

Příprava laboratorních úloh do předmětu Programovatelné automaty

Laboratory Assignment Preparation for the subject
Programmable Controllers

Peter Januška

Bakalářská práce
2010



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Peter JANUŠKA**
Osobní číslo: **A07533**
Studijní program: **B 3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**

Téma práce: **Příprava laboratorních úloh do předmětu
Programovatelné automaty**

Zásady pro vypracování:

1. Vypracujte literární rešerši na téma SCADA/HMI systémy a způsoby komunikace s technologickým procesem.
2. Nainstalujte na všechny počítače v laboratoři 54/303 systém Control Web 6 a odladte bezproblémovou komunikaci s PLC v laboratoři.
3. Vytvořte ukázkové programy znázorňující studentům filozofii a způsob tvorby projektu v prostředí Control Web 6.
4. Realizujte vzdálené řízení vybraných modelů s různými způsoby komunikace model – řídicí počítač.
5. Zpracujte manuál pro studenty, v kterém naleznou nejdůležitější nastavení a další informace k systému Control Web 6.

Rozsah bakalářské práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. **Martinásková, M., Šmejkal, L.: Řízení programovatelnými automaty, Vydavatelství ČVUT, Praha, 1998.**
2. **Martinásková, M., Šmejkal, L.: Řízení programovatelnými automaty II, Vydavatelství ČVUT, Praha, 2000.**
3. **Šmejkal, L., Martinásková, M.: PLC a automatizace, Nakladatelství BEN – technická literatura, Praha, 1999.**
4. **Firemní literatura k programovatelnému automatu Teco TC600.**
5. **Firemní literatura k produktům firmy Saia Burgess.**
6. **Bílý, R., Cagaš, P., Cagaš, R., Hladůvka, D., Kolařík, M., Sobotík, J., Zálešák, M., Zgarba, Z.: Control Web 2000, Computer Press, Praha, 1999.**

Vedoucí bakalářské práce:

Ing. Tomáš Sysala, Ph.D.

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce:

5. března 2010

Termín odevzdání bakalářské práce:

1. června 2010

Ve Zlíně dne 5. března 2010


prof. Ing. Vladimír Vašek, CSc.
děkan




doc. Ing. Ivan Zelinka, Ph.D.
ředitel ústavu

ABSTRAKT

Teoretická časť práce objasňuje čitateľovi pojmy týkajúce sa SCADA systémov. Ponúka pohľad na históriu a vývoj SCADA systémov, rôzne spôsoby architektúr a možnosti v použití odlišných komunikačných štandardov a protokolov. Pojednáva o HMI a popisuje programové vybavenie tvorby aplikácií. Praktická časť sa zaoberá prácou v Control Web 6. Obsahuje krátku užívateľskú príručku k prostrediu a niekoľko ukázkových aplikácií so zameraním na komunikáciu a tvorbu distribuovaných aplikácií, s dôrazom na optimálny návrh z pohľadu zaťaženia procesora a komunikačného systému.

Kľúčové slová: SCADA, komunikačný štandard, komunikačný protokol, HMI, Control Web 6, komunikácia, vzdialená komunikácia, tvorba aplikácií

ABSTRACT

The theoretical component of this work will explain to the reader concepts pertaining to SCADA systems. It offers an overview of the history and development of SCADA systems, various processes of construction, and possibilities in using different communication standards and protocols. It discusses HMI and describes various software used in the creation of applications. The practical component of this work concerns work in Control Web 6. It contains a short guide for the use of the environment and a few demonstrative applications, with a focus on communication and the creation of distributed applications, and emphasizing the best design in regards to strain on the processor and communication system.

Keywords: SCADA, communication standard, communication protocol, HMI, Control Web 6, communication, remote communication, creation of applications

PodĎakovanie patrí pánovi Ing. Tomášovi Sysalovi Ph.D. za možnosť pracovať pod jeho vedením, za pomoc pri tvorbe obsahu a za užitočné vecné poznámky a poskytnuté materiály.

Ďakujem pánovi Ing. Pavlovi Navrátilovi Ph.D. za zapožičanie príslušenstva nevyhnutného k nadviazaniu komunikácie s PLC, za poskytnutie programu na riadenie modelu výťahu pomocou PLC SAIA a za pripomienky počas tvorby ukázkových aplikácií.

V neposlednej rade patrí podĎakovanie rodinným príslušníkom a priateľom za pomoc pri korektúrach textu.

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD	10
I TEORETICKÁ ČASŤ	11
1 SCADA SYSTÉMY	12
1.1 ÚVOD DO SCADA SYSTÉMOV	12
1.2 ZAČIATKY SCADA SYSTÉMOV	14
1.2.1 Výhody jednoduchého SCADA systému typu snímač – panel	14
1.2.2 Nevýhody takýchto systémov	15
1.3 ZÁKLADNÉ PRINCÍPY MODERNÝCH SCADA SYSTÉMOV	15
1.3.1 DCS (Distributed Control System)	15
1.3.1.1 Výhody PLC/DCS SCADA systémov	16
1.3.1.2 Nevýhody PLC/DCS SCADA systémov	16
1.3.2 IED (Intelligent Electronic Device)	16
1.3.2.1 Výhody PC – IED systémov	17
1.3.2.2 Nevýhody PC – IED systémov	17
1.4 ARCHITEKTÚRA KOMUNIKAČNÝCH SIETÍ	18
1.4.1 Architektúra peer-to-peer	18
1.4.2 Multipoint architektúra	18
1.4.3 Architektúra bezdrôtových komunikácií	19
1.5 KOMUNIKAČNÉ ŠTANDARDY	20
1.5.1 RS-232	21
1.5.1.1 UART/USART	21
1.5.2 RS-485	21
1.5.2.1 RS- 485 zbernice	22
1.5.3 RS-422	22
1.5.4 USB	23
1.5.5 ETHERNET	23
1.5.5.1 Modbus TCP	25
1.5.5.2 Ethernet Powerlink	25
1.5.5.3 Profibus	26
1.5.5.4 Profinet	26
2 HMI (HUMAN – MACHINE INTERFACE)	27
2.1 JEDNOTKY OVLÁDANIA A INDIKÁCIE	27
2.1.1 LED indikátory	27
2.1.2 Páčkové ovládače	28
2.1.3 Moderné rozhrania	28
3 PROGRAMOVÉ VYBAVENIE	29

3.1	CONTROL WEB 6	29
3.2	TIRS.NET	30
3.3	WONDERWARE INTOUCH	31
3.4	PROMOTIC	31
II	PRAKTICKÁ ČASŤ	32
4	INŠTALÁCIA PROGRAMU CONTROL WEB 6	33
4.1	VLASTNÁ INŠTALÁCIA	33
4.2	ODLADENIE KOMUNIKÁCIE	33
5	UŽÍVATEĽSKÁ PRÍRUČKA K PROSTREDIU CONTROL WEB 6.....	35
5.1	PRACOVNÉ PROSTREDIE	35
5.2	INSPEKTOR PŘÍSTROJE	38
5.2.1	Popis najpoužívanějších parametrov	39
5.3	TVORBA APLIKÁCIÍ.....	41
5.3.1	Aplikácia reálneho času.....	41
5.3.2	Aplikácia riadená zmenou dát.....	41
5.4	ČASOVANIE A AKTIVÁCIA PŘÍSTROJOV SYSTÉMOV REÁLNEHO ČASU	42
5.5	NADVIAZANIE KOMUNIKÁCIE	43
5.5.1	Zavedenie ovládača	43
5.5.1.1	Mapovací súbor	43
5.5.1.2	Parametrický súbor pre PLC SAIA	44
5.5.1.3	Parametrický súbor pre PLC Tecomat	45
5.5.2	Definícia kanálov – premenných	47
5.5.3	Vzdialená komunikácia	47
5.5.3.1	Zdieľanie dátových sekcií.....	48
5.5.3.2	Zdieľanie prístrojov a procedúr	49
6	UKÁŽKOVÉ APLIKÁCIE	50
6.1	UKÁŽKOVÁ APLIKÁCIA KOMUNIKÁCIE S PLC TECOMAT.....	50
6.1.1	Panel VSTUPY / VÝSTUPY	51
6.1.1.1	Pripojenie a odpojenie stanice PLC	52
6.1.2	Panel REGISTRE	52
6.1.3	Panel STAV.....	53
6.1.4	Panel controls_panel	53
6.1.5	Popis použitých komunikačných kanálov.....	53
6.2	UKÁŽKOVÁ APLIKÁCIA KOMUNIKÁCIE S PLC SAIA	55
6.2.1	Hlavný panel aplikácie.....	55
6.2.2	Panel Ovládanie	56
6.2.3	Panel Registre	57
6.2.4	Panel Stav CPU	57
6.2.5	Popis použitých komunikačných kanálov.....	58
6.3	UKÁŽKA DISTRIBUOVANEJ APLIKÁCIE.....	59
6.3.1	Aplikácia modely_server	60

6.3.1.1	Panel Výt'ah.....	61
6.3.1.2	Panel Výt'ah_CW.....	62
6.3.1.3	Panel Fontána	67
6.3.2	Aplikácia modely_klient	68
6.3.2.1	Panel Výt'ah_CW.....	69
6.3.2.2	Panel Fontána	70
6.3.3	Aplikácia výt'ah_klient	70
6.3.4	Popis zavedených kanálov v distribuovanej aplikácii	70
6.3.4.1	Kanály ovládača Vytah.....	71
6.3.4.2	Kanály ovládača Vytah_CW.....	71
6.3.4.3	Kanály ovládača Fontana.....	71
ZÁVER		72
ZÁVER V ANGLIČTINE.....		73
ZOZNAM POUŽITEJ LITERATÚRY		75
ZOZNAM POUŽITÝCH SYMBOLOV A ZKRATIEK		78
ZOZNAM OBRÁZKOV		79
ZOZNAM PRÍLOH		81

ÚVOD

Nik v dnešnej dobe nepoprie tvrdenie, že rozvoj vedy a techniky napreduje míľovými krokmi. To, čo sa v minulosti zdalo nedosiahnuteľné a futuristické je dnes samozrejmosťou, s ktorou sa stretávajú deti už od narodenia. Tak je tomu i v obore automatizácie, v obore riadenia technologických procesov a v nemalom odvetví systémov komunikácie a v odvetví informatiky. Predstavy ľudí vykonávajúcich ručnú reguláciu sú dnes minulosťou. Veda priniesla mnoho vylepšení a zaviedla do systémov riadenia a regulácie nemalé množstvo inteligencie. Zavedenie nového pojmu, vizualizácie technologických procesov, prinieslo nový rozmer do sveta automatizácie. Vznik operátorských panelov, priemyselných počítačov a inteligentných embedded zariadení a ich zavedenie do prevádzky prispelo k zvýšeniu efektivity, kvality, rýchlosti, presnosti výroby a v nemalom množstve tiež k šetreniu nákladov a ku zvyšovaniu ziskov.

Pri návrhu a realizácii systémov integrovanej automatizácie je rozhodujúcim prvkom úspechu vhodná voľba komunikačného systému. V snahe zabezpečiť kompatibilitu rôznych komponent bolo zavedených niekoľko komunikačných štandardov a súborov pravidiel komunikácie – protokolov. Pri nástupe požiadavku na riadenie a reguláciu systémov reálneho času sa začal klásť dôraz na rýchlosť a odozvu komunikácie. Dnes už existujú systémy spĺňajúce vysoké požiadavky na komunikáciu a na vývoji nových sa stále pracuje. Samotný súbor technických prostriedkov však nezabezpečí úlohu vizualizácie. Rovnaká dôležitosť sa kladie na programové vybavenie systémov.

Dôvodom vzniku tejto práce bola snaha priblížiť predstavu o rýchlo sa rozvíjajúcich systémoch vizualizácie bežnému užívateľovi, podať predstavu o zložení, o práci a o návrhu takéhoto systému. Cieľom je tiež priniesť základné informácie o programovom vybavení slúžiacemu k návrhu aplikácií vizualizácie a podeliť sa o získané poznatky počas práce v jednom z týchto systémov, v prostredí Control Web 6, produkte českej spoločnosti Moravské přístroje a.s. a tiež informovať užívateľa o možnostiach tohto produktu.

Snahou uvedených ukázkových aplikácií je ponúknuť predstavu o realizácii komunikácie medzi PC a PLC, poukázať ako na možnosti vizualizácie tak i na schopnosť systému plniť úlohu riadenia.

I. TEORETICKÁ ČASŤ

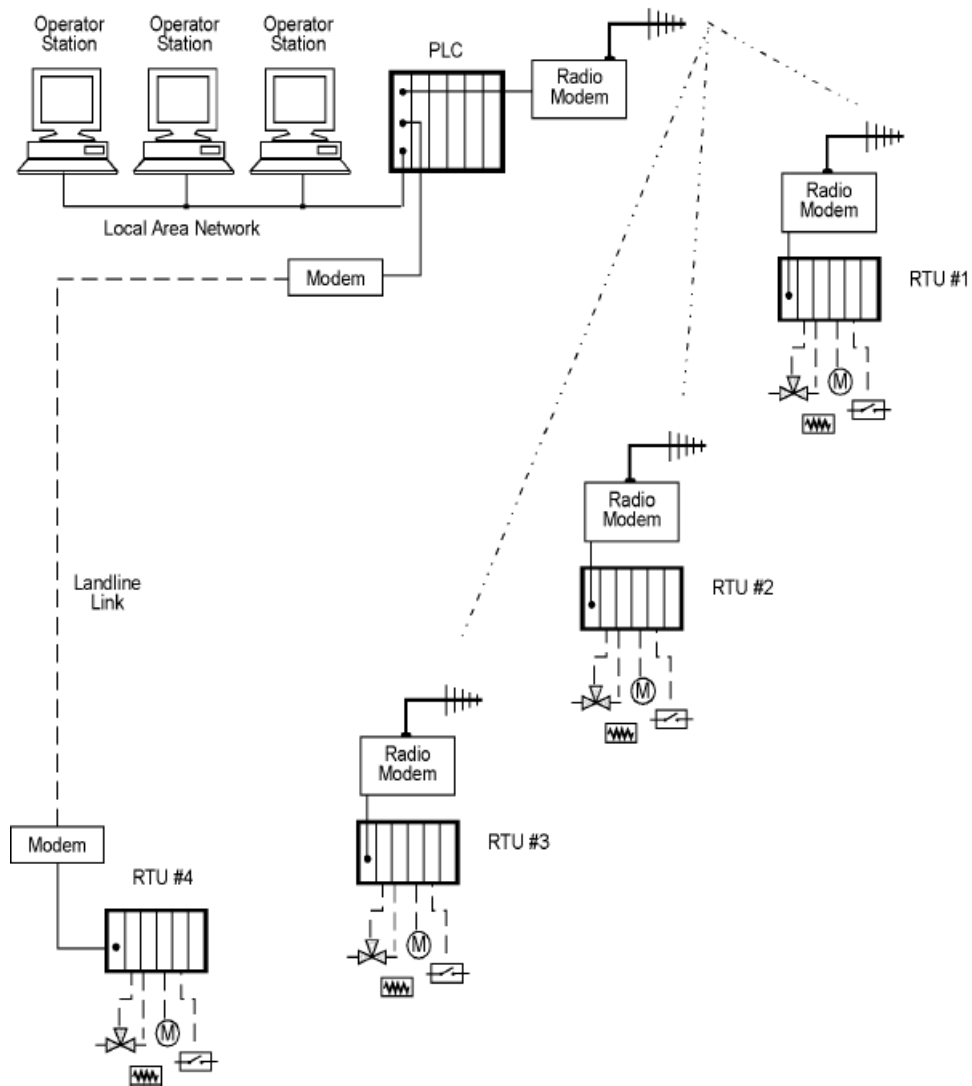
1 SCADA SYSTÉMY

1.1 Úvod do SCADA systémov

SCADA (Supervisory Control and Data Acquisition) systém je systém skladajúci sa z určitého množstva od seba vzdialených zariadení - RTU (Remote Terminal Unit), ktoré zberajú údaje z terénu a posielajú ich späť do centrálnej stanice prostredníctvom komunikačného systému. RTU, ako už sám názov napovedá, sú samostatné vzdialené koncové zariadenia, riadiace jednotky a jednotky zberu informácií, všeobecne obsahujúce mikroprocesor, ktoré monitorujú a riadia príslušenstvo nachádzajúce sa vo vzdialených miestach od centrálnej jednotky. Sú vybavené schopnosťou dynamického sťahovania riadiaceho programu z centrálnej jednotky, pričom je možná i lokálna zmena konfigurácie použitím prenosných programovacích zariadení. Najjednoduchšie prevedenia jednotiek RTU dokážu spracovať desať až dvadsať analógových a digitálnych signálov. Stredne veľké obsahujú sto digitálnych a tridsať až štyridsať analógových vstupov, veľké jednotky môžu tieto hodnoty presiahnuť niekoľkonásobne. Všetky jednotky sú spojené pomocou komunikačného systému s centrálnou jednotkou. Centrálna jednotka umožňuje zobrazenie získaných dát a taktiež povoľuje operátorovi vzdialene zasahovať do úloh riadenia. Presné a časovo veľmi rýchlo získané dáta (real time) prispievajú k optimalizácii všetkých operácií výrobného procesu. Ďalšími výhodami sú efektívnosť, spoľahlivosť a bezpečnosť. [3]

Špecifikom systémov SCADA je práve vzdialenosť medzi centrálnou jednotkou a koncovými zariadeniami. Na otázku, aká veľká táto vzdialenosť môže byť, neexistuje presná odpoveď. Každopádne však platí, že komunikačný systém je najkritickejším prvkom celého regulačného systému.

