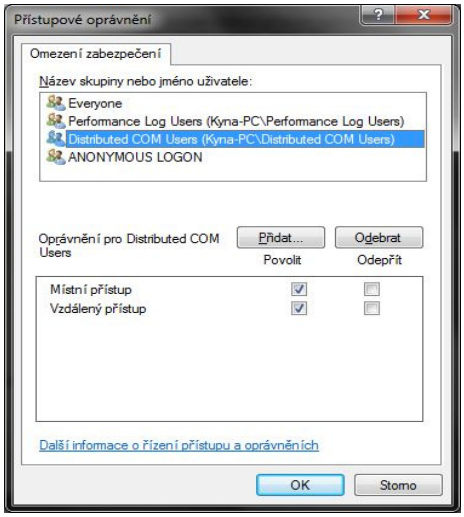
a) Ovládací panely-> Nástroje pro správu->Služby komponent
b) Do pole spustit v nabídce Start bude napsáno „dcomcnfg“

V tomto okně je třeba otevřít „Služba komponent - >Počítače“ pravým tlačítkem je třeba kliknout na „Tento počítač“ a zvolit „Vlastnosti“. Zde je třeba v kartě „Výchozí vlastnosti“ povolit „Povolit používání objektů DCOM v tomto počítači“ a „Povolit používání služby COM Internet Services v tomto počítači“. Dále je nutné nastavit „Výchozí úroveň ověřování“ na „Žádné“ a „Výchozí úroveň zosobnění“ na „Identifikovat“. Vše je patrné na obrázku 1.



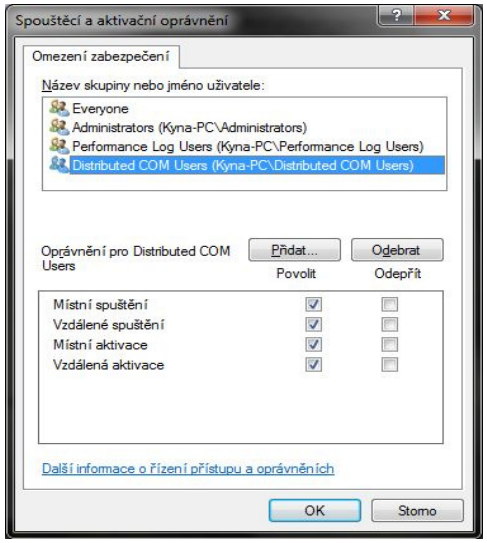
Obrázek 22 – Výchozí vlastnosti

Dále je třeba přejít na kartu „Zabezpečení COM“, zde je nutné nastavit oprávnění osobám a skupinám, které budou mít povolení přistupovat k službám COM tohoto počítače. V „Přístupová oprávnění“ je třeba otevřít „Upravit omezení…“. Zde je třeba pro všechny uživatele, kteří mají mít přístup povolit „Místní přístup“ a „Vzdálený přístup“, jak je vidět na obrázku 2.



Obrázek 23 – přístupová oprávnění

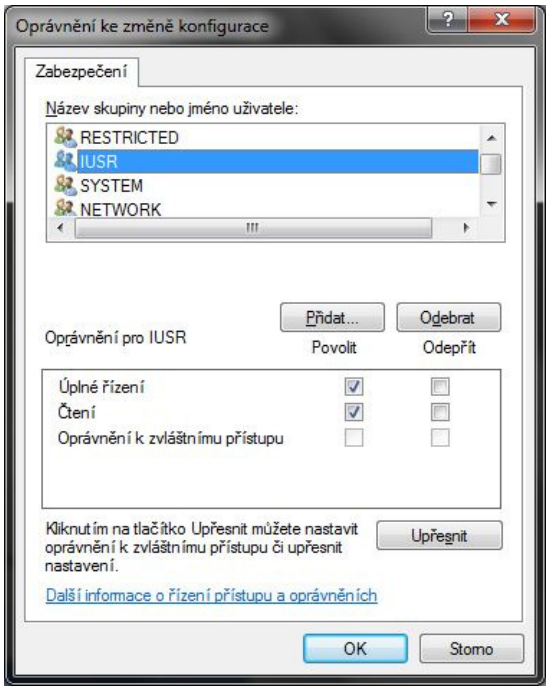
Okno je potvrzeno tlačítkem „OK“. Dále je třeba v části „Spouštěcí a aktivační oprávnění“ upravit „Upravit omezení…“. Zde je třeba pro patřičné uživatele povolit „Místní spouštění“, „Vzdálené spouštění“, „Místní aktivaci“ a „Vzdálená aktivace“ vše je vidět na obrázku 3.



Obrázek 24 – Spouštěcí a aktivační oprávnění

Tím byla dokončena konfigurace toho počítače a karty mohou být pomocí „Ok“ pozavírány.

Je třeba přejít na „Služba komponent->Počítače->Tento počítač->Konfigurace DCOM“ a nalézt „OpcEnum“, pomocí pravého tlačítka otevřít „Vlastnosti“. V kartě „Obecné“ nastavit „Úroveň ověřování“ na „Žádné“, v kartě „Identita“ zvolit „Systémový účet (jen služby)“, dále přejít na kartu „Zabezpečení“, kde je nutné nastavit „Spouštěcí a aktivační oprávnění“ a „Přístupová oprávnění“ na „Použít výchozí“, „Konfigurační oprávnění“ nastavit na „Vlastní nastavení“ a kliknout na „Upravit“. Zde je důležité povolit pro patřičné uživatele „Úplné řízení“ a „Čtení“, jak je vidět na obrázku 4.