Úspešné zavedenie SCADA systému závisí od využitia osvedčených a spoľahlivých technológií spolu s adekvátnym školením všetkých operátorov prístupujúcich k systému. Naopak slabým miestom systému môže byť neadekvátna integrácia rozmanitých komponentov systému, nadbytočná zložitosť, nespoľahlivosť zariadení a tiež použitie neovereného programového vybavenia.



Obr. 1. Ukážka typického SCADA systému [3]

Päť stupňov hierarchie komplexnejších SCADA systémov:

- Meracie zariadenia, prístroje ku kontrole a obsluhu.
- Zostava koncových zariadení a RTU.
- Komunikačný systém.
- Centrálné jednotky.
- Oddelenie spracovania komerčných dát.

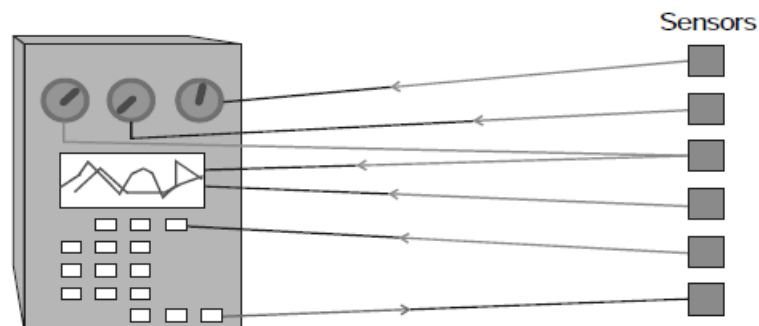
RTU zaisťuje rozhranie pre oblasť analógových a digitálnych signálov na strane každého vzdialeného zariadenia. Komunikačný systém zaisťuje komunikáciu medzi centrálnou

jednotkou a ostatnými vzdialenými zariadeniami. Tento komunikačný systém môže byť realizovaný rádiovým prenosom, prenosom po telefónnej linke, mikrovlnným bezdrôtovým prenosom alebo satelitným prenosom. Pre zabezpečenie efektívneho a optimálneho prenosu dát sú používané špecifické protokoly a rôzne filozofie detekcie chýb.

Centrálne stanica zhromažďuje dáta od rôznych RTU a všeobecne poskytuje operátorovi rozhranie pre zobrazenie informácií a riadenie všetkých vzdialených staníc. V rozľahlých telemetrických systémoch sú hlavné riadiace jednotky rozdelené na master a sub-master jednotky. V takýchto prípadoch sub-master jednotky zhromažďujú dáta z koncových zariadení a tie posielajú master jednotkám. [3]

1.2 Začiatky SCADA systémov

SCADA systémy majú rovnako dlhú históriu ako ktorékoľvek iné systémy riadenia. U prvých SCADA systémov boli namerané dáta spracovávané a vyhodnocované meracími prístrojmi, často grafickými zapisovačmi, zabudovanými v rôznych typoch jednoduchých operátorských panelov.



Obr. 2. Ukážka jednoduchého systému [3]

1.2.1 Výhody jednoduchého SCADA systému typu snímač – panel

- Jednoduchosť, bez potreby programového vybavenia, CPU, RAM alebo ROM.
- Snímače sú priamo prepojené s meracími prístrojmi alebo signalizačnými jednotkami.
- V mnohých prípadoch nízka nákladovosť na rozšírenie systému.

1.2.2 Nevýhody takýchto systémov

- Pri pripojení väčšieho množstva snímačov stráca operátor prehľad o systéme.
- S rastom systému stúpa náročnosť na jeho prevedenie.
- Zmena, prestavba systému je takmer nemožná.
- Neustála prítomnosť operátora je nutná.

1.3 Základné princípy moderných SCADA systémov

V moderných výrobných a priemyselných procesoch vo verejnom i súkromnom sektore je často potrebné prepojiť jednotlivé zariadenia navzájom od seba vzdialené. Môže sa jednať o vzdialenosť niekoľkých metrov až po tisícky kilometrov. Tieto prepojenia, tzv. telemetria, sa využívajú na prenos požiadaviek a tiež príjem informácie zo vzdialenej stanice.

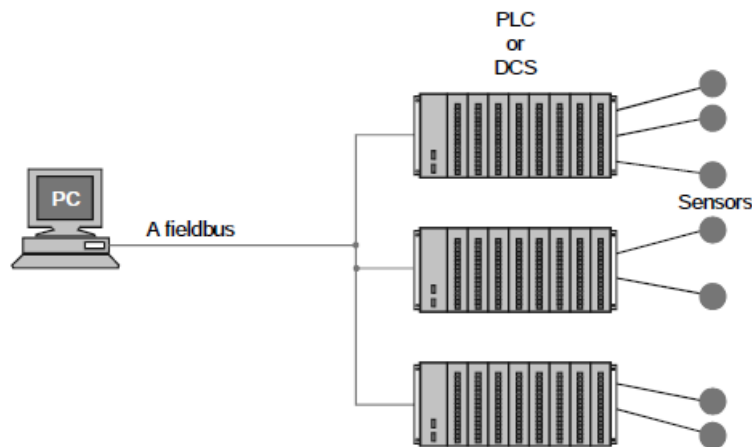
SCADA systém je kombináciou telemetrie a získavania dát. Zahŕňa zber informácií a ich spätný prenos do centrálnej jednotky, realizáciu všetkých potrebných analýz, samotné riadenie a následné zobrazenie výsledku spracovaných dát na viacerých operátorských paneloch a rôznych displejoch. [3]

1.3.1 DCS (Distributed Control System)

V počiatkoch systémov zberu dát bola používaná na kontrolu výroby v podnikoch reléová logika. S príchodom CPU a iných elektronických zariadení sa do výrobných procesov začala zavádzať digitálna elektronika namiesto reléovej logiky. PLC (Programmable Logic Controller) sa stali a stále sú najviac používanými prvkami priemyselných regulačných obvodov. Vzrast potreby monitorovať a riadiť väčšie množstvo dát a zariadení priniesol vznik DCS, ktoré sa vyznačujú väčšou inteligenciou a menším kompaktným prevedením. Ďalším dôvodom vzniku bola snaha použiť inteligentné zariadenia ako kľúčové komponenty systému.

V DCS systémoch je zber dát a funkcia riadenia zastúpená väčším počtom distribuovaných mikroprocesorových jednotiek umiestnených v blízkosti meracích zariadení alebo zariadení získavajúcich dáta. DCS systémy prerástli až do systémov prevádzajúcich veľmi sofistikované analógové riadenie. Vysoká integrácia operátorských rozhraní MMI

(Man Machine Interface) priniesla možnosť jednoduchého prístupu k ľubovoľnej časti systému. Dátové zbernice sú obyčajne schopné dosiahnuť rýchlosť 1Mbit/s až 10Mbit/s. [3]



Obr. 3. Systém s využitím PLC/DCS [3]

1.3.1.1 Výhody PLC/DCS SCADA systémov

- Zaznamenávanie a ukladanie veľkého množstva dát.
- Mnoho možností zobrazovania dát.
- Možnosť pripojiť tisíce senzorov do jedného systému i v rozsiahlych priestoroch.
- Možnosť simulácie použitím reálnych nameraných dát.
- Možnosť spracovania rôznych druhov signálov.

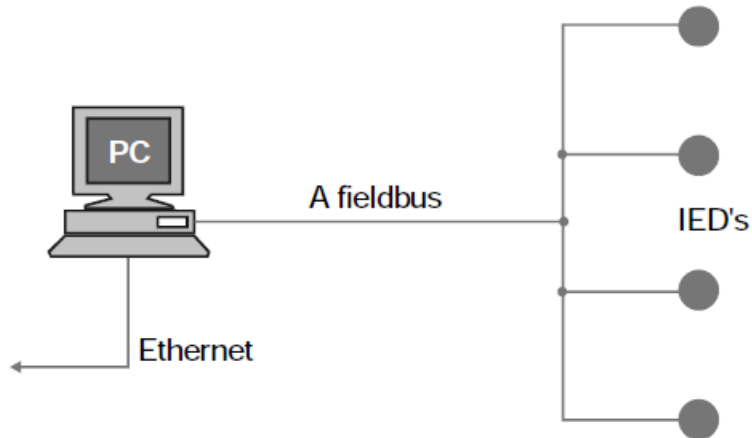
1.3.1.2 Nevýhody PLC/DCS SCADA systémov

- Vyššia komplikovanosť oproti systémom typu snímač – panel.
- Nutnosť znalosti rôznych operátorských zručností (analytik, programátor).
- Stála prítomnosť veľkého množstva káblových prepojení.

1.3.2 IED (Intelligent Electronic Device)

S nárastom požiadaviek pre menšie a elegantnejšie systémy boli vyvinuté snímače s inteligenciou PLC a DCS systémov. Tieto zariadenia sú známe ako IED. IED zariadenia sú priamo pripojené na zbernicu. Sú vybavené dostatočnou inteligenciou na získanie dát,

komunikáciu s ostatnými zariadeniami a uchovanie potrebnej časti programu. Jedno zariadenie môže obsahovať viac snímačov. Vyznačujú sa prítomnosťou analógového vstupu, analógového výstupu, PID regulátora, systému komunikácie a pamäte programu. [3]



Obr. 4. Systém s využitím IED [3]

1.3.2.1 Výhody PC – IED systémov

- Minimálna elektroinštalácia.
- Operátor má prehľad o systéme až do úrovne snímačov.
- Prijaté dáta z koncových zariadení môžu obsahovať informácie ako sériové číslo, model, dátum inštalácie zariadenia.
- Všetky zariadenia sú typu plug and play, inštalácia a výmena sú jednoduché.
- Menšie zariadenia znamenajú menej fyzického priestoru pre systém zberu dát.

1.3.2.2 Nevýhody PC – IED systémov

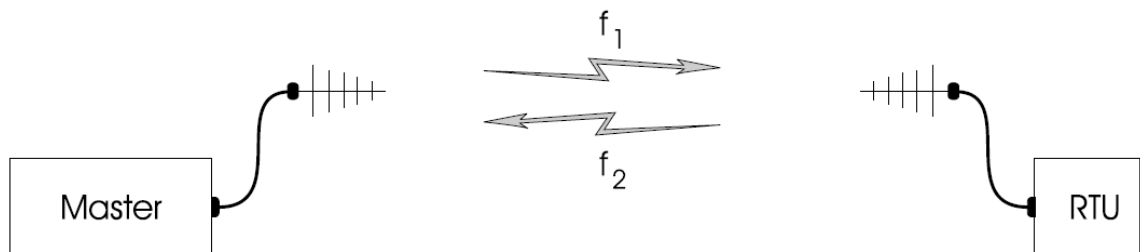
- Viac sofistikované systémy si vyžadujú vyššiu kvalifikáciu zamestnancov.
- Ceny snímačov sú vyššie.
- IED sú závislé na komunikačnom systéme.

1.4 Architektúra komunikačných sietí

Nasledujúce tri uvedené architektúry fyzických realizácií môžu byť ľubovoľne kombinované a použité pre tvorbu komunikačného systému.

1.4.1 Architektúra peer-to-peer

Jedná sa o najjednoduchšie usporiadanie, kedy dochádza k výmene dát len medzi dvoma stanicami. Jedna stanica môže byť nastavená ako master, druhá ako slave. Táto architektúra umožňuje komunikáciu medzi dvoma stanicami ako v plne duplexnom móde (vysielanie a príjem na dvoch samostatných frekvenciách), tak i v polo duplexnom móde s použitím jedinej frekvencie. [2]

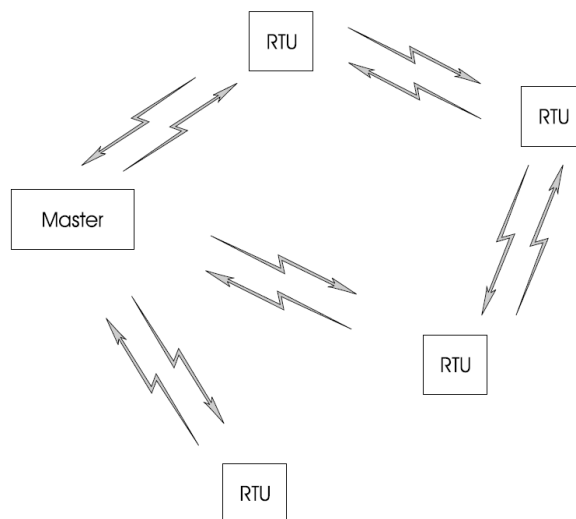


Obr. 5. Architektúra peer-to-peer [2]

1.4.2 Multipoint architektúra

Konfigurácia sa vyznačuje jednou master jednotkou a viacerými slave jednotkami. Obyčajne dáta prechádzajú medzi hlavnou master jednotkou a každou z ostatných slave jednotiek. Ak dve slave jednotky požadujú prenos informácií medzi sebou, master jednotka slúži ako prostredník.

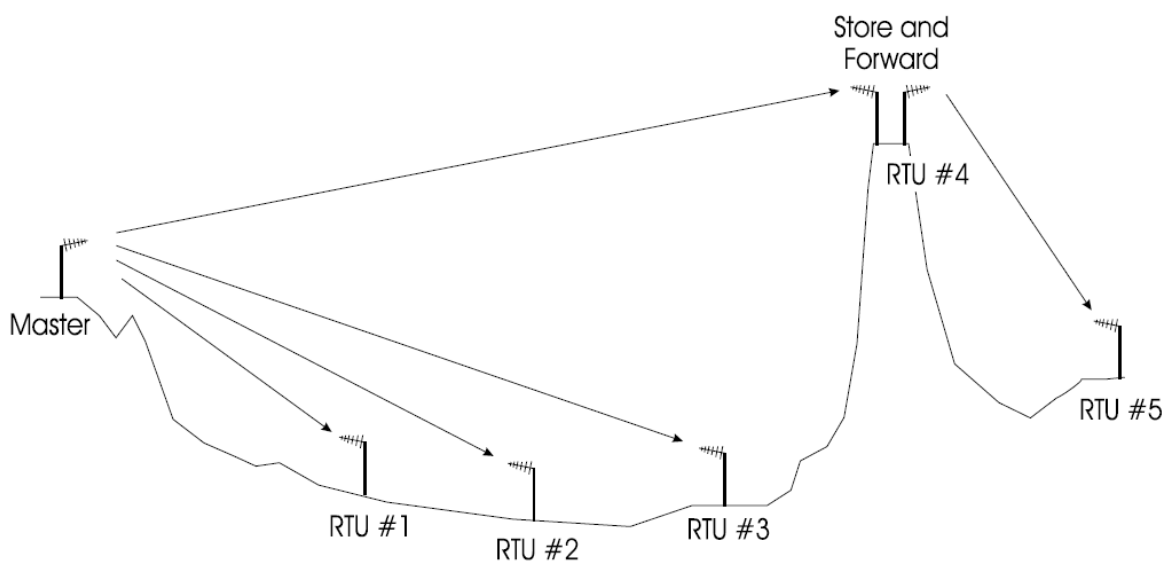
Alternatívou tohto zapojenia je použitie peer-to-peer prepojení medzi všetkými stanicami. Vyžaduje si zložitejšie rozloženie a prepojenie zariadení a použitie sofistikovanejších protokolov k udržaniu bezporuchovej komunikácie i v kolíznych situáciách, ako je napríklad požiadavka o komunikáciu medzi dvoma stanicami v rovnakom čase. [2]



Obr. 6. Viacnásobné prepojenia [2]

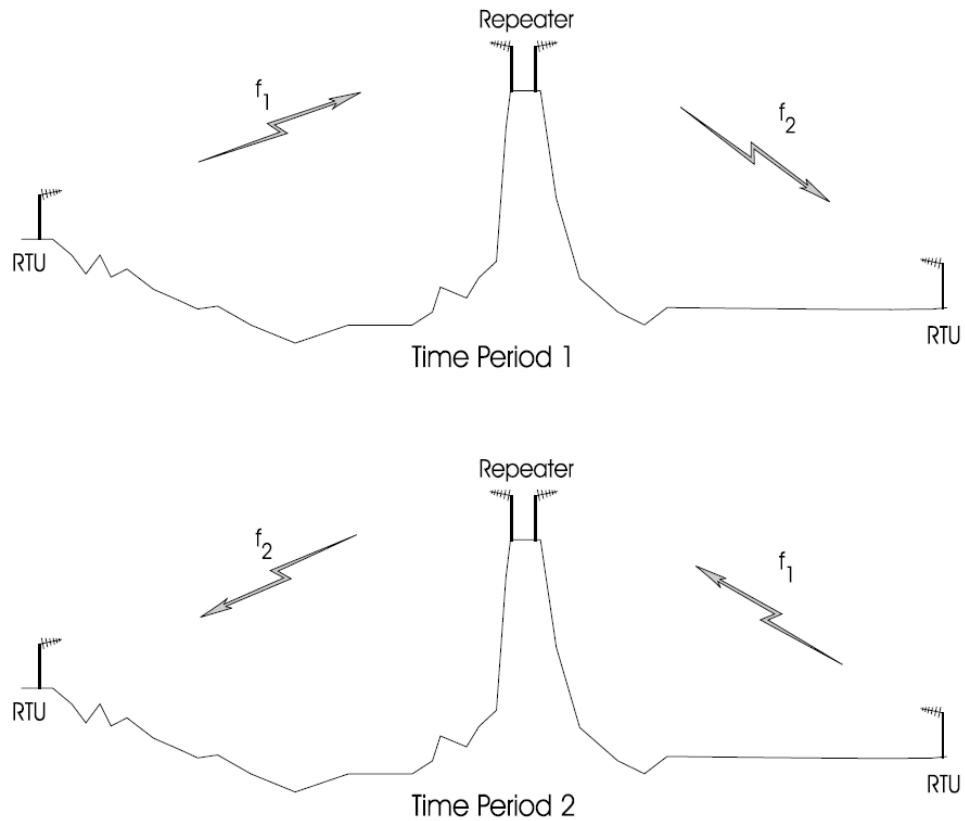
1.4.3 Architektúra bezdrôtových komunikácií

Existujú dva spôsoby pre prenos informácií medzi stanicami, ktoré nie sú v priamom spojení. Jedným z riešení je využitie stanice, ktorá je v dosahu hlavnej jednotky a tiež v dosahu vzdialenej jednotky, ktorá nie je v spojení s hlavnou jednotkou. Hovoríme o prostredníku v procese komunikácie. Prostredná stanica zhromažďuje dáta posielané od centrálnej jednotky a po prijatí všetkých dát ich posiela koncovej stanici. V tomto prípade sa však nejedná o súčasný prenos dát. Prijímanie a následné vysielanie prenášaných dát na tej istej frekvencii znižuje rýchlosť prenosu. Výhodou je menšia cenová náročnosť.



Obr. 7. Využitie jednotky ako prostredníka v komunikácii [2]

Preferovanou cestou u bezdrôtových komunikácií na väčšie vzdialenosti je použitie opakovacích zariadení. Takéto zariadenie je umiestnené na geograficky najvyššom bode. Prijímaný rádiový signál je súčasne vysielaný ďalej na odlišnej frekvencii. [2]



Obr. 8. Použitie opakovacích zariadení [2]

1.5 Komunikačné štandardy

Komunikačný štandard definuje konštrukčné prevedenie komunikačnej siete a ustanovuje platné pravidlá, na základe ktorých je riadená komunikácia. Medzi najznámejšie štandardizačné organizácie patria:

- IEEE (Institute of Electrical and Electronics Engineers).
- ISO (International Organization for Standardization).

1.5.1 RS-232

Sériové rozhranie určené na prenos dát medzi dvoma zariadeniami na vzdialenosť desiatok metrov, pri nízkych komunikačných rýchlostiach, maximálne 20kbit/s. Prenos dát je plne duplexný – dáta môžu byť prenášané oboma smermi v rovnakom okamžiku. Signály sú reprezentované napätovými úrovňami voči zemi. Sú prenášané po jednotlivých vodičoch, pričom jeden vodič, voči ktorému sú napätové úrovne vzťahnuté, je spoločný. Fyzicky môže byť na jeden port pripojené jedno zariadenie. Maximálna možná použitá dĺžka vodičov vzhľadom k odporu vedenia je približne 30 metrov. [4], [19]

V prevažnej väčšine sa používa prenos asynchrónny. Pri synchronnom prenose je jedna linka vyhradená na prenos časovacích impulzov a nie je možné ju využiť na prenos dát.

1.5.1.1 UART/USART

Universal (Synchronous) Asynchronous Receiver/Transmitter je zariadenie, ktoré zabezpečuje všetky detaily sériovej komunikácie. Počítače, operačné systémy a programovacie jazyky zahŕňajú podporu pre programovanie sériovej komunikácie bez nutnosti znalosti detailov UART architektúry. Po výbere portu a príprave dát na odoslanie zariadenie UART posiela dáta, bit po bite v požadovanom formáte a pridáva štart, stop, v prípade nutnosti paritný bit. Taktiež umožňuje vygenerovať prerušenie procesora a tým upozorniť aplikáciu na príchod dát popripade inú udalosť. [4]

Ak komunikujúce zariadenie neobsahuje UART alebo ak ich potrebujeme viac ako je v konfigurácii zariadenia, môžeme pripojiť externé UART alebo simulovať jeho funkciu v programovom kóde. Dnešné PC obsahujú multifunkčný čip, ktorý zahŕňa ako UART pre sériovú komunikáciu, tak podporu paralelnej komunikácie a mnohé ďalšie systémové komponenty. [4]

1.5.2 RS-485

Rozhranie určené k prenosu dát medzi viacerými zariadeniami až do vzdialenosti 1200 m, rýchlosťou od 300bit/s až do 10Mbit/s. Štandardne môže byť pripojených až 32 zariadení na jeden segment. Z pohľadu realizácie ide o veľmi úsporné prevedenie. Pre každý signál využíva jediný krútený pár vodičov. Informácia je prenášaná na základe rozdielu potenciálu medzi vodičmi krúteného páru. Na krátkych vzdialenostiach nemusí byť tieneny. [19]

1.5.2.1 RS-485 zbernice

Zbernica je digitálna linka navrhnutá pre funkcie monitorovania a riadenia systému. Príkladom sú:

- BACnet – využíva sa v systémoch automatizácie budov, vyvinutá spoločnosťou ASHRAE (American Society of Heating, Refrigerating and Air – Conditioning Engineers).
- DIN 66348 – zbernica určená pre meracie a testovacie účely v priemysle, vyvinula Association of Measurement Bus Users.
- IEEE-1118 – určená pre všeobecné použitie medzi master a slave jednotkami. Pôvodne Intel BITBus.
- Profibus (Process Fieldbus) – použitie v systémoch budov, vo výrobe, v systémoch riadenia, navrhnutá spoločnosťou Profibus International.
- SAE J1708 – aplikovaná v automobilových systémoch, od Society of Automotive Engineers.

1.5.3 RS-422

Štandard sa radí niekde medzi RS-232 a RS-485. Ako prenosové médium je použitá dvojica krútených párov vodičov, zaisťujúcich prenos dát oboma smermi. Často sa uplatňuje ako náhrada komunikačnej trasy RS-232 na dlhších vzdialenostiach.

Na potlačenie nežiaducich odrazov na komunikačnej linke RS-485 slúžia ukončovacie impedancie. Typická hodnota je 120Ω , čo odpovedá menovitej impedancii vedenia. Ak je vedenie realizované na vzdialenosť jednotiek až desiatok metrov, nie je použitie ukončovacích impedancií nevyhnutné. Rovnako je tak i pri štandardnom režime RS-422. V režime multi-master je impedančné prispôsobenie rovnaké ako u RS-485. [17]

Veľkou nevýhodou pri prepojení zariadení od rôznych výrobcov je neexistencia jednotného komunikačného protokolu. Normy pre RS-232 a RS-485 definujú len fyzickú vrstvu a najnižšiu komunikačnú vrstvu. Spôsob prenosu dát je preto daný zvyklosťami jednotlivých výrobcov.

1.5.4 USB

Cieľom vývoju štandardu USB bolo nahradiť množstvo rôznych konektorov jedným. Prioritou bolo tiež zjednodušenie spôsobu komunikácie periférií a odstránenie konektorov s kolíkmi. Rozhranie USB tvoria štyri vodiče: VCC, GND, D+, D-. Vodiče D sú použité k diferenciálnemu prenosu dát. Privedené napätie 5V na VCC umožňuje napájať menšie zariadenia priamo zo zbernice, bez nutnosti externého napájania. Pri použití hviezdicovej architektúry siete môžeme pripojiť do portu USB rozbočovač. Maximálny počet sériovo pripojených rozbočovačov je päť, pričom celkový počet zariadení spolu s rozbočovačmi nesmie presiahnuť počet 127. [15]

Povolená dĺžka prepojovacích káblov u verzie USB 1.1 je tri metre. U verzii USB 2.0 je táto vzdialenosť predĺžená až na päť metrov. USB 1.1 umožňuje prenos na rýchlostiach 1,5Mbit/s a 12Mbit/s. Teoretická priepustnosť u verzii USB 2.0 stúpla takmer štyridsaťnásobne, na 480Mbit/s. [15]

1.5.5 ETHERNET

Pre použitie v riadiacich systémoch pracujúcich v reálnom čase nie je klasický kancelársky Ethernet vhodný. Keďže prístup k prenosovému médiu je riadený kolíznou metódou CSMA/CD (Carrier Sense Multiple Access/Collision Detection), poradie doručených rámcov nemusí odpovedať poradiu vyslaných a doba uskutočnenia komunikácie je závislá na zaťažení siete. [20]

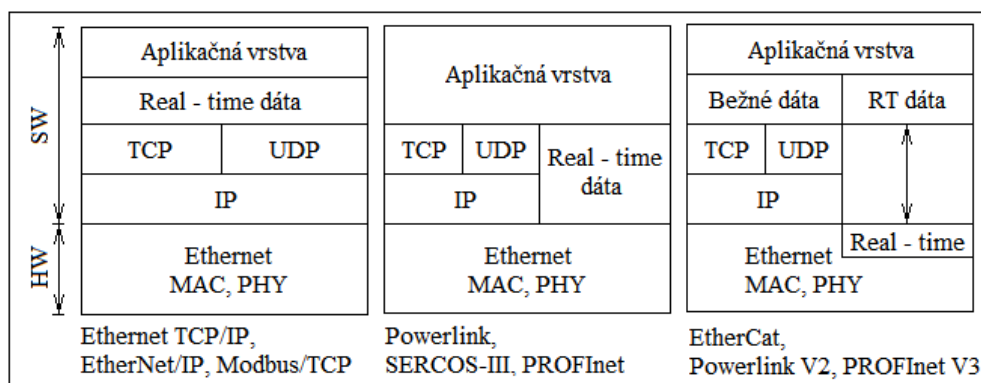
Tento problém rieši priemyselný Ethernet, napríklad RTnet, PROFINET od firmy Siemens, EtherCAT firmy Beckhoff a ďalšie. V priemyselnom Ethernete je kolízna metóda prístupu k médiu nahradená vlastnou deterministickou metódou, splňujúcou požiadavky na HRT (Hard Real – Time) systém. Ethernet POWERLINK definuje tzv. komunikačný cyklus. V sieti existuje práve jeden riadiaci uzol a až 239 riadených uzlov. Každý uzol uchováva svoje dáta v slovníku objektov a komunikáciu prevádza cyklicky. Riešenie vychádza zo štandardu CANopen. [7], [20]

Priemyselný Ethernet má na rozdiel od kancelárskeho napájanie 24V DC. Koncové zariadenia musia pracovať bez použitia ventilátorov pri vyšších teplotách nepretržite, pričom musia byť prispôbené na montáž na nosnú lištu. Musia vykazovať veľmi krátku dobu prístupu.

Rozlišujeme líniovú architektúru, hviezdicovú, kruhovú, redundantnú hviezdicovú, zdvojenú kruhovú a kombinácie týchto prevedení. Líniová nie je preferovaná vzhľadom k tomu, že pri výpadku v jednom mieste dôjde k výpadku celej siete. Hviezdicová sa používa častejšie, i keď výpadok centrálného prvku môže tiež viesť k výpadku siete. Riešením je redundantné prepojenie na úrovni centrálného prvku. Pri výpadku spojov v kruhovej architektúre dochádza k automatickej rekonfigurácii siete, ktorá však môže trvať dostatočne dlhý čas na to, aby došlo k rozpadu komunikácie. Riešením je architektúra zdvojeného kruhu, ktorá poskytuje najvyšší stupeň funkčnej pohotovosti. Absencia štandardu pre redundantné prevedenia komunikačných sietí vedie k vlastným riešeniam jednotlivých výrobcov. Príkladom je redundantná sieť Hiper – Ring od firmy Hirschmann, ktorá zaručuje zotavenie do 500ms i pri veľkých vzdialenostiach medzi prepínačmi. [11]

Medzi ďalšie štandardy priemyselného Ethernetu patrí napríklad Powerlink, Ethernet/IP a High Speed Ethernet (HSE), ktorý je súčasťou štandardu Foundation Fieldbus.

Fyzické prevedenie priemyselného Ethernetu sa od kancelárskeho líši nielen samotným prevedením, ale aj architektúrou, determinizmom a funkčnou spoľahlivosťou. Prevedenie káblov a konektorov odpovedá potrebám prevádzky, môže dosahovať krytie až IP 65. Je teda plne vyhovujúci pre použitie v DCS. Vysoká a stále rastúca rýchlosť ethernetových budičov skraca dobu odozvy na komunikačnom kanáli. Použitie UDP (User Datagram Protocol) namiesto TCP (Transmission Control Protocol) je vhodné pre časovo kritické úlohy, pretože táto nespojovaná služba umožňuje posielanie správy hneď v ďalšom takte po odhalení kolízie. Používa sa tiež PTP (Precision Time Protocol) podľa štandardu IEEE 1588, ako najúčinnjší a zároveň lacný mechanizmus pre zaručenie synchrónneho prenosu správ. [7], [14]



Obr. 9. Prehľad architektúr systémov s použitím priemyselného Ethernetu

1.5.5.1 Modbus TCP

Protokol bol vyvinutý v roku 1979 spoločnosťou Modicon, neskôr bol jeho vývoj spojený s organizáciou Modbus.org, ktorej cieľom bolo zdokonaľiť a propagovať Modbus ako prvý priemyselný ethernetový protokol. Ide o otvorený stabilný protokol, ktorý bol používaný hlavne v aplikáciách typu klient – server pre komunikáciu v reálnom čase. Doba odozvy je dve desatiny až jedna sekunda. Výkonnosť protokolu zvyšuje implementácia služby I/O scanning, ktorá zabezpečuje rýchlu výmenu dát medzi vstupno-výstupnými jednotkami. Nedokáže však zaistiť synchronizáciu v reálnom čase v rozsiahlych distribuovaných aplikáciách. Túto úlohu zabezpečuje služba Global Data, založená na princípe takzvaného protokolu Real Time Publish - Subscribe, ktorý umožňuje zariadeniam svoje dáta do tohto protokolu zapisovať a iným opätovne tieto dáta získať. Nie je potrebné dáta žiadať, pretože sú poskytovateľom pri každej zmene posielané. [5]

1.5.5.2 Ethernet Powerlink

Pre svoju funkčnosť nevyžaduje žiadne špeciálne technické vybavenie, pretože vychádza zo štandardu Ethernet. Dosahuje výborné vlastnosti v aplikáciách reálneho času a to s čisto programovým riešením. Jeho prednosť spočíva v dôslednom časovom rozdelení prenosového cyklu na dve časti. Prvá, izochrónna časť, slúži na prenos časovo kritických dát v reálnom čase. Asynchrónna časť slúži na prenos časovo nekritických dát protokolom IP. Ďalšou výhodou, ktorá prispieva k determinizmu je modifikovaný spôsob tvorby bezkolíznych domén. Používajú sa rozbočovače s opakovaním správ namiesto štandardne používaných prepínačov, pretože tieto vnášajú do izochrónnej rýchlej časti cyklu nežiaduce oneskorenia. Ďalším novým prvkom je oddelenie segmentov siete pracujúcich v reálnom čase od segmentov siete, ktoré prácu v reálnom čase nepodporujú. [12]

Ethernet Powerlink umožňuje vytvárať siete ľubovoľných architektúr. V jednej doméne reálneho času môže byť až 240 uzlov. Plne zaručuje deterministický prenos dát v reálnom čase. Najkratšia doba cyklu je 200 μ s s neistotou synchronizácie do 1 μ s. Vďaka priamej komunikácii všetkých uzlov navzájom môžu byť dáta odoslané z jedného uzla okamžite prijaté ktorýmkoľvek ďalším uzlom v sieti. Zariadenia je možné do siete pripájať alebo zo siete odpájať za chodu, bez negatívneho ovplyvnenia vlastností funkcie reálneho času. [10]

1.5.5.3 Profibus

Patrí k najrozšírenejším komunikačným zberniciam. Rozlišujeme tri varianty:

- Profibus DP – určený pre automatizáciu výroby. Dosahuje prenosovú rýchlosť až 12Mbit/s. Komunikačným médiom je linka RS-485 alebo optické vlákno. Využíva sa pre rýchlu výmenu relatívne malého obsahu dát v líniovej architektúre.
- Profibus FMS – pri rovnakom fyzickom prevedení ako Profibus DP umožňuje prenos vyšších objemov dát, pri prenosovej rýchlosti do 500kbit/s.
- Profibus PA – vyznačuje sa asynchrónnym prenosom s konštantnou prenosovou rýchlosťou 31,25kbit/s a možnosťou použitia v EEx prostrediach (prostredie s nebezpečenstvom výbuchu). Architektúra stromová alebo líniová.