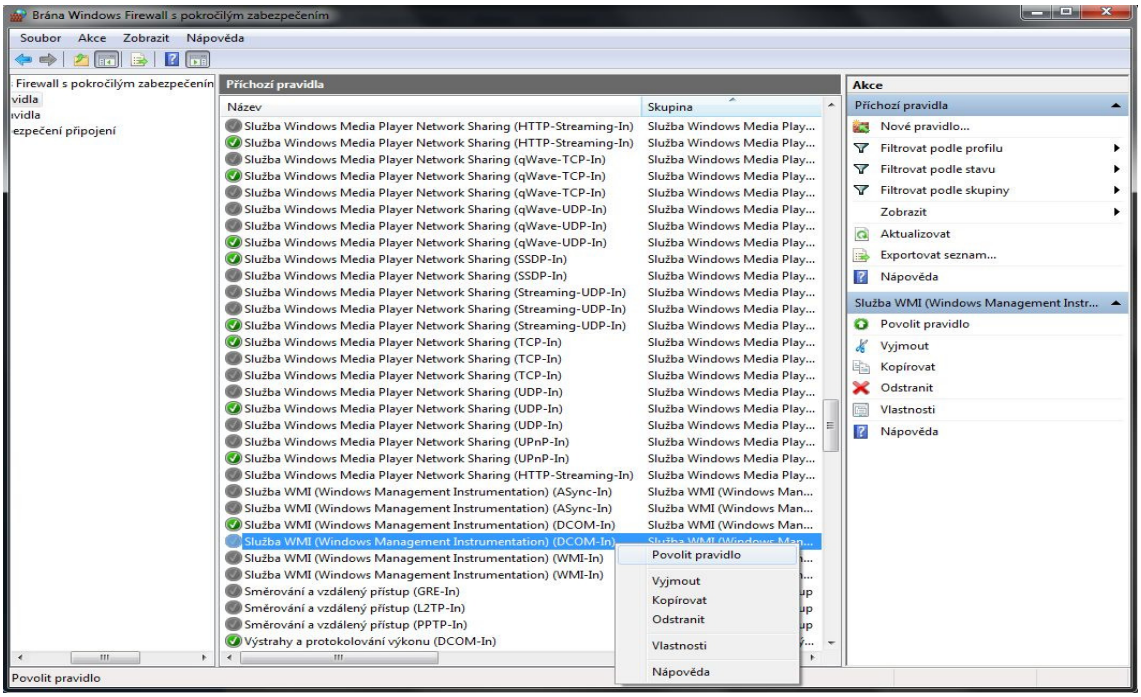
Obrázek 25 – Oprávnění ke změně konfigurace

Okna pomocí „OK“ budou potvrzena.

Dále je nutné nalézt OPC server, který se v případě Teco OPC serveru jmenuje „TECO“ a provést stejné kroky jako při konfiguraci OpcEnum, jen s rozdílem, že v kartě „Identita“ bude zvoleno „Interaktivní uživatel“.

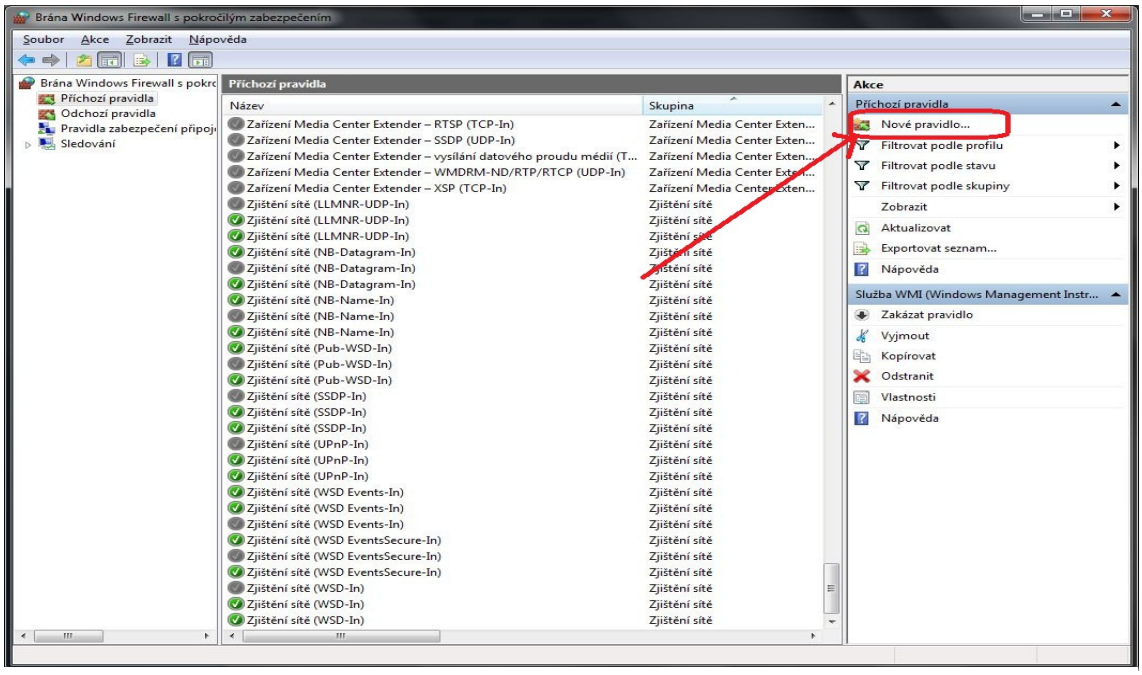
Nastavení Firewalu

Do pole spustit v nabídce Start bude napsáno „wf.msc“. Po potvrzení se otevře nastavení brány firewall systému Windows. Zde je třeba otevřít „Příchozí pravidla“ a nalézt pravidlo, které má název „Služba WMI (Windows Management Instrumentation) (DCOM-In)“, je zde přítomno dvakrát, obě je třeba povolit. To bude uděláno tak, že na pravidlo bude kliknuto pravým tlačítkem a zvoleno „povolit pravidlo“, viz obrázek 5.



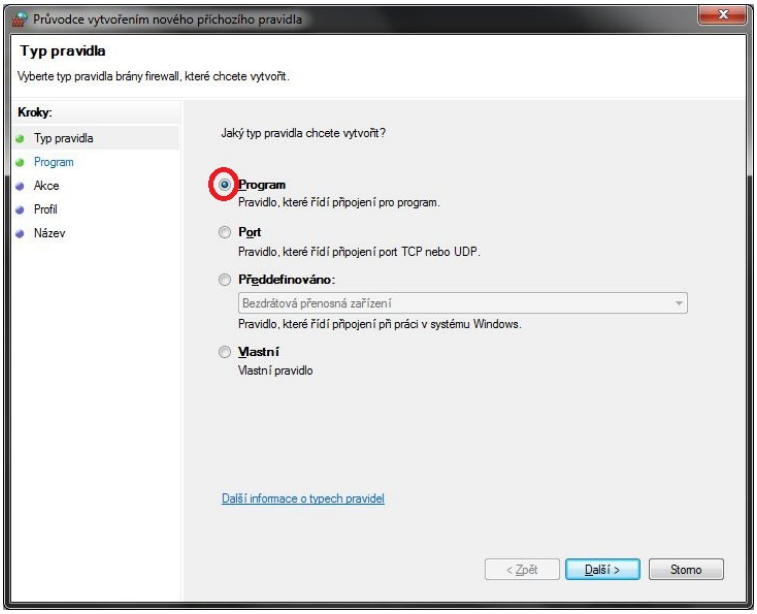
Obrázek 26 – povolení pravidel

Posledním krokem je přidání nového pravidla pro samotný server. To bude provedeno tak, že bude kliknuto na „Nové pravidlo…“ (obrázek 6).