Na zbernicu môže byť pripojených viacero riadiacich staníc, ktoré si medzi sebou posielajú telegram. Tento ich oprávňuje k nadviazaniu komunikácie s inou stanicou. [8], [16]

1.5.5.4 Profinet

Otvorený komunikačný štandard medzinárodnej organizácie Profibus International, založený na štandarde Ethernet. Tento nástupca úspešných zbernicových a komunikačných systémov (Profibus, Interbus) a priemyselného Ethernetu zastrešuje všetky úspechy predošlých systémov a je určený pre vyššiu úroveň riadiacich systémov s prenosmi väčších dátových objemov. Poskytuje odstupňovanú komunikačnú architektúru, pokrývajúcu celý rozsah podnikovej automatizácie od časovo nenáročných priemyselných procesov až po špecifické nároky aplikácii v oblasti riadenia pohybu. [9]

Je založený na štandardoch informačnej techniky (TCP/IP – pre časovo nekritické aplikácie), ale navyše poskytuje možnosť komunikácie v reálnom čase. SRT (Soft Real - Time) zastrešuje aplikácie reálneho času v automatizácii s dobou cyklu zbernice 10 ms. Izochrónna komunikácia IRT (Isochronous Real - Time) je určená pre veľmi náročné úlohy riadenia pohybu, ktoré vyžadujú prísne deterministické chovanie. Doba cyklu zbernice je v tomto prípade jedna milisekunda. Všetky protokoly je možné pri komunikácii ľubovoľne kombinovať. Používaným prenosovým médiom je buď tienový krútený pár vodičov alebo optický kábel. Dosahuje prenosovú rýchlosť až 100Mbit/s. [9], [14]

2 HMI (HUMAN – MACHINE INTERFACE)

Rozhranie človek – stroj je miestom stretu technológie s človekom. Či ide o stlačenie tlačidla nejakého ručného prístroja alebo ovládanie palubnej dosky dopravného lietadla, úloha HMI je rovnaká – urobiť funkcionality technológie jasnou, pochopiteľnou a užívateľsky prívetivou. Zahŕňa všetky funkcie potrebné pre kontrolu procesu a dohliada na prevádzku stroja. Operátorovi umožňuje v rámci behu proces zastaviť alebo spustiť, poprípade definuje podmienky a postup opätovného spustenia procesu. Taktiež umožňuje riadenie ovládať a vykonávať zmeny pre dosiahnutie spoľahlivého chodu, ako aj monitorovať priebeh procesu. V prípade neočakávaných alebo kritických udalostí môže operátor svojím zásahom vykonať nápravné akcie skôr ako dôjde k vážnejšiemu narušeniu situácie na základe signalizačných upozornení, poprípade môže prejsť na plne manuálne riadenie. [21]

2.1 Jednotky ovládania a indikácie

Všetky navrhnuté signalizačné a ovládacie prvky musia spĺňať potrebné požiadavky na prevedenie. Farebné označenia sú presne definované normou. Použitie v priemyselnej prevádzke si vyžaduje vyššie požiadavky na krytie a robustnosť prevedenia, zvýšenú odolnosť a spoľahlivosť.

2.1.1 LED indikátory

Signálne LED svetlá sú vhodné pre malé prístroje alebo pri potrebe veľkého počtu indikátorov. Vyznačujú sa svojou malou spotrebou, malými rozmermi, odolnosťou voči otrasom, vibráciám a dlhou životnosťou. Svetelné majáky a stĺpiky umožňujú sledovať stav zariadenia na veľké vzdialenosti, v rozsahu 360°. Nesúlad medzi prevádzanými inštrukciami a aktuálnym stavom indikujú blikajúce svetlá. Blikajúce a rotujúce majáky plnia úlohu informovanosti s najvyššou prioritou. V miestach, kde je prítomnosť operátora veľmi dôležitá alebo je použitie iných signalizačných zariadení nedostatočné, zabezpečujú signalizáciu húkačky a sirény. [21]

2.1.2 Páčkové ovládače

Používajú sa na ovládanie pohybu v jednej alebo dvoch osiach, pre pohyb v dvoch až ôsmich smeroch. Prevedenia sú rôzne, jedno alebo dvoj kontaktné pre každý smer, s návratom alebo bez návratu do východiskovej polohy.

2.1.3 Moderné rozhrania

Pokrok v oblasti vývoja HMI priniesol rozhrania s novými možnosťami, moderným a atraktívnym prevedením a jednoduchou ovládateľnosťou. Medzi výhody vstavaných ovládacích panelov patrí možnosť nastavovania parametrov produktov, prehľad stavov akčných členov a tiež napríklad voľba jazyka užívateľského prostredia. Sú navrhnuté pre konkrétny účel.

Obrazkové a klávesnicové terminály slúžia pre všeobecné použitie v ľubovoľných aplikáciách. Komunikujú s procesom prostredníctvom komunikačnej zbernice. Sú určené pre stacionárnu i prenosnú montáž.

Do priemyselného prostredia s elektromagnetickým rušením sú určené robustne a odolne prevedené priemyselné PC, či už v kompaktnom alebo modulárnom prevedení. Umožňujú realizovať i veľmi zložité riešenia automatizácie, rozsiahle matematické úlohy, simulácie a modelovanie úloh reálneho času, adaptívne i optimálne riadenie i zber veľkého objemu dát. Rôzne typy obrazkových panelov umožňujú grafické vizualizácie vo vysokom rozlíšení. [21]



Obr. 10. Príklady prevedení priemyselných PC [21]

3 PROGRAMOVÉ VYBAVENIE

Programové vybavenie preberá kľúčovú úlohu v riadiacich systémoch. Je vrstvou systému, ktorá ho činí unikátnym a špecifickým podľa potrieb užívateľa. Program implementuje riadiace algoritmy a dáva systému vlastnú inteligenciu.

3.1 Control Web 6

Aplikácie systému Control Web majú k dispozícii možnosti komunikácie prostredníctvom siete Ethernet, USB, RS-232, RS-422, RS-485, Wi-Fi, Bluetooth, súčasne môže obsahovať internetový http server. Má k dispozícii webového klienta, umožňuje posielat' e-maily, posielat' a prijímať správy SMS, komunikovať cez GPRS, spolupracovať so zariadeniami plug and play, SQL databázami, s OPC a ActiveX komponentmi. Môže využívať trojrozmernú grafiku, dokáže priamo komunikovať so stovkami typov zariadení prostredníctvom natívnych ovládačov. [1]

Control Web má k dispozícii všetky potrebné komponenty pre tvorbu aplikácií vizualizácie technologických procesov. Obsahuje zobrazovacie a ovládacie prvky, alarmy a archívy, podporuje zber dát a iné. Poskytuje skutočnú programovateľnosť a otvorenú komponentovú architektúru. Virtuálne prístroje nie sú pevne dané a zabudované v systéme, ale každý z týchto prístrojov je linkovanou dynamickou knižnicou, rozpoznanou pri štarte systému. Množinu virtuálnych prístrojov je možné ľubovoľne rozširovať. [1]

Control Web umožňuje tvorbu aplikácií reálneho času. Hodnoty každého vstupno-výstupného kanála sú získavané presne v čase, kedy tieto hodnoty požaduje jeden alebo skupina virtuálnych prístrojov. Na presnosť časovania takýchto aplikácií je kladený veľký dôraz. Časovanie virtuálnych prístrojov môže byť na základe zmeny dát alebo v presne definovanom čase, poprípade presne definovaných časových sekvenciách.

Control Web umožňuje tvorbu distribuovaných aplikácií, v ktorých sprístupňuje ľubovoľný dátový element všetkým spojeným aplikáciám v sieti Ethernet a tiež Internet. Aktivácia i dynamické metódy rozhrania virtuálnych prístrojov môžu byť volané na diaľku.

Plnohodnotný http server umožňuje vizualizáciu technológie pomocou štandardov http a html, pomocou ľubovoľného webového klienta pod rôznymi operačnými systémami.

Server dokáže dynamicky generovať stránky podľa stavu technológie, pričom dokáže pomocou html a http technológií i riadiť.

Použitie programu Control Web nezávisí na použitom technickom vybavení. Jeho natívne ovládače umožňujú pracovať mnohonásobne efektívnejšie ako napríklad ovládače DDE (Dynamic Data Exchange). Je samozrejme tiež podporovaný spolu s OPC (OLE for Process Control) a ďalšími štandardmi pre priemyselné automaty, samostatné moduly a meracie karty.

Ku zvýšeniu produktivity práce prispievajú aplikačné knižnice, predlohy virtuálnych prístrojov, nástroje pre ladenie aplikácií, funkcie UNDO/REDO v grafickom vývojovom prostredí, virtuálne prístroje pre statické riadenie výrobných procesov, interný binárny databázový systém pre možnosť práce bez SQL servera, celoobrazovkový editor 3D scény a tiež možnosť práce Control Web systému ako služby operačného systému. [22]

3.2 TIRS.NET

Produkt spoločnosti CORAL s.r.o. určený pre tvorbu menších a stredne veľkých aplikácií, kde sa kladie dôraz na rýchlosť a jednoduchosť riešenia. Skladá sa z dvoch logicky oddelených častí. Viditeľná časť slúži na návrh vizuálnej stránky aplikácie, neviditeľnú časť predstavuje jadro systému a napríklad komunikačné a databázové moduly. Funkcia jadra systému nie je závislá na prihlásení užívateľa, preto môže plniť svoje dispečerské úlohy hneď po štarte počítača.

Umožňuje tvorbu rôznych typov aplikácií, od jednoduchých až po distribuované systémy s mnohými vzdialenými klientmi s komunikáciou i po Internete. Produkt nerozlišuje medzi vývojovou a runtime verziou, možnosti užívateľa závisia od jeho oprávnení. Je tak možné upravovať alebo vytvárať nové vizuálne stránky z ľubovoľnej klientskej stanice za plného chodu aplikácie. Klient sa môže prepínať medzi prevádzkovým a editačným módom.

Systém TIRS.NET umožňuje komunikáciu s ľubovoľnou technológiou. Komunikačné knižnice si môže užívateľ vytvárať sám pomocou nástroja Microsoft Visual Studio. [18]

3.3 Wonderware InTouch

Najnovšia verzia produktu prináša nové objektové grafické možnosti založené na technológii ArchestrA. Aplikácie vytvorené v starších verziách sú s novými kompatibilné vďaka integrovanému vývojovému prostrediu Wonderware IDE (Integrated Development Environment) aplikačného servera Wonderware Application Server.

Objektovo orientovaná grafika ArchestrA je reprezentovaná novým typom symbolov ArchestrA Symbols. Symboly sa vytvárajú v editore, ktorého množina geometrických tvarov obsahuje mnohé zložité objekty. Tieto objekty vektorovej grafiky sú prehľadne zobrazované v stromovej štruktúre s možnosťou zoskupovania a hromadnej manipulácie. Implementovaný je tiež editor pre tvorbu animácií.

Kombinácia programovej platformy Wonderware System Platform s produktom InTouch významne uľahčuje, štandardizuje a urýchľuje vytváranie i veľmi rozsiahlych automatizačných a informačných riešení pre zvýšenie efektivity, flexibility a výkonnosti výrobných operácií. [13]

3.4 Promotic

Je komplexným objektovo orientovaným programovým nástrojom od firmy MICROSYS pre tvorbu distribuovaných, decentralizovaných, flexibilných a otvorených aplikácií vizualizácie i riadenia v rôznych odvetviach priemyselnej automatizácie. Zabudované komunikačné ovládače spĺňajú požiadavky na vysokú konfigurovateľnosť, možnosť definície priority správ a podporu modemových a bezdrôtových prenosov.

Promotic Web server zabezpečuje všetky základné služby ako prenos dát, http protokol i autentizáciu užívateľov. Zabudovaný jazyk VBscript so syntaxou Visual Basic slúži pre zápis užívateľom definovaných algoritmov. K odladeniu slúži VBscript debugger.

Pre tvorbu aplikácií slúžia dva editory, editor aplikácie a editor obrazov. Systém je koncipovaný formou stavebnice objektov, ich vlastností a metód. V editore aplikácie užívateľ vytvára hierarchický strom objektov podľa vlastných predstáv. Editor grafických obrazov ponúka súbor elementárnych a komplexných grafických prvkov a poskytuje veľa možností v tvorbe operátorských obrazov. [6]

II. PRAKTICKÁ ČASŤ

4 INŠTALÁCIA PROGRAMU CONTROL WEB 6

4.1 Vlastná inštalácia

Pred začiatkom inštalácie programu Control Web 6 v laboratóriu 54/303 bolo uskutočnené odinštalovanie jednej z predošlých verzii, Control Web 5 na všetkých stanicach. Tento postup však nie je nutný, staršie i nové verzie programu sa pri používaní na tej istej stanici neovplyvňujú, pričom sa ale užívateľ môže stretnúť so spätnou nekompatibilitou u projektov vytvorených v novších verziách. Control Web 6 je určený pre 32 bitové operačné systémy MS Windows.

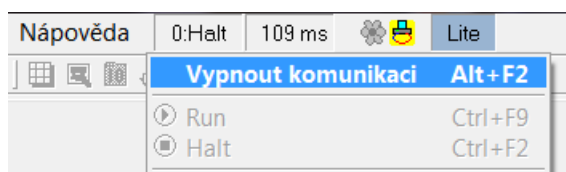
Počas priebehu inštalácie sa užívateľ nestretne so žiadnymi výnimočnými nastaveniami, ktoré by bolo nutné dokumentovať. Sprievodca prevedie užívateľa krok po kroku celou inštaláciou. Po ukončení inštalácie je užívateľ vyzvaný k registrácii programu. Po zadaní licenčného kľúča je potrebné produkt aktivovať. Aktiváciu je možné uskutočniť prostredníctvom Internetu na adrese <http://register.mii.cz/activation>. Užívateľ získa od spoločnosti Moravské přístroje a.s. aktivačný kľúč, pomocou ktorého dokončí aktiváciu produktu. Po inštalácii samotného programu je nutné doinštalovať balíčky ovládačov pre zariadenia, s ktorými sa bude komunikovať. V učebni 54/303 bola vykonaná inštalácia ovládačov pre komunikáciu s PLC od spoločností Teco a.s. a Saia-Burgess Controls Ltd. V oboch prípadoch je nutné po ukončení inštalácie previesť aktiváciu produktov obdobným spôsobom ako v prípade aktivácie programu Control Web 6. Následne je produkt pripravený k plnohodnotnému použitiu.

4.2 Odladenie komunikácie

Po úspešnom ukončení inštalácie celého programového vybavenia sa môže užívateľ pri pokuse o nadviazanie komunikácie s PLC od spoločnosti Saia-Burgess Controls Ltd. stretnúť s problémom pri výbere správneho komunikačného ovládača - SAIA Communication Driver. V prípade prítomnosti dvoch a viacerých inštalácií rôznych verzií tohto ovládača dochádza k nerozhodnosti, ktorý z ovládačov použiť. Preto je nutné dbať na to, aby pracovná stanica neobsahovala viac ako jednu inštaláciu tohto produktu, inak nebude komunikácia nadviazaná. Po zabezpečení tejto podmienky prebieha komunikácia medzi rôznymi programami a samotným PLC súčasne, nezávisle na sebe a bez

obmedzení. Ovládač SAIA Communication Driver je súčasťou programového balíčka inštalácie programovacieho prostredia PG5 a nemá nič spoločné s ovládačom od spoločnosti Moravské přístroje a.s.

V prípade PLC od spoločnosti Teco a.s. nie je potrebné riešiť okolnosti komunikačného ovládača, ale je nutné podotknúť, že nie je možná súčasná komunikácia medzi PLC, systémom Control Web 6 a programovacím prostredím Mosaic samotného PLC. Vyžadovaným krokom pred nadviazaním komunikácie s prostredím Control Web 6 je ukončenie komunikácie s PLC z prostredia Mosaic.



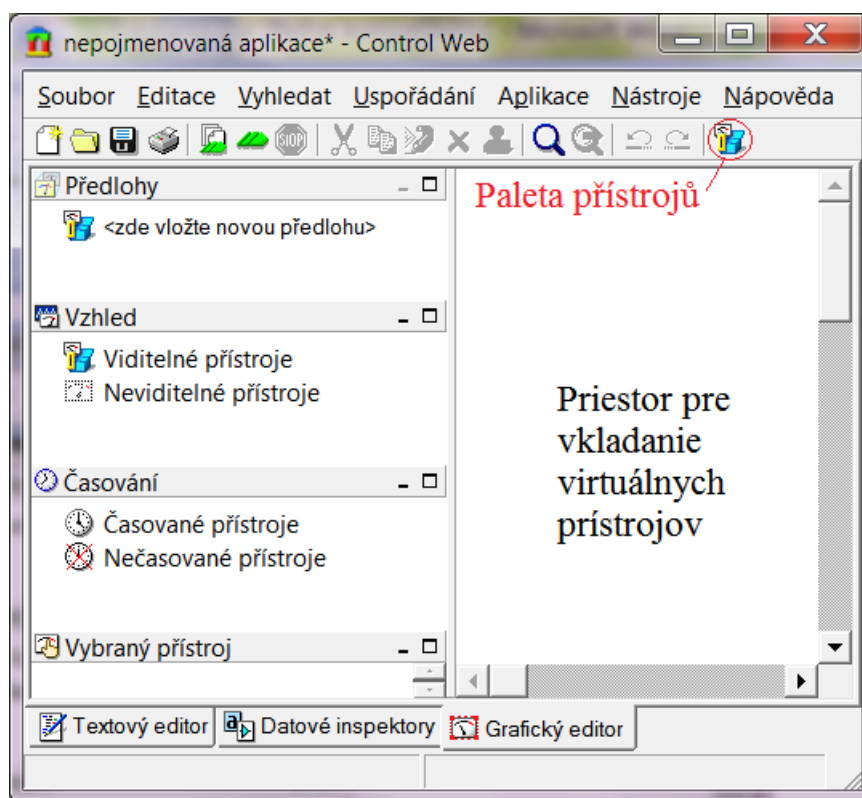
Obr. 11. Zmena stavu komunikácie v prostredí Mosaic

5 UŽIVATEĽSKÁ PRÍRUČKA K PROSTREDIU CONTROL WEB 6

Aplikácie, ktoré sú vytvorené v prostredí Control Web 6 a nasadené v prevádzke sa spúšťajú pomocou verzie Control Web Runtime. Pre návrh, vývoj a ladenie aplikácií je potrebná vývojová verzia, ktorá je opísaná v tejto príručke. Vývojová verzia umožňuje tvorbu i vizualizáciu funkčných riadiacich systémov.

5.1 Pracovné prostredie

Najviac využívaným prostredím systému Control Web 6 je *Grafický editor*. Jeho ekvivalentom je *Textový editor*. Oba sú vzájomne prepojené, takže pri editácii jedného z nich sa zmeny prejavajú i v druhom editore. Grafické prostredie je užívateľsky prívetivejšie, no pri vývoji aplikácií užívateľ určite ocení i možnosti textového editora.



Obr. 12. Ukážka prostredia systému Control Web 6

Pravá časť obrazovky slúži pre tvorbu grafickej podoby aplikácie. Do tohto okna môžeme postupne vkladat' všetky potrebné virtuálne prístroje a vzájomne ich prepájať s cieľom dosiahnutia požadovaného výsledku. Farbu pozadia je možné ľubovoľne meniť. Virtuálne prístroje vyberáme z ponuky, ktorá sa zobrazí po kliknutí na ikonu *Paleta prístrojů*.

Každý nový prístroj vložený do poľa musí mať definovaného svojho vlastníka. Vlastník je iný prístroj, ktorý tvorí pozadie, na ktorom sa nový prístroj nachádza. Pri jednoduchých aplikáciách bude postačovať, ak budú mať všetky použité prístroje rovnakého vlastníka. Táto voľba bude popísaná neskôr. Treba však podotknúť, že pri nezvolení žiadneho vlastníka bude automaticky zvolené ako vlastník pozadie a aplikácia nepobeží v žiadnom okne, ale priamo v prostredí MS Windows.

Ľavá časť obrazovky sa delí na niekoľko hlavných častí, z ktorých každá obsahuje svoju vlastnú stromovú štruktúru. Do týchto štruktúr sa postupne rozdeľujú použité virtuálne prístroje podľa umiestnenia a použitého časovania.

- *Předlohy* – do tohto poľa môžeme vložiť prístroje, ktoré sa v aplikácii vyskytujú viacnásobne, vždy s rovnakými nastaveniami. Pri neskoršom použití stačí prístroj vybrať priamo z tejto ponuky a nie je potrebné ho vyberať z palety a následne nastavovať.
- *Vzhled* – delí sa na *Viditelné přístroje* a *Neviditelné přístroje*. Prístroje sú usporiadané hierarchicky, podľa vlastníka. Prístroje, ktoré majú byť neviditeľné sa neumiestňujú do pravej časti obrazovky, ale je nutné ich z palety presunúť priamo do ľavej časti, *Neviditelné přístroje*.
- *Časování* – v tejto časti sú prístroje delené na *Časované přístroje* a *Nečasované přístroje*. U jednotlivých prístrojov sa nachádza priamo informácia o spôsobe a dobe časovania. Prístroje sú usporiadané podľa svojho druhu.

V dolnej časti sa nachádzajú tri záložky. Tieto umožňujú užívateľovi prepínať sa medzi textovým a grafickým editorom a dátovými inšpektormi.

Vytvorenú aplikáciu spustíme pomocou ikony zelenej šípky a zastavíme pomocou červenej ikony *STOP*. Pri vývoji aplikácie je vhodné zvoliť si ukončenie behu aplikácie uvedením príkazu `system.StopApplication()`; do procedúry `OnWindowClose()` panela pozadia.

Pri výbere ľubovoľného prístroja umiestneného v aplikácii sa tento zvýrazní vo všetkých oknách, čo zabezpečuje prehľadnosť a rýchlu manipuláciu s prístrojom. Vybraný prístroj je označený červeným rámkom. Pomocou myši alebo šípok klávesnice je možné meniť jeho polohu. Červeným prerušovaným rámkom je označený jeho vlastník. Pri výbere vlastníka

rozhoduje umiestnenie ľavého horného rohu prístroja. Pomocou klávesy *Insert* vlastníka určíme, poprípade kombináciou *Shift + Insert* vlastníka zrušíme.

Ponuka *Datové inspektory* je rozdelená podobne ako grafický editor na ľavú a pravú časť. V pravej časti nájdeme hlavný prehľad všetkých položiek ktoré máme k dispozícii. Pri pohybe po jednotlivých položkách ľavej časti sa v pravej zobrazuje prehľadná tabuľka s odpovedajúcimi parametrami, ktoré môžeme podľa požiadaviek nastavovať. Pri práci v ľavej časti je vhodné mať vybratú ponuku zobrazenia všetkých parametrov.



Obr. 13. Rozšírená ponuka parametrov

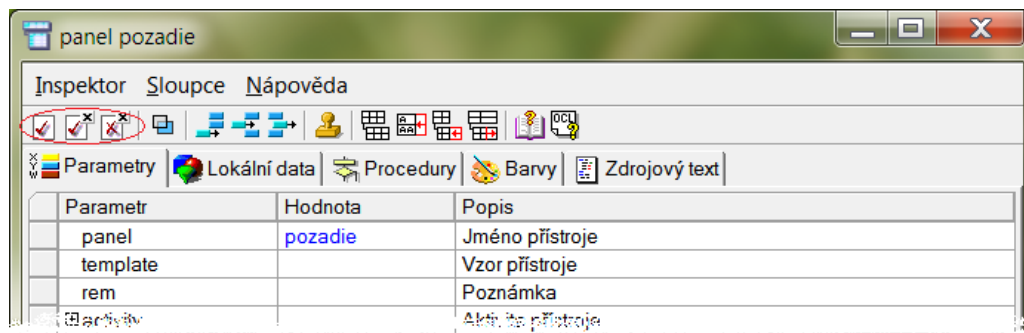
Najdôležitejšie nastavenia sa nachádzajú v sekciách:

- *Ovladače* – v tejto sekcii sú definované všetky použité ovladače na komunikáciu s PLC. Tvorba parametrického a mapovacieho súboru je popísaná v kapitole 5.5.1.
- *Datové elementy* – sekcia obsahuje všetky použité dátové elementy, ktoré sú rozdelené na podskupiny:
 - *const* – premenné, ktoré sa za behu aplikácie nemenia.
 - *var* – skupina premenných zvoleného dátového typu. Hodnoty je možné za behu aplikácie meniť.
 - *channel* – kanál definuje prepojenie medzi aplikáciou a vstupno-výstupným zariadením.
 - *expression* – v tejto sekcii je možné definovať celé matematické výrazy, ktoré budú v aplikácii volané viacnásobne. Daný výraz je zapísaný len raz a vyhodnotený pri každom použití. Každý výraz má definované svoje jednoznačné meno. Používa sa ako premenná.

Pri tvorbe aplikácie je nutné dodržiavať jednoznačnosť mien jednotlivých virtuálnych prístrojov, premenných, kanálov, názvu ovládačov i definovaných výrazov. Pomocou jednoznačných mien sú jednotlivé prvky používané. Pri písaní desatinných čísiel je nutné dodržiavať bodkovú notáciu.

5.2 Inspektor přístroje

Vyvoláme ho dvojitým kliknutím na ľubovoľný prístroj. Služi na nastavenie voliteľných parametrov, definíciu lokálnych dát, procedúr prístroja a ďalšie možnosti. Každý prístroj má svojho jednoznačného inšpektora. Je možné otvoriť inšpektora nad skupinou vybraných prístrojov, ale v ponuke voliteľných parametrov sa zobrazia len tie, ktoré sú spoločné. Na uloženie či zrušenie zmien služi trojica označených ikon.



Obr. 14. Ukážka inšpektora prístroja u panela s názvom „pozadie“

Záložka *Parametry* služi napríklad na nastavenie názvu, vzhľadu, aktivity prístroja, správania po spustení, na nastavenie vyhodnocovaného výrazu alebo výstupnej premennej, a mnoho ďalších možností.

V záložke *Lokální data* je možné definovať lokálne premenné, ktoré budú používané v rámci aktivácie prístroja. Lokálne premenné nie sú prístupné iným prístrojom. Pri zhode názvu lokálnej a globálnej premennej je v rámci aktivácie prístroja uprednostnená jeho definovaná lokálna premenná.

V záložke *Procedury* sú definované procedúry viažuce sa na udalosť prístroja, ale tiež užívateľom definované vlastné procedúry. Procedúry viažuce sa na nejakú udalosť prístroja sú vykonané v momente, keď daná udalosť nastane. V rámci procedúr môže byť definované aktivovanie iných virtuálnych prístrojov. Pri volaní procedúr iných prístrojov je nutné dodržať bodkovú notáciu. Príklady volania procedúr:

- *Start()*; - volanie procedúry s názvom *Start* rovnakého prístroja, v rámci ktorého je volaná.
- *Motor.Start()*; - volanie procedúry *Start* prístroja s názvom *Motor*.

Podobne, dodržaním bodkovej notácie môže byť volaná i procedúra prístroja inej aplikácie.

5.2.1 Popis najpoužívanějších parametrov

Nasledovné parametre sú spoločné pre väčšinu prístrojov.

- *template* – hodnotou tohto parametra môže byť názov prístroja rovnakého typu, ktorý bude predlohou k danému prístroju. Prístroj zdedí vlastnosti predlohy, stane sa jeho potomkom. Je vhodné využiť v prípade väčšieho množstva rovnakých alebo podobných prístrojov. Predlohy sa uvádzajú v záložke *Grafický editor* v časti *Předlohy*.
- *rem* – poznámky k prístroju. Poznámky zostávajú zachované i po preklade aplikácie. Poznámky v textovom editore nie sú zachovávané.
- *activity* – sekcia definuje spôsob aktivovania prístroja. Prístroj môže byť aktivovaný periodicky alebo časovačom. Parameter *condition* definuje podmienku aktivácie. V prípade nesplnení podmienky prístroj nie je aktivovaný.
- *gui* – súbor parametrov definujúcich vzhľad prístroja. Parameter *owner* definuje vlastníka prístroja. Každý prístroj sa zobrazuje na svojom vlastníkovi. Vlastník predstavuje tretiu súradnicu v usporiadaní prístrojov. V prípade panelov sa využívajú parametre sekcie *window*. Parametre *type*, *title* a *icon* umožňujú nastaviť typ okna, titulok okna a ikonu v titulku.
- *startup_options* – súbor parametrov definujúcich správanie prístroja pri štarte aplikácie. Umožňuje zakázať volanie procedúr a aktiváciu prístrojov uvedených v časti *receivers*. Parameter *output_action* riadi spôsob zápisu hodnôt u prístrojov (*control*, *switch*) do ich výstupov. Prístroj môže hodnotu do svojho výstupu zapísať, nezapísať alebo svoj stav podľa aktuálnej hodnoty svojho výstupu zmeniť.
- *send_same_data* – definuje spôsob zápisu rovnakých dát do výstupného kanála.

U prístroja typu *panel* je možné navyše definovať nasledovné parametre:

- *mode* – umožňuje nastaviť spôsob prekresľovania panelov s významom menšieho zaťaženia systému Windows. Nie je nutné túto hodnotu nejako špecifikovať, no v prípade veľkého množstva neprekrývajúcich sa prístrojov na jednom paneli je vhodné zvoliť hodnotu 'window_less'.

- *dv_id, dv_file* – dvojica parametrov špecifikujúcich pozadie panela. Ak prvý parameter nastavíme na hodnotu `‘.GRA’`, druhý parameter umožňuje výber obrázka pozadia panela. Slúži k zjednodušeniu návrhu grafiky zložitých systémov.
- *item* – definícia záložiek prekrývajúcich sa panelov.

Prístroj typu *indicator* sa vyznačuje špecifickým parametrom *expression*, ktorý definuje vyhodnocovaný výraz. Parametrom môže byť názov premennej dátového typu *boolean* alebo celý výraz. Parametre *true_icon* a *false_icon* slúžia na vloženie vlastných zobrazovaných ikon prístroja v oboch stavoch.

Prístroj typu *switch* sa vyznačuje nasledovnými parametrami:

- *output* – výstup prístroja. Premenná, ktorej hodnotu prístroj nastavuje.
- *mode* – ponúka možnosť výberu zobrazovania ikony alebo textu.
- *init_value* – predvolená hodnota prístroja pri štarte aplikácie.
- *true_icon, false_icon* – možnosť nastavenia zobrazovaných ikon v jednotlivých stavoch.
- *font* – možnosť nastavenia zobrazovaného textu.
- *logic* – definuje správanie prístroja, možnosť definície tlačidla alebo prepínača.
- *auto_update* – umožňuje spätné nastavenie stavu prístroja podľa výstupnej hodnoty v momente aktivácie.