Obrázek 27 – přidání nového pravidla

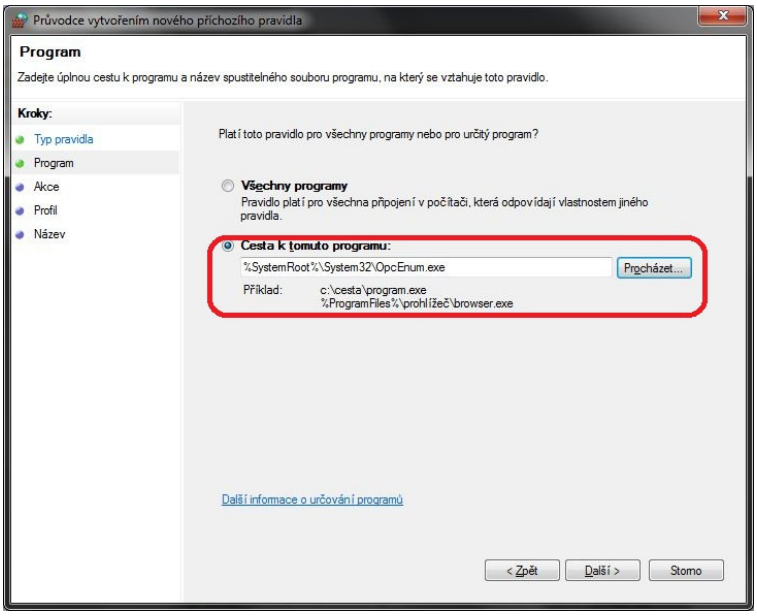
V dalším okně bude zvoleno „program“



Obrázek 28 – Typ nového pravidla

Dále bude kliknuto na „Další >“.

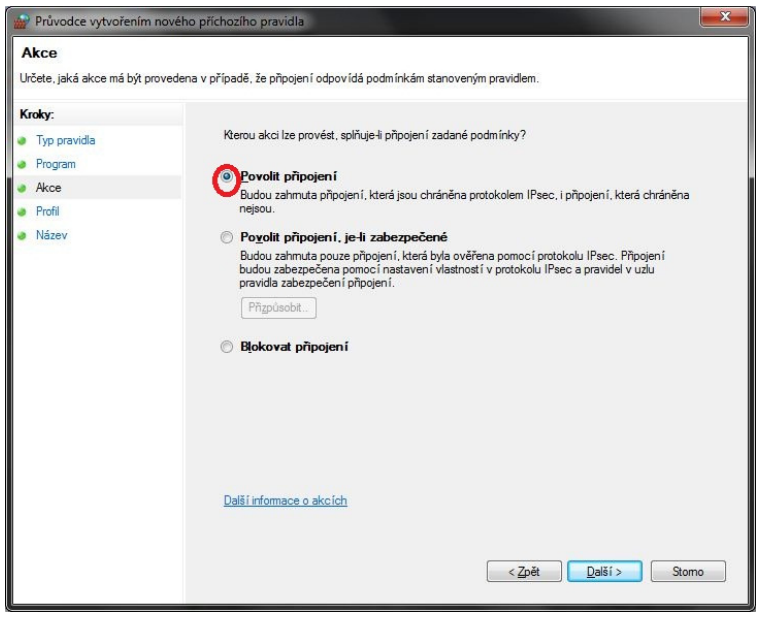
V tomto okně bude vloženo do pole „Cesta k tomuto programu“ cesta, kde se nachází OpcEnum. Cesta je většinou C:\Windows\System32\OpcEnum.exe.



Obrázek 29 – Cesta k programu

Pomocí „Další >“ bude potvrzeno.

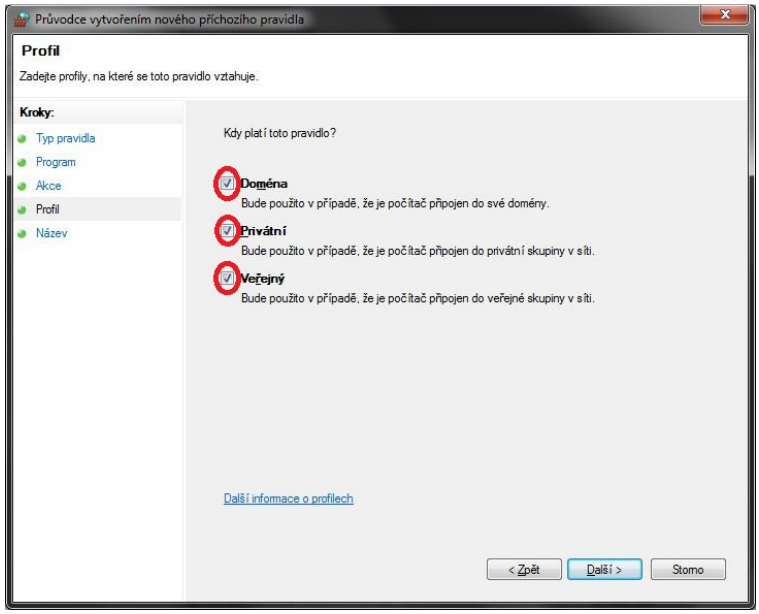
Na dalším okně je třeba zatrhnout „Povolit připojení“.



Obrázek 30 – povolit připojení

Okno bude potvrzeno stiskem „Další >“.

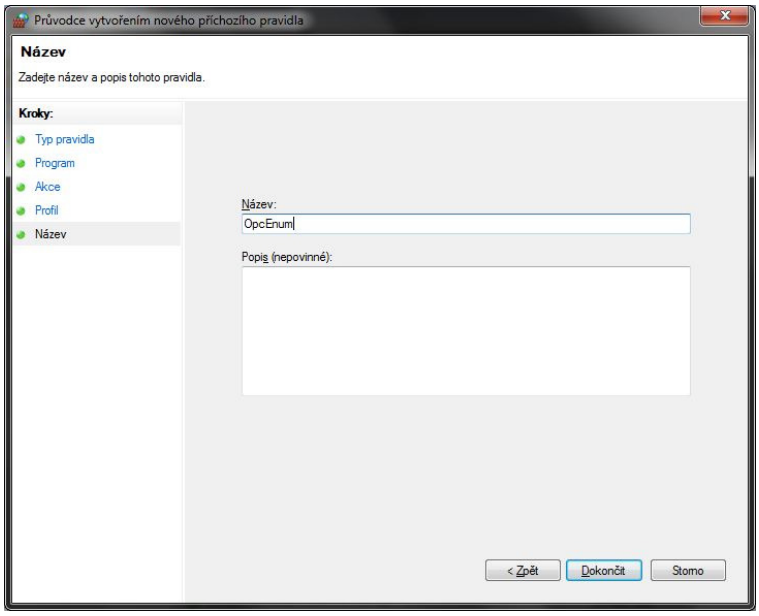
Na dalším okně je třeba zatrhnout „Privátní“, „Doména“ a „Veřejný“.