Jednou z veľmi užitočných procedúr slúžiacich k zjednodušeniu ovládania aplikácií pri ich návrhu je procedúra *OnWindowClose()*. Príkaz *system.StopApplication()*; umožňuje zastavenie behu aplikácie pri zatvorení okna spustenej aplikácie. Tento príkaz je nutné uviesť v procedúre panela pozadia. Panel musí mať uvedené v parametroch *gui - window - type* hodnotu `‘normal’`.

5.3 Tvorba aplikácií

Pred začiatkom tvorby aplikácie je nutné rozhodnúť, či daná aplikácia bude aplikáciou reálneho času, alebo bude riadená zmenou dát. Hlavný rozdiel medzi týmito dvoma módmi tvorby aplikácií je v spôsobe časovania virtuálnych prístrojov.

5.3.1 Aplikácia reálneho času

Chod aplikácie je závislý na presnom načasovaní všetkých použitých virtuálnych prístrojov. Aplikácie slúžia pre priame riadenie strojov v reálnom čase. V momente, kedy nie je žiadny prístroj aktivovaný, systém nevykonáva žiadnu operáciu. Až v momente aktivácie prístroja je systém upovedomený o požiadavkách prístroja, sú mu pridelené požadované dáta alebo naopak, dáta jeho výstupu sú zapísané. Pri časovaní prístrojov je potrebné dbať na optimálnosť návrhu aplikácie vzhľadom na výkon procesora. Prioritou systému je dodržať časy aktivácií všetkých prístrojov zadané užívateľom, ale môže sa stať, že pri nevhodnom návrhu procesor nevyhoví požiadavkám užívateľa. Preto je dôležité, aby prístroje, ktoré nemusia byť časované, časované neboli a aby bolo správne rozhodnuté o spôsobe časovania.

5.3.2 Aplikácia riadená zmenou dát

Úlohou aplikácií tohto typu je vizualizácia procesu, nie jeho riadenie. Prístroje, až na niektoré výnimky, nie je možné časovať. Žiadna aktivita prístroja nespôsobí komunikáciu s perifériami. Systém sám sleduje dáta na vstupno-výstupných zariadeniach. Až po zmene týchto dát voči predošlému stavu aktivuje všetky prístroje, ktoré daný kanál využívajú. Prístroje sú rovnako aktivované pri zmene ľubovoľného typu dátového elementu. V prípade, že prístroj mení hodnotu určitej veličiny a sám túto veličinu spracováva, potom môže byť aktivovaný cyklicky sám sebou. K zabráneniu cyklickej aktivácie slúži parameter *passive*. U tohto typu aplikácií je treba podotknúť, že minimálnu periódu aktivácie, čas aktivácie ani poradie aktivácií prístrojov nie je možné zaručiť.

Je vhodné, zvoliť typ aplikácie na začiatku návrhu a postupovať podľa možností, ktoré daný typ aplikácie ponúka. Zmena módu je možná i počas tvorby, no keďže je filozofia tvorby oboch typov aplikácií úplne rozdielna, nie je vhodné takúto zmenu realizovať.

5.4 Časovanie a aktivácia prístrojov systémov reálneho času

Aktiváciou virtuálneho prístroja sa rozumie stav, kedy prístroj vykoná všetky svoje naprogramované udalosti. Spracuje vstupné dáta, vyhodnotí ich, poprípade dáta pošle k ďalšiemu spracovaniu. Súčasne sa vykonajú príslušné naprogramované procedúry prístroja. Prístroj môže svojou aktiváciou spôsobiť aktiváciu ďalších prístrojov. Aktiváciu prístroja riadime časovaním prístroja.

Základným spôsobom je časovanie periódou. O tento typ časovania sa stará sám systém. Hodnota parametra *period* môže byť uvedená číselne, v sekundách, alebo názvom zdefinovanej konštanty. V prípade potreby aktivácie prístroja jednorazovo, len pri spustení aplikácie, použijeme hodnotu 'infinite'.

Ďalšou možnosťou časovania je použitie iného prístroja, časovača. Časovač v každom svojom časovom kroku prechádza podskupinu svojich prístrojov a vykonáva ich aktiváciu v príslušnom poradí, čo časovanie periódou nezaručuje. Z toho plynie, že perióda aktivácie prístrojov časovaných časovačom je rovnaká ako perióda časovača. Prístroje časované časovačom majú uvedené v parametri *timer* názov svojho časovača. Druhy časovačov sú:

- *sequencer* – zabezpečuje aktiváciu prístrojov v presnom poradí.
- *iterator* – k aktivácii prístrojov v presnom poradí pridáva možnosť opakovania tejto postupnosti aktivácií. Aktiváciu prístrojov vykonáva cyklicky až do splnenia definovanej podmienky, preto sa môže ľahko stať nekonečnou slučkou. Používa sa zriedka.
- *selector* – časovač sa delí na niekoľko vetiev. Používa sa v aplikáciách obsahujúcich viacero navzájom sa prekrývajúcich panelov, kedy nepožadujeme aktiváciu prístrojov nezobrazeného panela. Časovač sa stará len o časovanie aktuálne viditeľných prístrojov, čím zabezpečuje menšie nároky aplikácie na procesor.
- *calendar* – zabezpečuje aktiváciu v presne stanovenom dátume a čase.

Aktiváciu prístroja prístrojom zabezpečíme uvedením názvu aktivovaného prístroja za parameter *receivers* prístroja aktivujúceho. Týmto spôsobom je možné aktivovať celú skupinu prístrojov, oddelených čiarkou. Aktivácia prístrojov časovaním sa nastavuje len u prístrojov vykonávajúcich operácie závislé na čase.

5.5 Nadviazanie komunikácie

Na komunikáciu s perifériami slúži ovládač pre konkrétne zariadenie. Po nakonfigurovaní ovládača je možné do aplikácie zaviesť komunikačné kanály, s ktorými sa následne pracuje ako s ľubovoľnými inými dátovými elementmi.

5.5.1 Zavedenie ovládača

Konfiguráciu ovládačov realizujeme v sekcii *Datové inspektory* v časti *Ovladače*. V pravej časti do tabuľky napíšeme nami zvolený názov ovládača, vyberieme typ ovládača z ponuky inštalovaných ovládačov a určíme cestu k mapovaciemu a parametrickému súboru. Ak je požadované spustenie aplikácie bez pripojeného zariadenia, je možné vybrať vhodný mód z ponuky módov ovládača:

- *run* – beh ovládača so zavedením všetkých kanálov. Zariadenie musí byť pripojené.
- *check* – simulačný beh, pri štarte aplikácie nie je ovládač zavedený, zápis i čítanie kanálov sú ignorované, zároveň je prekontrolovaná správnosť zápisu mapovacieho a parametrického súboru.
- *simulate* – beh ovládača ako v prípade *check*, ale kontrola potrebných súborov nie je prevedená.

5.5.1.1 Mapovací súbor

V tomto súbore je presne definovaný typ kanála alebo skupiny kanálov. Užívateľovi sú k dispozícii kanály od poradového čísla 100 nahor. Nižšie kanály si vyhradzuje výrobca pre vlastné využitie. Kanál môže byť typu:

- *input* – definuje vstupný tok informácií. Tento typ kanála využíva systém Control Web 6 pre príjem dát vyslaných z PLC.
- *output* – týmto typom kanála sú dáta posielané zo systému Control Web 6 do PLC.
- *bidirectional* – kanál vstupno-výstupný. Tento typ kanála sa využíva v prípade, ak je potrebné hodnotu čítať, spracovať a následne spätne zapísať. Využíva sa len v prípade nutnosti použitia, viac zaťažuje komunikačnú linku.

Ak kanálom prechádzajú dvojstavové hodnoty, je typu *boolean*. V prípade práce s ľubovoľnými hodnotami označíme kanál typom *real*. Podrobnejšia definícia dátového typu je uvedená v parametrickom súbore.

Nie je nutné vždy vytvárať nový mapovací súbor. Pri návrhu aplikácie je možné použiť vopred vytvorený mapovací súbor k danému zariadeniu a definovať v ňom len užívateľom navrhnuté kanály.

5.5.1.2 Parametrický súbor pre PLC SAIA

Ovládač je určený pre sériovú komunikáciu a komunikáciu v sieti Ethernet protokolom TCP/IP. Umožňuje čítať a zapisovať dáta registrov (R), časovačov (T), čítačov (T), binárnych výstupov (O), príznakov (F) a časových údajov (K). Hodnoty binárnych vstupov (I), registrov displeja (D) a stavu CPU (S) umožňuje len čítať.

Sekcia [ComPort] slúži na definíciu komunikácie. Parameter *DriverPath* definuje cestu k súborom podpory ovládača SAIA Communication Driver. Parameter *Com* môže nadobúdať hodnotu 'COM' v prípade štandardného sériového portu, 'SOCKET' v prípade komunikácie v sieti Ethernet, ďalej 'USB' alebo 'PC104'. Parameter *Device* udáva číslo sériového rozhrania. *Protocol* definuje použitý komunikačný protokol, môže nadobúdať hodnotu 'sbus' alebo 'p800'. *SBusMode* definuje mód pre protokol S-bus, najčastejšie používaná hodnota je 'parity'. *BaudRate* udáva komunikačnú rýchlosť. V prípade komunikácie v sieti Ethernet je dobré definovať parameter *Timeout*, ktorý udáva dobu čakania na odpoveď stanice. Hodnota sa udáva v milisekundách.

Sekcia [Channels] definuje bloky kanálov ovládača. Každý blok je reprezentovaný jedným riadkom zápisu. V skutočnosti každý blok definuje jeden celok, ktorý je možné preniesť v rámci jednej komunikácie so stanicou. Za definíciou „Block =“ je uvedených niekoľko parametrov, oddelených čiarkou. Prvý údaj je číselná hodnota adresy stanice. Táto adresa sa musí zhodovať s údajom uvedenom pri konfigurácii stanice v programovacom prostredí PG5. V prípade komunikácie v sieti Ethernet je toto číslo kombináciou adresy stanice, IP adresy a IP portu. Hodnota IP portu nie je povinná. Príklad zápisu je nasledovný: 01@10.5.16.185. Parameter *ChannelFrom* definuje číslo udávajúce prvý kanál bloku. Číslo posledného kanála je uvedené parametrom *ChannelTo*. Parameter *Media* definuje dátovú oblasť stanice (R,T,C,F,I,O,S,D,K). Číslo príslušnej dátovej oblasti definuje

parameter *Offset.SubType* definuje dátový typ pre oblasť registrov. Parameter *Bidirect* s hodnotou 'bidirect' je uvedený v prípade obojsmernej komunikácie. Ovládač sa sám postará o to aby nedošlo k nesprávneému zápisu dát. Dátové typy pre oblasť registrov môžu nadobúdať nasledovné hodnoty:

- uint – 32-bitová hodnota bez znamienka – jeden kanál.
- int – 32-bitová hodnota so znamienkom.
- float – 32-bitová *real* hodnota.
- uint16 – Dve 16-bitové hodnoty bez znamienka – dva kanály.
- int16 – Dve 16-bitové hodnoty so znamienkom.
- uint8 – Štyri 8-bitové hodnoty bez znamienka – štyri kanály.
- int8 – Štyri 8-bitové hodnoty so znamienkom.
- bool – Jedna *boolean* hodnota v jednom 32-bitovom registri – jeden kanál.
- bits – 32 *boolean* hodnôt v jednom registri – 32 kanálov.

Počet definovaných kanálov v parametrickom súbore musí odpovedať počtu definovaných kanálov v mapovacom súbore.

Sekcia [Options] nie je povinná. Obsahuje jediný parameter *FinalizeAfterEachBlock*. Prednastavená hodnota je 'false'. Hodnota 'true' znamená, že budú dáta predané aplikácii po uskutočnení komunikácie každého bloku. V prípade 'false' budú dáta predané aplikácii po dokončení prenosu všetkých blokov komunikujúcich v rámci daného časového kroku.

5.5.1.3 Parametrický súbor pre PLC Tecomat

Ovládač je určený pre sériovú komunikáciu a komunikáciu v sieti Ethernet. Okrem iného umožňuje čítanie vstupov (X), čítanie a zápis registrov (R), (S) a výstupov (Y). Je potrebné dbať na zápis veľkých a malých písmen.

Sekcia [Settings] slúži pre konfiguráciu komunikácie ovládača. Parameter *ComDriver* udáva názov knižnice ovládača (CWCOMM.DLL) a meno komunikačného portu (COMx). *NumRepeat* udáva počet opakovaní požiadavku na komunikáciu v prípade výskytu chyby. *Timeout* definuje čas čakania na odozvu zariadenia, v milisekundách. *SwapBytes* je príznak, ktorý určuje prehodenie poradia bajtov v slove. Parameter *InterMessageDelay* udáva

minimálne časové oneskorenie v milisekundách medzi dvoma komunikáciami. Ovládač neuskutoční ďalšiu komunikáciu skôr ako uplynie tento čas. Pri komunikácii v sieti Ethernet sa neuplatňuje. *Readback* slúži k automatickému filtrovaniu správ u niektorých zariadení, ktoré prijatú správu vysielajú späť (echo). Parameter *Ethernet* slúži na definovanie komunikácie v sieti Ethernet.

Sekcia [Channels] slúži k definovaniu komunikačných blokov. Každý blok má určené presne v poradí počiatočné číslo kanála, koncové číslo kanála, adresu stanice, dátovú oblasť stanice, dátový typ prenášaných údajov, adresu začiatku dát v dátovej oblasti, poprípade parameter *bidirect* pre obojsmerný kanál a možnosť *SwapBytes*. Dátový typ oblasti môže nadobúdať nasledovné hodnoty:

- *bits* – Predstavuje 8 *boolean* kanálov, pracuje s celým bajtom v pamäti PLC. Najmenšie prenesené množstvo je 1 byte.
- *sbit* – Slúži pre nastavenie jednotlivých bitov. Pri čítaní sa postupuje obdobne ako pri type *bits*, po celých bajtoch. Pri zápise viac ako osem bitov je efektívnejšie použitie typu *bits*.
- *bool8* – Predstavuje jeden *boolean* kanál v jednom bajte pamäti PLC.
- *uint8* – 8 bitové číslo bez znamienka.
- *int8* – 8 bitové číslo so znamienkom.

Rovnako sú definované 16, 32 a 64 bitové čísla.

- *float32* – 32 bitové desatinné číslo.

Adresovanie dátových oblastí prebieha po jednotlivých bajtoch.

Sekcia [Unit_X] nie je povinná. Používa sa v prípade, ak je na rovnakej zbernici pripojených viacero zariadení. V prípade požiadavky o komunikáciu s nefunkčnou jednotkou by sa celý riadiaci proces zdržoval čakaním na odozvu žiadosti o komunikáciu. Ovládač dokáže nefunkčné zariadenie rozpoznať, odpojiť ho a požiadavky na komunikáciu s týmto zariadením ignorovať. To, ktoré zariadenie odpojiť, respektíve opätovne pripojiť a na ako dlhú dobu, upresňujú parametre tejto sekcie. Parameter *UnitAdr* definuje adresu zariadenia, ktoré má byť odpojené. *StatusChannel* určuje číslo kanála, pomocou ktorého je možné zistiť stav zariadenia. Kanál je typu *real*, hodnotu 0 vracia v prípade odpojeného zariadenia.

Inak vracia hodnotu 1. *CheckTime* stanovuje dobu v sekundách, po ktorej bude odpojené zariadenie opäť automaticky pripojené.

Sekcia [comm] definuje nastavenia sériovej komunikačnej linky. Pre úspešnú komunikáciu je potrebné definovať prenosovú rýchlosť – *baudrate*, paritu – *parity*, počet dátových bitov – *databits* a počet stop bitov – *stopbits*. V prípade komunikácie v sieti Ethernet nie je potrebné túto sekciu uvádzať.

5.5.2 Definícia kanálov – premenných

Po zavedení ovládača v záložke *Datové inspektory* v sekcii *Ovladače* je nutné na nadviazanie komunikácie definovať komunikačné kanály. Komunikačný kanál definujeme v časti *Datové elementy-channel-Skalární*. Každý jeden kanál predstavuje premennú, ktorú chceme v aplikácii využívať. Tak, ako sú v mapovacom a parametrickom súbore definované jednotlivé kanály, ktorým odpovedajú presné číselné hodnoty, tak tieto hodnoty zapíšeme v časti *channel* spolu s názvom, s ktorým sa bude v aplikácii pracovať ako s premennou. V hlavnej definícii vlastností kanálov je potrebné nastaviť parameter *timeout*, ktorý definuje maximálne časové oneskorenie komunikácie. V prípade nulovej hodnoty nebude možné danú komunikáciu stihnúť a pri zmene stavov komunikovaných premenných v aplikácii Control Web sa tieto zmeny nemusia prejaviť na zariadení.

5.5.3 Vzdialená komunikácia

V systéme Control Web 6 je možné vytvárať aplikácie modulárne, distribuované a synchronizované. Modulárna aplikácia je aplikácia pozostávajúca z viacerých modulov. Modulom je každá jedna vytvorená aplikácia, preto je možné povedať, že najjednoduchšia aplikácia je jednomodulárna. Modulárna aplikácia vznikne vzájomným prepojením viacerých jednomodulárnych aplikácií. Distribuovaná aplikácia je aplikácia, ktorej jednotlivé prepojené moduly typu klient – server bežia na viacerých počítačoch, pričom majú presne definované závislosti. Klienti a server sú presne určení. U synchronizovaných aplikácií nie je presne definované, ktorý modul je klient a ktorý je server. Všetky moduly môžu byť súčasne klientmi a súčasne servermi, závisí to od spôsobu zdieľania svojich dát alebo prístrojov.

Serverová aplikácia beží nepretržite, pričom klient sa môže na server pripojiť, využívať jeho dáta, meniť ich, poprípade sa prihlásiť na iný server. Klient môže rovnako aktivovať prístroje servera alebo volať jeho procedúry. Server môže poskytovať dáta svojej dátovej

sekcie, popřípadě může klientovi poskytnout ovládní svých prístrojov. Existují dva spôsoby zdieľania:

- zdieľanie dátových sekcií
- zdieľanie prístrojov a procedúr

5.5.3.1 Zdieľanie dátových sekcií

Tento typ zdieľania umožňuje klientovi čítať dáta servera a tiež ich meniť. Komunikácia pri takomto type zdieľania prebieha obojsmerne, od servera klientovi a po zmene naspät' od klienta serveru. Klient tak napríklad sprostredkováva vzdialenú vizualizáciu deja prebiehajúceho na serveri. Medzi klientom a serverom je možné vytvoriť pevnú väzbu, kedy sa zmena na serveri prejaví zmenou u klienta a naopak, zmena u klienta sa rovnako prejaví i na serveri. Tento spôsob si vyžaduje časovanie použitých prístrojov na oboch stranách, čo zvyšuje nároky na systém komunikácie. Takýmto spôsobom je možné zdieľať ľubovoľnú dátovú sekciu. Prístup k procedúram prístroja však nie je povolený.

Pri realizácii tohto typu spojenia sa označí na strane servera dátová sekcia, ktorá má byť prístupná klientovi a nastaví sa nutné parametre *scope* a *client_access*. Dátová sekcia musí byť pomenovaná. Parameter *scope* nadobúda hodnotu 'shared_locally' v prípade, ak serverová aplikácia i klient bežia na jednej stanici. Pri tvorbe distribuovanej aplikácie bude nadobúdať hodnotu 'shared_remotely'. Parameter *client_access* definuje prístupové práva vzdialených klientov. Hodnota 'allow' umožní pripojenie ľubovoľného počtu nešpecifikovaných klientov. Skupinu a počet klientov je možné obmedziť definovaním povolených IP adries klientov. Pri práci servera s dátami dátových sekcií sprístupnených klientom nie sú definované žiadne obmedzenia.

V klientskej aplikácii využívajúcej dáta servera sa vytvorí nová dátová sekcia rovnakého typu ako sekcia zdieľaná serverom, s ľubovoľným názvom. Dátová sekcia zostane prázdna. Parameter *scope* nadobúda hodnotu 'imported_locally' v prípade lokálneho zdieľania na rovnakej stanici. Klient distribuovanej aplikácie nadobúda hodnotu 'imported_remotely'. Parameter *server* definuje server, na ktorý sa klient pripája. Musí obsahovať IP adresu servera, názov modulu – serverovej aplikácie, a tiež názov dátovej sekcie poskytovanej serverom. V prípade lokálnej aplikácie je IP adresa ignorovaná. K dátam sa pristupuje nasledovne: *meno_miestnej_sekcie.meno_vzdialenej_premennej*.

5.5.3.2 Zdieľanie prístrojov a procedúr

Tento spôsob tvorby modulárnych a distribuovaných aplikácií nevytvára medzi klientom a serverom pevnú väzbu. Prístroje na strane klienta nezobrazujú stav na strane servera, slúžia k aktivácii prístrojov serverovej aplikácie, k nastaveniu požadovaných dát alebo k volaniu procedúr jej prístrojov. Na klientove výstupné dáta server reaguje, nevykonáva však spätnú komunikáciu s klientom, čím nezaťažuje komunikačný systém. Voľby vykonané na strane servera sú nadradené voľbám klienta. Tento spôsob zdieľania je vhodné voliť v prípade, kedy klient zadáva požiadavky serveru a nie je požadovaná vizualizácia na strane klienta.

V serverovej aplikácii je nutné nastaviť parameter *scope* v záložke *Datové inspektory* v sekcii *Vzdialený prístup*. Nastavenie tohto parametra spolu s parametrom *client_access* je rovnaké ako v prípade zdieľania dátových sekcií.

Na strane klientskej aplikácie je nutné vytvoriť v záložke *Datové inspektory* nový modul s ľubovoľným názvom. Nastavenie parametrov *scope* a *client_access* je obdobné ako v predošlom prípade, pričom server je definovaný IP adresou a názvom serverovej aplikácie. Prístup k vzdialeným prístrojom je nasledovný:

meno_miestneho_modulu.meno_vzdialeného_prístroja;

Volanie procedúr vzdialených prístrojov je nasledovné:

meno_miestneho_modulu.meno_vzdialeného_prístroja.názov_procedúry;

Takýmto spôsobom je možné vytvárať rozsiahle distribuované a synchronizované aplikácie. Jednotlivé dátové sekcie a moduly môžu byť ľubovoľne prístupňované iným modulom. Klient môže mať definovaný väčší počet serverov, kedy pri výpadku jedného sa automaticky prihlási na ďalší možný. Klienti i servery môžu svoju činnosť kedykoľvek ukončiť, preto je dobré dbať na väčší počet serverových aplikácií, ktoré zabezpečia správny chod celého systému i pri čiastočnej poruche. V prípade lokálnych aplikácií sa pri spustení klienta automaticky spustí i jeho server.

6 UKÁŽKOVÉ APLIKÁCIE

Uvedené aplikácie znázorňujú spôsob realizácie komunikácie medzi PC a PLC, spôsob časovania virtuálnych prístrojov s dôrazom na minimálne zaťaženie procesora, využitie komunikačných kanálov, možnosti ovládačov, prácu s procedúrami a tiež napríklad prácu s rôznymi dátovými typmi. Uvedený je taktiež príklad tvorby distribuovaných aplikácií na základe komunikácie klient – server s ukážkou riadenia vybraných modelov.

6.1 Ukážková aplikácia komunikácie s PLC Tecomat

Komunikácia s PLC je realizovaná prostredníctvom sériovej linky RS-232. Aplikácia pozostáva z troch panelov, medzi ktorými sa môže užívateľ prepínať pomocou výberu príslušnej záložky v hornej časti okna alebo stlačením príslušnej klávesy (Alt + počítačové písmeno názvu záložky). Vlastníkom všetkých troch panelov je prístroj *hlavný_panel*, ktorý predstavuje hlavné okno aplikácie. Tento prístroj nevykonáva žiadnu operáciu, preto nie je časovaný. Parameter *gui - window - type* udáva typ okna. Hodnota 'normal' predstavuje klasické okno systému MS Windows s možnosťou voľby zobrazenia ikony a titulku. Obmedzenie maximalizovania a iných operácií s oknom zabezpečuje parameter *gui - window - disable*. Výhodou pri práci je uvedenie príkazu *system.StopApplication()*; v procedúre *OnWindowClose()*, ktorý zabezpečí ukončenie behu spustenej aplikácie zavretím okna aplikácie.

Hlavné nastavenia aplikácie sa nachádzajú v záložke *Datové inspektory*, v časti *Nastavení aplikace*. Parameter *time_error_limit* uvedený v sekcii *Časování* udáva čas v sekundách, kedy systém cyklicky upozorňuje užívateľa na sklz v aplikácii, čo znamená, že systém nedokáže dodržať všetky operácie v stanovených periódach aktivácií prístrojov. Samotná aplikácia obsahuje dva typy dátových elementov. Prvým je dátový element typu *var*, ktorý obsahuje jednu skalárnu premennú s názvom *ActivePanel*, vytvorenú sprievodcom novej aplikácie. Je využívaná pri určení viditeľnosti panelov a časovaní ich prístrojov. Druhý dátový element je typu *channel*, ktorý obsahuje všetky premenné odpovedajúce jednotlivým komunikačným kanálom uvedeným v mapovacom a parametrickom súbore. Všetky premenné sú skalárne. Parameter *timeout* tohto elementu predstavuje čas v milisekundách, vyhradený k uskutočneniu komunikácie. Nulová hodnota môže spôsobiť nesprávny výsledok komunikácie.

6.1.1 Panel VSTUPY / VÝSTUPY

V hlavnej časti panela sa nachádza osem prístrojov typu *indicator* zobrazujúcich stav vstupov PLC, pomenovaných *in_1* až *in_8*, ktoré vyhodnocujú rovnako nazývajúce sa premenné. Stav vstupov PLC sa môžu kedykoľvek zmeniť, preto ak požadujeme vždy aktuálnu hodnotu vstupu, je nutné tieto indikátory časovať. Časovanie je realizované pomocou časovača typu *selector*, vďaka ktorému sú všetky časované prístroje daného panela časované len v prípade viditeľnosti panela. Tým dochádza k zníženiu požiadaviek na prácu procesora a k zníženiu zaťaženia komunikačnej linky.

Prístroje *out_1* až *out_8* slúžia užívateľovi k nastaveniu úrovne jednotlivých výstupov PLC. Prístroje nie sú časované periodicky. Hodnota 'infinite' parametra *activity - period* zabezpečuje jednorazovú aktiváciu prístrojov pri štarte aplikácie. Dôvodom je požiadavka na súlad medzi skutočným stavom výstupov a zobrazením polohy prepínačov. Súčasne je vhodné nastaviť hodnotu parametra *startup_options - output_action* na 'init_self' a povoliť prístroju nastavenie podľa hodnoty svojho výstupného elementu pomocou parametra *auto_update*, hodnotou 'true'. Pri štarte prístroj prečíta hodnotu svojho výstupu uvedeného v parametri *output* a nastaví sa do odpovedajúcej polohy. Vzhľad prístroja v jednotlivých úrovniach je možné zmeniť vložení nových ikon pomocou parametrov *true_icon* a *false_icon*. Parameter *logic* stanovuje logiku správania prepínača. Hodnota 'set_flip_flop' predstavuje funkciu prepínača, 'set_true' a 'set_false' predstavujú jednorazovú voľbu bez možnosti návratu a hodnoty 'set_true_on_press' a 'set_false_on_press' predstavujú funkciu tlačidiel. Každý prepínač má v parametri *receivers* uvedený názov prístroja zobrazujúceho stav odpovedajúceho výstupu. Prístroje *ind_1* až *ind_8* preto nie je nutné časovať. Pri zmene stavu ľubovoľného výstupu užívateľom dôjde zároveň k aktivácii odpovedajúceho prístroja signalizácie. Vyhodnocovanými premennými nie sú rovnaké premenné ako výstupy prepínačov, i keď oboje odkazujú na rovnaké miesto v pamäti PLC. Tým je zabezpečená indikácia skutočného stavu výstupu PLC a nie indikácia voľby stavu.

Prístroj *výstup_binárne* slúži na hromadné nastavenie všetkých ôsmich výstupov. Zvolenú hodnotu od 0 do 255 zapisuje do miesta pamäte obrazov výstupov vo forme reálneho osembitového čísla dátového typu *uint8*.

6.1.1.1 Pripojenie a odpojenie stanice PLC

Menšiu časť panela VSTUPY / VÝSTUPY tvorí panel s názvom *dis_connect*, ktorý zobrazuje možnosť práce s pripojenými jednotkami v prípade komunikácie napríklad po zbernici RS-485, kedy môže dôjsť k odpojeniu poškodeného zariadenia a tým k nezaťažovaniu komunikačnej linky čakaním na odozvu komunikácie. Hodnota 'tool' parametra *gui - window - type* určuje typ okna s titulkom bez ikony. Manipulácia s oknom je zakázaná.

V pravej časti sa nachádza prístroj typu *multi_label* znázorňujúci stav komunikácie. Použitie prístroja *multi_label* je jeden z možných spôsobov v prípade potreby indikácie viacerých možných stavov textovým popisom. Prístroj je aktivovaný časovačom v prípade viditeľnosti. Jednotlivé stavy s podmienkou zobrazenia sú uvedené za parametrom *item*. Tento parameter ponúka celú tabuľku možných nastavení. Každý riadok tabuľky odpovedá jednému zobrazovanému stavu. Hodnota 'expression' uvedená v prvom stĺpci tabuľky znamená vyhodnocovanie výrazu. Výraz, ktorý prístroj vyhodnocuje spolu s podmienkou, je uvedený v druhom stĺpci tabuľky. Ak je podmienka splnená, prístroj zobrazuje text, ktorý je spolu s formátovaním uvedený vo štvrtom stĺpci tabuľky.

Prístroje *meno_p* a *meno_o* v ľavej časti panela sprístupňujú možnosť zadávania adresy pripájanej alebo odpojanej stanice. Pomocou príkazov *Show()*; uvedených v procedúre *OnMouseClicked()* u oboch prístrojov sú zobrazované prístroje *c_stanice_p* a *c_stanice_o*, do ktorých je adresa stanice zapisovaná. Tieto prístroje majú v procedúre *OnStartup()* uvedený príkaz *Hide()*; preto nie sú po spustení aplikácie viditeľné. Rovnaký príkaz je uvedený v procedúre *OnOutput()*, čím po zadaní a potvrdení adresy stanice dôjde ku skrytiu prístroja.

6.1.2 Panel REGISTRE

V tomto paneli môže užívateľ sledovať a meniť hodnotu dvoch registrov v pamäti PLC Tecomat, register R10 a register R11. Jednotlivé bity registrov sú zadávané jednotlivo, do pamäte sa preto prístupuje skrze osem jednotlivých kanálov. Dátový typ kanála, *uint8*, umožňuje zobrazenie číselnej hodnoty celého registra, údaj sa vyhodnocuje ako osembitové číslo bez znamienka. Indikátor vyhodnocujúci premennú typu *bool8* signalizuje stav *true* v prípade nenulovej hodnoty registra. Komunikačný kanál je opäť len jeden. Dátový typ

uint16 pristupuje ku dvojici registrov ako ku šestnástbitovému číslu bez znamienka. Pre názornosť grafického prevedenia je nutné nastaviť vzájomnú zámenu bajtov. V parametrickom súbore je u tohto kanála uvedená nepovinná hodnota 'SwapBytes', ktorá zaručí, aby bit s najnižšou váhou začínal v registri R11. Na zobrazenie numerických hodnôt registrov sú použité virtuálne prístroje typu *meter*. Tieto prístroje nie sú časované. Ku ich aktivácii a aktualizácii zobrazovanej hodnoty dochádza po zmene ktoréhokoľvek bitu registrov, vďaka uvedeniu ich názvu za parametrom *receivers* u všetkých prepínačov.

6.1.3 Panel STAV

Na paneli STAV sa nachádzajú dva farebne rozlíšené panely s názvami *p_stav* a *p_udaje*. Panel *p_stav* podáva textovú informáciu o stave samotného PLC a tiež o stave jeho výstupov. Na zobrazenie sú využité prístroje typu *multi_label* obdobným spôsobom ako na paneli VSTUPY / VÝSTUPY. Oba prístroje sú časované časovačom v prípade viditeľnosti.

Panel *p_udaje* podáva informácie o dni, dátume a čase. Tieto informácie PLC uchováva v registroch S. Časovačom je časovaný len prístroj *sekunda* zobrazujúci sekundy času. Ostatné prístroje nie je nutné časovať periódou časovača, preto sú časované periodicky v menších intervaloch. Výsledkom je menšie zaťaženie komunikačného systému.