Obrázek 31 – platnost pravidla

Okno je třeba potvrdit stiskem „Další >“.

Před dokončením pravidla musí být ještě zadán jeho název např.: „OpcEnum“ a kliknutím na „Dokončit“ bude pravidlo přidáno.



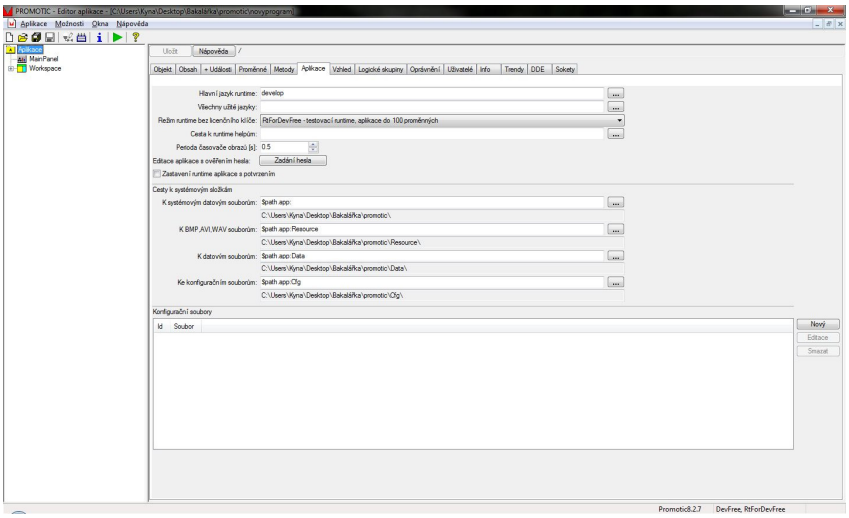
Obrázek 32 – Dokončení pravidla

Poté je třeba zkontrolovat, zda se pravidlo skutečně přidalo.

Stejný postup přidání pravidla bude zopakován i pro TecoOPC server či jiný, který bude používán.

3.4 Promotic

Promotic je SCADA softwarový nástroj pro řízení a zobrazování technologických procesů. Využívá jazyka Basic pro zápis algoritmů. Program je složen z Editoru aplikace, kde je nastavována programová struktura a Editoru obrazů, kde je tvořen vzhled aplikace. Po spuštění aplikace je možné zvolit mezi vytvořením nové aplikace, otevřením aplikace stávající, demo aplikace, kde jsou zobrazeny možnosti programu a příklady v aplikaci, kde jsou formou příkladů s nápovědou popsány problémy s tvorbou skutečných aplikací. Po volbě nové aplikace je zobrazeno okno, kde je třeba zadat název souboru, následně je možné zvolit typ výchozího grafického zobrazení a na konec je třeba zadat název aplikace. Program je následně přeložen.

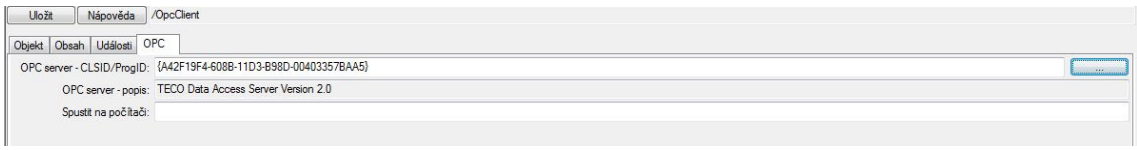


Obrázek 33 – promotic editor aplikace

V levé části okna jsou přidávány jednotlivé objekty aplikace a v pravé části je možné provádět jejich editaci.

PmOpcClient

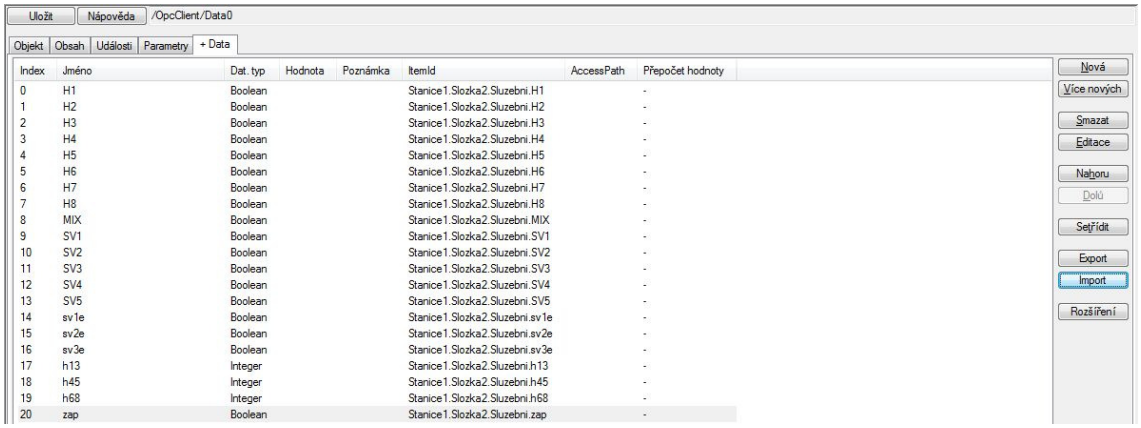
Objekt zajišťuje spojení aplikace Promotic s OPC serverem. Pouze v něm lze založit objekt PmOpcClientData. V tomto objektu je nutné nakonfigurovat připojení k patřičnému OPC serveru. To lze nastavit v kartě „OPC“ v kolonce OPC server po kliknutí na tři tečky výběrem požadovaného dostupného serveru.



Obrázek 34 – PmOpcClient

PmOpcClientData

Objekt, sloužící k získání požadovaných dat z OPC serveru. Zde v kartě „Data“ pomocí tlačítka „Import“ je možno nastavit proměnné potřebné pro obsluhu.

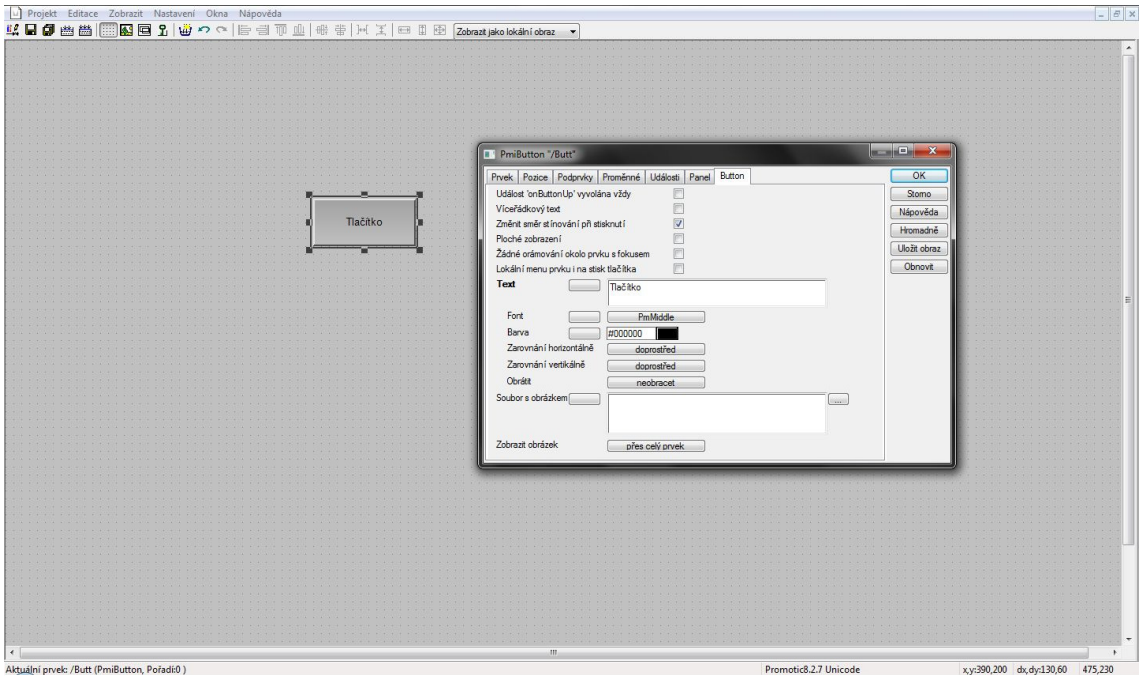


Index	Jméno	Dat. typ	Hodnota	Poznámka	ItemId	AccessPath	Přepočet hodnoty
0	H1	Boolean			Stanice1.Slozka2.Sluzebni.H1		-
1	H2	Boolean			Stanice1.Slozka2.Sluzebni.H2		-
2	H3	Boolean			Stanice1.Slozka2.Sluzebni.H3		-
3	H4	Boolean			Stanice1.Slozka2.Sluzebni.H4		-
4	H5	Boolean			Stanice1.Slozka2.Sluzebni.H5		-
5	H6	Boolean			Stanice1.Slozka2.Sluzebni.H6		-
6	H7	Boolean			Stanice1.Slozka2.Sluzebni.H7		-
7	H8	Boolean			Stanice1.Slozka2.Sluzebni.H8		-
8	MIX	Boolean			Stanice1.Slozka2.Sluzebni.MIX		-
9	SV1	Boolean			Stanice1.Slozka2.Sluzebni.SV1		-
10	SV2	Boolean			Stanice1.Slozka2.Sluzebni.SV2		-
11	SV3	Boolean			Stanice1.Slozka2.Sluzebni.SV3		-
12	SV4	Boolean			Stanice1.Slozka2.Sluzebni.SV4		-
13	SV5	Boolean			Stanice1.Slozka2.Sluzebni.SV5		-
14	sv1e	Boolean			Stanice1.Slozka2.Sluzebni.sv1e		-
15	sv2e	Boolean			Stanice1.Slozka2.Sluzebni.sv2e		-
16	sv3e	Boolean			Stanice1.Slozka2.Sluzebni.sv3e		-
17	h13	Integer			Stanice1.Slozka2.Sluzebni.h13		-
18	h45	Integer			Stanice1.Slozka2.Sluzebni.h45		-
19	h68	Integer			Stanice1.Slozka2.Sluzebni.h68		-
20	zap	Boolean			Stanice1.Slozka2.Sluzebni.zap		-

Obrázek 35 – PmOpcClientData

PmPanel

Objekt představující jedno okno vizualizace. Jeho obsah se vytváří pomocí Editoru obrazů, ten je spuštěn po stisku tlačítka „Editace grafického obsahu“ v editační části okna tohoto objektu. Jednotlivé ovládací a vizualizační prvky se vkládají pomocí dvojkliku na požadovanou pozici a výběrem požadovaného prvku. Každý prvek má vlastní nastavení, kde se nastavují parametry a vazby na proměnné.

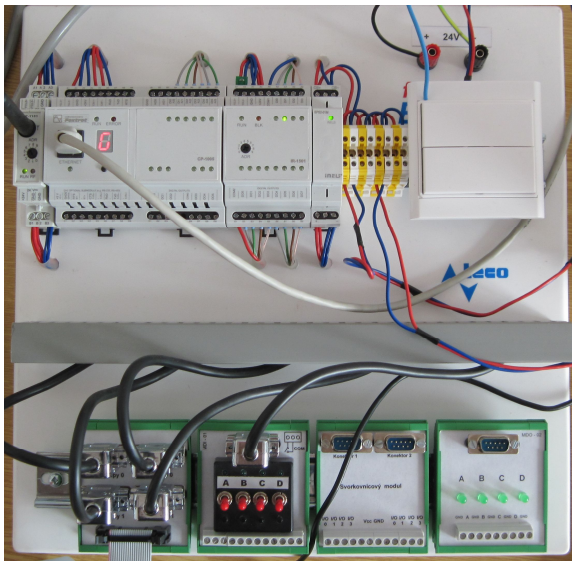


Obrázek 36 – graficky editor a nastavení vlastností tlačítka

3.5 Laboratorní úloha – Mísicí jednotka OPC server

3.5.1 EDUtec

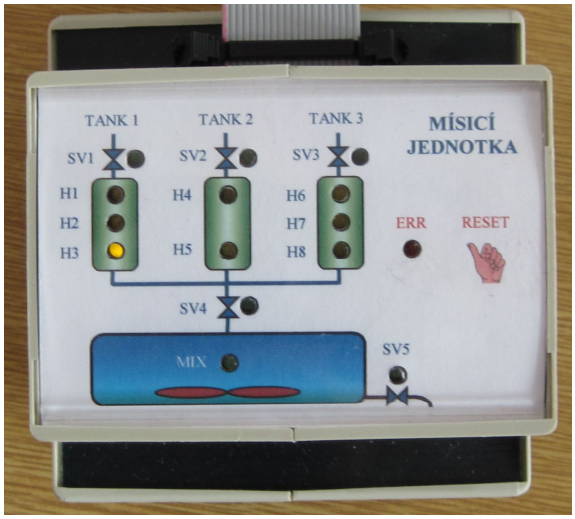
Pro úlohu je využit výukový systém EDUtec Foxtrot vytvořený Ing. Luďkem Kohoutem. Systém obsahuje PLC automat Tecomat Foxtrot CP-1005, rozšiřující modul IR- 1501, který navyšuje množství vstupů a výstupů základního modulu, oddělovacím modulem InelsBPS2-01M sloužícím k impedančnímu oddělení sběrnice CIB a komunikačním převodníkem RF-1131 pro komunikaci s bezdrátovými moduly. Systém je dále vybaven výměnnými moduly s tlačítky a modulem pro připojení modelů EDU – mod.



Obrázek 37 – EDUtec

3.5.2 EDU-mod Mísicí jednotka

Jedná se o aktivní modul simulující tři nádoby a mísicí zařízení. Jednotka je řízena 5 výstupy, které představují ventily, a jedním výstupem, který představuje motor mísicího zařízení. Jednotka simuluje informace z 8 čidel představujících hladinoměry.



Obrázek 38 – EDU-mod Mísicí jednotka

Hladinoměry H3, H5 a H8 představují cca 10 l kapaliny, H2 a H7 polovinu nádrže a hladinoměry H1, H4 a H6 představují plnou nádrž (84 l). Průtok ventily SV1, SV2 a SV3 je 6 l/s. Mísicí nádoba má objem 253 l. Ventily SV4 a SV5 mají průtok 18 l/s. Modul je dále

opatřen tlačítkem reset, které uvede jednotku do inicializačního stavu (všechny nádoby prázdné) a diodou ERR představující chybové hlášení přetečení některé nádoby[9].

Tabulka 12 – Přiřazení vstupů a výstupů modulu k PLC

Modul	**PLC**
H1	r1_p0_DI.DI3
H2	r1_p0_DI.DI2
H3	r1_p0_DI.DI1
H4	r1_p0_DI.DI0
H5	r0_p3_DI.DI3
H6	r0_p3_DI.DI0
H7	r0_p3_DI.DI1
H8	r0_p3_DI.DI2
SV1	r1_p0_DO.DO0
SV2	r1_p0_DO.DO1
SV3	r1_p0_DO.DO2
SV4	r1_p0_DO.DO5
SV5	r1_p0_DO.DO4
MIX	r1_p0_DO.DO6

3.5.3 Laboratorní úloha

Zadání

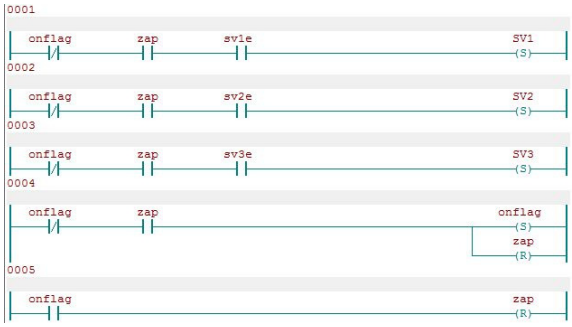
Vytvořte systém, jenž umožní napustit danou nádrž či nádrže po zvolenou úroveň, následně dojde automaticky k přetečení do mísicí nádrže, zamíchání a vypuštění ze systému. Použijte jazyk LD. Celý systém bude pomocí OPC řízen z programu Promotic!.

Řešení

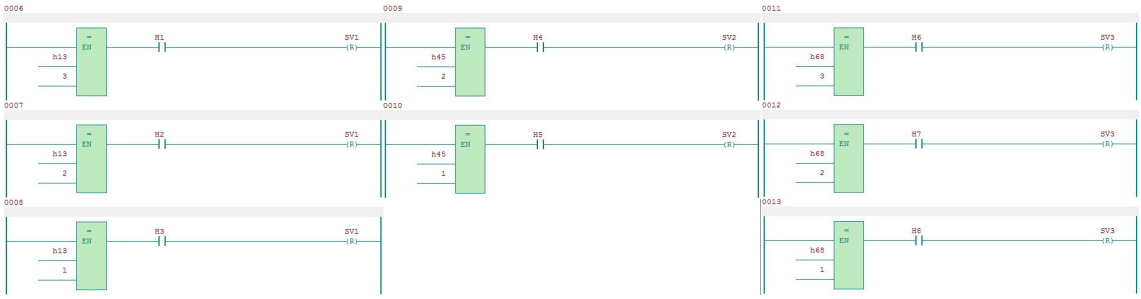
1. Vytvoření programu pro PLC
2. Konfigurace TecoOPC serveru
3. Vytvoření vizualizace v programu Promotic!

Vytvoření programu pro PLC

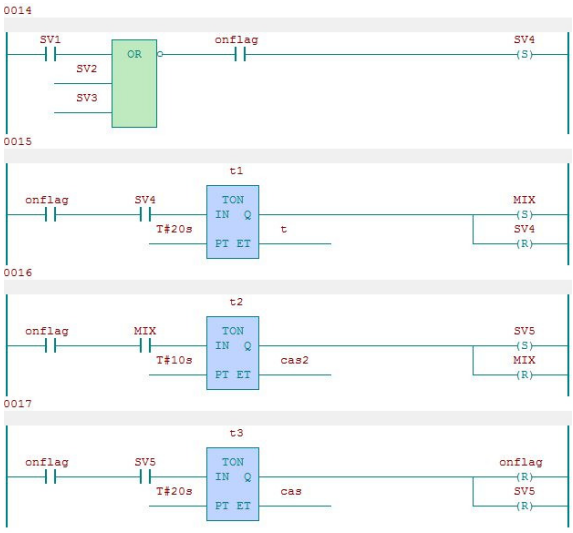
K umožnění napouštění libovolné nádrže je třeba vytvořit kontakty (sv1e,sv2e a sv3e), které budou řízeny z OPC klienta. Výška hladiny bude předávána pomocí číselné hodnoty 1 až 3 pro 1. a 3. nádrž (proměnné h13 a h68) a 1-2 pro nádrž druhou (proměnná h45). Pro zapnutí systému bude využit spínací kontakt řízený z OPC (proměnná zap). Pro znemožnění opětovného spuštění již spuštěného systému bude vytvořena systémová proměnná „onflag“. Vstupy a výstupy modelu mísicí jednotky budou nastaveny podle tabulky 16. K napouštění a vypouštění mísicí nádoby byl stanoven čas 20 s a samotný proces mísení trvá 10 s.



Obrázek 39 – volba nádrží a omezení násobného spuštění



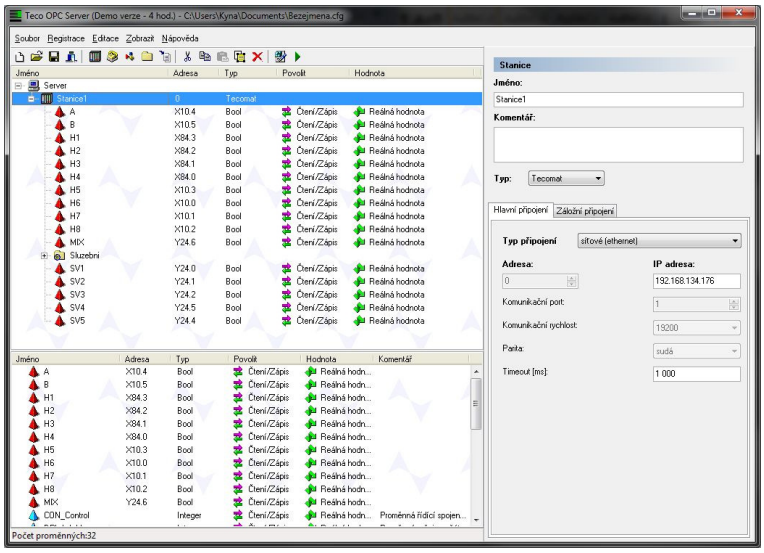
Obrázek 40 – plnění nádrží po zvolenou hladinu



Obrázek 41 – ovládání mísicí nádoby

Konfigurace TecoOPC serveru

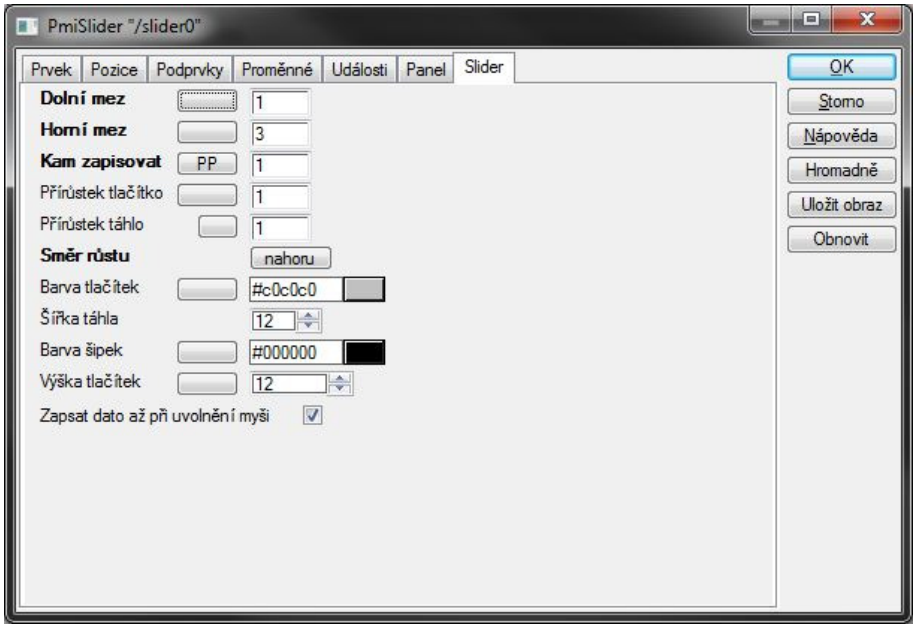
Server bude nakonfigurován pomocí vyexportovaného souboru z programu Mosaic.



Obrázek 42 – Konfigurace TecoOPC

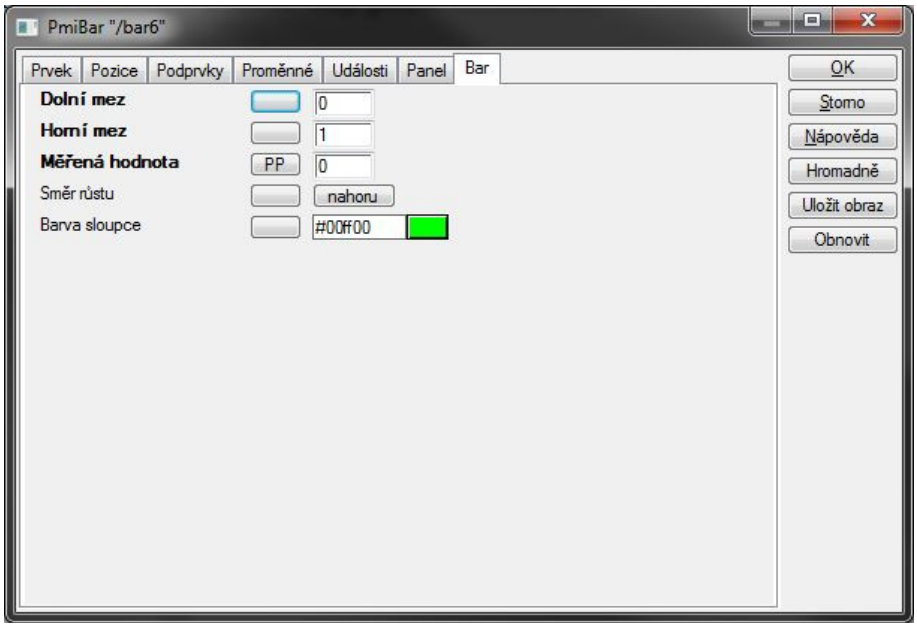
Vytvoření vizualizace v programu Promotic!

Nejprve je třeba nakonfigurovat připojení k OPC serveru. To je provedeno pomocí objektu „PmOpcClient“ (obrázek 34), následně získat potřebné proměnné pomocí „PmOpcClientData“ (obrázek 35). Vizualizace je vytvořena vložením objektu „PmPanel“, kde jsou vkládány jednotlivé komponenty. Pro nastavení požadované výšky hladiny je použit „PmiSlider“ s vazbou na příslušnou proměnou (h13,h45 nebo h68). U něj je třeba nastavit přírůstek táhlem a tlačítkem, a také horní a dolní mez.



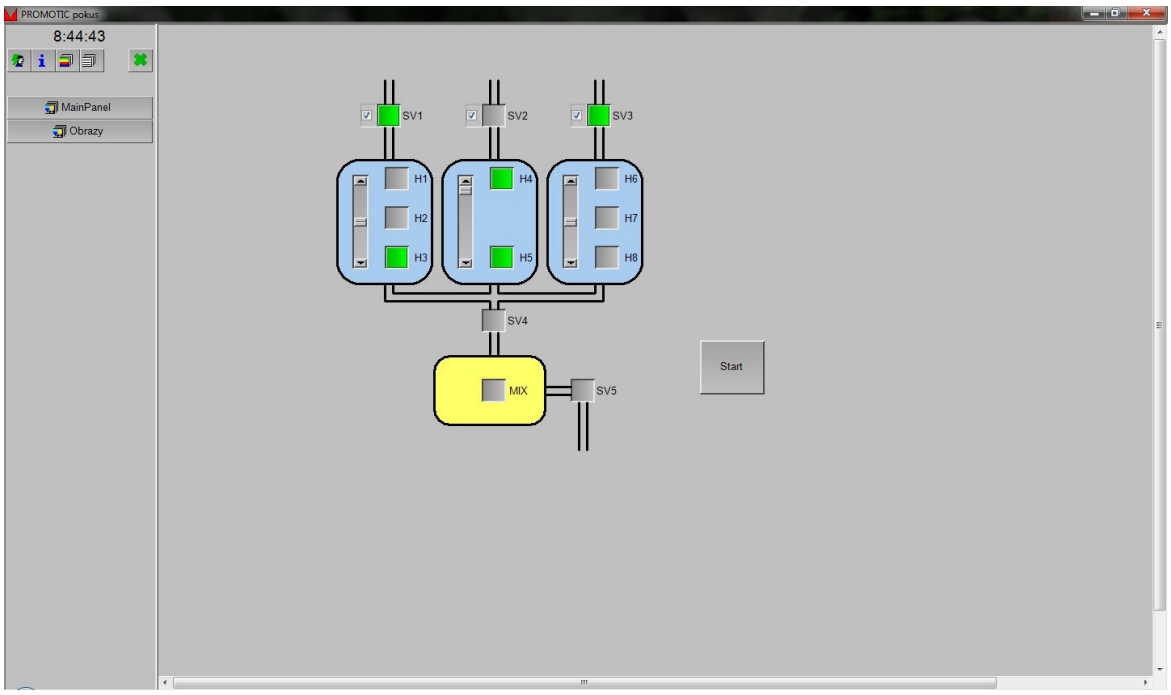
Obrázek 43 – PmiSlider

Pro signalizaci dosažení hladin H1-8, otevření ventilu SV1-5 a spuštění mísicí jednotky je využit „PmiBar“ s vazbou na příslušnou proměnnou a nastavený podle obrázku 44.



Obrázek 44 - PmiBar

Pro zvolení nádrže, která bude napouštěna, jsou použity „PmiWCheck“ s vazbou na proměnné sv1e, sv2e a sv3e. Na spuštění celého procesu je použito tlačítko „PmiButtonTwo“, kterému je přidána vazba na proměnnou „zap“. Výsledná vizualizace je zobrazena na obrázku 45.



Obrázek 45 – Výsledná vizualizace

4 Závěr

Cílem bakalářské práce bylo sestavit úlohu demonstrující funkci standartu OPC za využití výukového systému EDUtec obsahujícího logický automat Tecomat Foxtrot CP-1005. Prvním krokem ke splnění tohoto cíle byla instalace programu Mosaic, kterým se PLC programuje. Program je v češtině a je vybaven podrobnou nápovědou. Dalším krokem byla instalace TecoOPC serveru. Posledním krokem byla instalace aplikace Promotic a vytvoření vizualizace. Výsledná komunikace mezi OPC serverem a programem Promotic se dlouhou dobu nedařila, což bylo způsobeno špatným nastavením rozhraní DCOM systému Windows 7. Tento bod se ukázal jako stěžejní bod celé konfigurace. Po konzultaci s technickou podporou programu Promotic vyšlo najevo, že se způsob nastavení DCOMna jednotlivých verzích operačního systému liší. Nakonec bylo z kombinace několika návodů sestaveno použitelné řešení, ve kterém je nutné nakonfigurovat rozhraní COM a také Firewall systému Windows. Celá komunikace pomocí standartu OPC byla realizována na jednom PC. Vytvořená úloha bude sloužit studentům předmětu Průmyslové řídicí systémy.

Výsledné programy pro laboratorní úlohu a návod na nastavení rozhraní DCOM jsou uloženy na přiloženém CD.

Literatura

[1] *Logické řízení* [online]. 65 s. [cit. 2012-05-08]. Dostupné z: http://autnt.fme.vutbr.cz/lab/a4-603/opory/elr.pdf

[2] Klopný obvod R-S. *Mikrokontroléry PIC: Web o číslicové technice a mikrokontrolérech PIC* [online]. c2012 [cit. 2013-05-2]. Dostupné z: http://mikrokontrolery-pic.cz/zaciname/cislicova-technika/sekvencni-logicke-obvody/klopny-obvod-rs/

[3] Klopný obvod D řízený úrovní. *Mikrokontroléry PIC: Web o číslicové technice a mikrokontrolérech PIC* [online]. c2012 [cit. 2013-05-2]. Dostupné z: http://mikrokontrolery-pic.cz/zaciname/cislicova-technika/sekvencni-logicke-obvody/klopny-obvod-d-rizeny-urovni/

[4] Klopný obvod J-K. *Mikrokontroléry PIC: Web o číslicové technice a mikrokontrolérech PIC* [online]. c2012 [cit. 2013-05-2]. Dostupné z: http://mikrokontrolery-pic.cz/zaciname/cislicova-technika/sekvencni-logicke-obvody/klopny-obvod-jk/

[5] TECO A.S. Nápověda k programu Mosaic: Programování podle IEC 61 131-3. 2011.

[6] Co je OPC ? OPC Server ? OPC Klient ?. FOXON S.R.O. FOXON [online]. 2007 [cit. 2013-05-01]. Dostupné z: http://www.foxon.cz/index.php?main_page=faq_info&fcPath=29&faqs_id=91

[7] Whatis OPC?. *OPC Foundation* [online]. 2013 [cit. 2013-04-10]. Dostupné z: http://www.opcfoundation.org/Default.aspx/01_about/01_whatis.asp?MID=AboutOPC

[8] TECOMAT FOXTROT: ZÁKLADNÍ DOKUMENTACE MODULU. *Tecomat, PLC formachine, process, technology, transport and buildingautomation* [online]. 2. vydání. 2009 [cit. 2013-04-20]. Dostupné z: http://tecomat.com/wpimages/other/DOCS/cze/TXV11005_00_Foxtrot_CP-1005_cz_en.pdf

[9] KOHOUT, Luděk. *Modely EDU-mod*[online]. 2008 [cit. 2012-03-21]. Dostupne z: http://www.edumat.cz/produkty.php?produkt=mixer