6.1.4 Panel controls_panel

Panel obsahuje jediný prístroj – *tab_switch*. O nastavenie tohto prístroja spolu s vytvorením štruktúry viacerých navzájom prekrývajúcich sa panelov sa stará sprievodca novou aplikáciou. Tento umožňuje nastaviť časovanie prístrojov v závislosti od viditeľnosti panela s voľbou periódy časovačov typu *selector*. Medzi ďalšie možnosti patrí voľba rozmerov hlavného panela aplikácie, spôsob prepínania prekrývajúcich sa panelov a napríklad nastavenie detailov časovania.

6.1.5 Popis použitých komunikačných kanálov

- 100 – 107 Osem vstupných kanálov typu *boolean* predstavujúcich osem vstupov PLC začínajúcich adresou X0.0. Viazu sa na premenné *in_1* až *in_8*, vyhodnocované prístrojmi *in_1* až *in_8* na paneli VSTUPY / VÝSTUPY.

- 110 – 117 Osem obojsmerných kanálov typu *boolean* predstavujúcich osem výstupov PLC začínajúcich adresou Y0.0. Viazu sa na premenné *out_1* až *out_8*, vyhodnocované prístrojmi *out_1* až *out_8* na paneli VSTUPY / VÝSTUPY.
- 120 – 127 Osem vstupných kanálov typu *boolean* slúžiacich na zobrazenie stavu výstupov PLC. Predstavujú ich premenné *inout_1* až *inout_8*, sú vyhodnocované prístrojmi *ind_1* až *ind_8*.
- 140 Jeden obojsmerný kanál typu *uint8* slúžiaci na zápis výstupov pomocou reálneho čísla v rozmedzí od 0 do 255.
- 141 – 148 Využitie sú dva vstupné kanály 143 a 146. Poskytujú informáciu o stave výstupov PLC a o stave komunikácie z registrov S2.2 a S2.5. Odpovedajú im premenné *stav_PLC* a *stav_vystupov* vyhodnocované prístrojmi *PLC_stav* a *PLC_v_stav*.
- 150 – 156 Sedem vstupných kanálov začínajúcich v registri S6. Poskytujú informácie o aktuálnom dni, dátume a čase.
- 160 – 175 Šestnásť výstupných kanálov prístupujúcich k jednotlivým bitom registrov R10 a R11, začínajúc bitom R10.0. Odpovedajúcimi premennými sú *R10_0* až *R11_7*, vyhodnocovanými prístrojmi *r10_0* až *r11_7* na paneli s názvom REGISTRE.
- 180 – 181 Dva vstupné kanály dátového typu *bool8* prenášajúce hodnotu typu *boolean* o stave registrov R10 a R11. Premenné *Bool_R10* a *Bool_R11* sú vyhodnocované prístrojmi *bool_R10* a *bool_R11*.
- 190 – 191 Dvojica vstupných kanálov prístupujúcich ku registrom R10 a R11 prenášajúcich informáciu o stave registrov v podobe reálneho čísla. Premenné *Uint_R10* a *Uint_R11* sú vyhodnocované prístrojmi *uint_r10* a *uint_r11*.
- 200 Jeden vstupný kanál prístupujúci ku šestnásťbitovej hodnote spojených registrov R10 a R11. Bajty registrov sú navzájom vymenené. Táto hodnota premennej *R10_R11* je vyhodnocovaná prístrojom *r10_r11*.
- 1000 Jeden vstupný kanál získavajúci hodnoty odpovedajúce stavu komunikácie.

6.2 Ukázková aplikácia komunikácie s PLC SAIA

Základné rozloženie aplikácie bolo vygenerované pomocou sprievodcu novou aplikáciou. Ide o aplikáciu reálneho času, skladajúcu sa z troch prekrývajúcich sa panelov umiestnených na hlavnom paneli. Medzi panelmi má užívateľ možnosť prepínať pomocou záložiek v hornej časti alebo pomocou kláves Alt + začiatkové písmeno názvu záložky. V aplikácii je vytvorená štruktúra časovačov s periódou časovania dvoch desiatin sekundy, čo je dostatočne krátky čas na zobrazenie aktuálneho indikovaného stavu. Každý panel má pridelený svoj vlastný časovač, preto nedochádza k časovaniu prístrojov neaktívneho panela.

Medzi možnosťami aplikácie patrí vizualizácia vstupov a výstupov s možnosťou nastavenia stavu výstupov, objasnenie práce s pamäťou PLC a príklad využitia vnútorných informácií registrov PLC. Na disku CD sa aplikácia nachádza v troch rôznych variantoch. Každý sa líši použitým parametrickým súborom, navrhnutým samostatne pre komunikáciu po sériovej linke RS-232, USB alebo skrze sieť Ethernet.

Všeobecné nastavenia aplikácie sa nachádzajú v záložke *Datové inspektory*. Parameter *time_error_limit* v sekcii *Časování* je nastavený na hodnotu '0', čo znamená, že aplikácia upozorní na prácu v časovom sklze hneď pri výskyte sklzu. Do sekcii *Ovladače* bol zavedený ovládač SAIA PCD Driver a napojený mapovací a parametrický súbor s definíciou komunikačných kanálov. V sekcii *Datové elementy* sa nachádzajú dve skupiny elementov. Prvá, typu *var*, obsahuje jednu skalárnu premennú s názvom *ActivePanel*, ktorá je využívaná pri rozpoznávaní aktívneho panela a pri riadení práce selektívnych časovačov. Druhá skupina je typu *channel*. Hodnota jeho parametra *driver* definuje ovládač, na ktorého kanály sú skalárne premenné v sekcii napojené. Parameter *timeout* vyhradzuje čas 200 milisekúnd na uskutočnenie komunikácie.

6.2.1 Hlavný panel aplikácie

Hlavným panelom aplikácie je panel s názvom *hl_panel*. Tvorí hlavné okno aplikácie. Nie je časovaný, pretože nevykonáva žiadnu operáciu závislú na čase. Parameter *gui - window - type* definuje hodnotou 'normal' typ klasického okna s uvedením titulku za parametrom *title* a zobrazovanú ikonu v ľavom hornom rohu definovanú za parametrom *icon*. Parameter *disable* umožňuje zakázať niektoré operácie s oknom. Vhodnou voľbou je

zákaz zväčšovania a maximalizovania okna. Procedúra *OnStartup()* obsahuje správanie panela po spustení aplikácie. Je vygenerovaná sprievodcom novou aplikáciou. Príkaz *system.StopApplication()*; v procedúre *OnWindowClose()* umožňuje ukončenie behu aplikácie zatvorením hlavného okna aplikácie.

6.2.2 Panel Ovládanie

Na paneli sa nachádzajú prístroje typu *indicator* reprezentujúce stavy vstupov a výstupov PLC. Prístroje *in_1* až *in_8* vyhodnocujú premenné *Vst_1* až *Vst_8*. Hodnota 'true' parametra *transparent* definuje objekty bez pozadia. O zobrazenie aktuálneho stavu vstupov sa stará časovač typu *selector*. Prístroje *out_1* až *out_8*, vyhodnocujúce premenné *Out_01* až *Out_08* zobrazujú aktuálny stav výstupov PLC. Nie sú časované, ku ich aktivácii dôjde v dôsledku zmeny stavu výstupu užívateľom.

Na označenie jednotlivých prístrojov a skupín prístrojov slúžia prístroje typu *label*. Tieto popisy nie sú časované, sú len zobrazované, nevykonávajú žiadnu operáciu. Zobrazovaný text je definovaný v skupine parametrov *text_list*. Parameter *font* tejto skupiny definuje typ, veľkosť a farbu písma. Parameter *text* obsahuje samotný zobrazovaný text. Parameter *colors* je spoločným parametrom väčšiny prístrojov, umožňuje definovať farby zobrazovaného prístroja v rôznych situáciách.

Prístroje typu *switch* s názvami *set_1* až *set_8* slúžia užívateľovi na vykonanie zmeny stavov výstupov PLC. Hodnota 'infinite' parametra *activity - period* spôsobí jednorazovú aktiváciu prístrojov po spustení aplikácie. Táto aktivácia zabezpečuje nastavenie prepínačov podľa stavu svojich výstupov. Súčasne je vhodné uviesť hodnotu 'init_self' parametra *startup_options - output_action* a je nutné uviesť hodnotu 'true' parametra *auto_update*. Hodnota 'true' parametra *startup_options - activate_receivers* povolí aktivovanie prístrojov uvedených za parametrom *receivers* počas prvej aktivácie prístroja po spustení aplikácie. Tým je zabezpečená indikácia stavu príslušného výstupu hneď po spustení aplikácie. Každý z prístrojov *set_1* až *set_8* má za parametrom *receivers* uvedený odpovedajúci prístroj *out_1* až *out_8*, indikujúci stav príslušného výstupu. Parameter *output* u prepínačov definuje premennú, do ktorej prístroj zapisuje svoj stav. Typ prevedenia prístrojov *switch* definuje parameter *mode*. Hodnota parametra *init_value* definuje počiatočný stav prístroja po spustení aplikácie v prípade, ak nezískava spätné hodnotu svojho výstupu. Parametre *true_icon* a *false_icon* definujú ikony stavov a parameter *logic* logiku správania prístroja.

6.2.3 Panel Registre

Panel sa skladá z dvoch častí. Hornú časť predstavuje prístroj *reg* a *reg_in*. Prístroj *reg* slúži na zápis hodnoty do registra. Prístroj *reg_in* túto hodnotu zobrazuje. Prístroje pracujú s rôznymi premennými, pričom obe odkazujú na rovnaké miesto v pamäti PLC. Komunikácia s PLC SAIA prebieha po blokoch o veľkosti 32 bitov. Vytvoria sa štyri komunikačné kanály po ôsmich bitoch, z ktorých jeden slúži na komunikáciu zápisu do registra. Prístroj zobrazujúci hodnotu registra nie je časovaný periodicky, ale jeho názov je uvedený za parametrom *receivers* prístroja *reg*. K jeho aktivácii dochádza vždy pri zmene hodnoty registra užívateľom. Na činnosti týchto dvoch prístrojov je možné sledovať rýchlosť odozvy rôznych spôsobov komunikácie. Pri komunikácii pomocou sériovej linky RS-232 je rýchlosť nepostačujúca na uskutočnenie komunikácie v čase aktivovania, preto prístroj *reg_in* nezobrazuje vždy aktuálnu hodnotu nastavenú prístrojom *reg*. Nevykoná sa totiž komunikácia po poslednom aktivovaní prístrojom *reg*. U ostatných komunikačných liniek je možné pozorovať vyššiu rýchlosť odozvy.

Dolná časť panela názorne poukazuje na rôzne spôsoby prístupu do pamäte PLC. Ľavá časť slúži pre nastavenie hodnoty registrov, pravá časť na ich zobrazenie. Prístup do pamäte je realizovaný buď pomocou jedného alebo ôsmich kanálov. Takto možno doceliť zobrazovanie hodnôt binárne, i vo forme reálneho čísla.

Všetky prístroje slúžiace k nastaveniu hodnoty príslušného registra sú aktivované jednorázovo po spustení aplikácie. Hodnota 'true' parametra *auto_update* zaručí nastavenie prístroja podľa hodnoty registra. Názvy prístrojov zobrazujúcich hodnoty registrov sú uvedené za parametrom *receivers* prístrojov do registrov zapisujúcich. V čase, kedy nedochádza ku zmene hodnôt registrov nedochádza k zbytočnému časovaniu prístrojov.

6.2.4 Panel Stav CPU

V hornej časti panela sa nachádza prístroj typu *multi_label*, ktorý vyhodnocuje hodnotu premennej *Stav*, podľa ktorej informuje užívateľa o aktuálnom stave PLC. Prístroj je časovaný len v prípade viditeľnosti panela, preto je po prvom spustení aplikácie možné nakrátko zahliadnuť stav PCD OFFLINE, než dôjde v ďalšom kroku komunikácie k aktualizovaniu stavu.

Dolná časť panela ponúka informácie napríklad o dátume a čase.

6.2.5 Popis použitých komunikačných kanálov

- 101 – 108 Osem vstupných komunikačných kanálov slúžiacich na zobrazenie stavu vstupov. Odpovedajú im premenné *Vst_1* až *Vst_8*.
- 109 – 116 Osem vstupných kanálov, premenné *Out_01* až *Out_08* zobrazujú aktuálne stavy výstupov.
- 120 – 127 Osem obojsmerných kanálov slúžiacich k zápisu stavov výstupov a aktualizácii hodnôt po spustení aplikácie. Naviazanými premennými sú premenné *Vys_1* až *Vys_8*.
- 128 – 131 Štyri obojsmerné kanály po ôsmich bitoch, využívaný je prvý kanál pre prácu s registrom R 22, pomocou premennej *Register*.
- 132 – 135 Štyri vstupné kanály, z ktorých prvý slúži k zobrazeniu hodnoty registra R 22 pomocou premennej *Reg_in*.
- 136 Vstupný kanál slúžiaci k zobrazeniu hodnoty premennej *Stav*.
- 137 – 144 Vstupné kanály na prenos informácií o čase, dátume, dni a týždni.
- 145 Obojsmerný kanál slúžiaci k nastaveniu hodnoty registra R 44. Odpovedá premennej *k_8*.
- 146 – 177 32 jednotlivých vstupných kanálov. Využitých je prvých osem, slúžiacich k zobrazeniu stavov jednotlivých bitov registra R 44. Odpovedajú im premenné *v_1* až *v_8*.
- 180 – 211 32 obojsmerných kanálov prístupujúcich k jednotlivým bitom v pamäti. Využitých je prvých osem bitov registra R 33, s názvom premenných *k_1_1* až *k_1_8*.
- 200 – 223 Štyri vstupné kanály prístupujúce k 32 bitom v pamäti PLC. Prvých osem bitov zobrazuje hodnotu registra R 33.

6.3 Ukážka distribuovanej aplikácie

Vytvorená ukážka pozostáva zo serverovej aplikácie *modely_server* a klientskej časti s názvom *modely_klient* a *výtah_klient*. Funkciou aplikácie je regulácia modelu fontány a riadenie modelu výťahu. Aplikácia poukazuje na spôsoby vzdialeného zdieľania dát a prístrojov a na možnosť využitia systému Control Web 6 ako riadiaceho prvku bez nutnosti ďalšieho programového vybavenia.

Serverová aplikácia obsahuje tri časti. Prvú časť tvorí vizualizácia funkcie modelu výťahu. Riadenie výťahu v tomto prípade nie je funkciou aplikácie, ale stará sa oň samostatný program v PLC SAIA. Časť komunikovaných signálov je zdieľaná a slúži klientskej aplikácii *výtah_klient* na ovládanie modelu.

Druhú časť tvorí prevedenie, v ktorom sa o riadenie modelu výťahu stará samotný systém Control Web 6. Počet komunikovaných kanálov je vďaka tomu minimálny, komunikované sú len hodnoty snímačov na jednotlivých poschodiach a hodnoty slúžiace k pohybu kabíny. Aplikácia klienta prístupuje k jednotlivým zdieľaným prístrojom serverovej aplikácie. Komunikácia prebieha jednosmerne a prispieva tým k minimalizácii zaťaženia komunikačného systému.

Tretou časťou je regulácia úrovne loptičky vznášajúcej sa na vodnom stĺpci. Užívateľ má možnosť nastaviť požadovanú úroveň loptičky a sledovať priebeh regulácie. Klient prístupuje k prístroju voľby úrovne v serverovej aplikácii a môže meniť výstupnú hodnotu tohto prístroja.

Oba modely sú napojené na PLC SAIA a pripojené do siete Ethernet. Každé PLC má svoju vlastnú IP adresu. Postupy konfigurácií PLC sú uvedené v prílohe P I a P II. Serverová aplikácia môže byť spustená na ktorejkoľvek stanici pripojenej k danej sieti Ethernet. Taktiež aplikácie klientov môžu byť spúšťané na ktorýchkoľvek staniach pripojených do rovnakej siete Ethernet. Klienti však musia mať presne definované, na aký server sa majú pripájať. Klient sa nepripojí v prípade, ak nie je spustená serverová aplikácia a ak mu to brána firewall neumožní. Preto je dobré nastaviť na strane servera výnimky brány firewall alebo túto ochranu vypnúť.

Celá serverová aplikácia je zdieľaným modulom v sieti, čo umožňuje prístup nezávislých klientov k jeho prístrojom a procedúram. Počet klientov je možné obmedziť.

6.3.1 Aplikácia modely_server

Serverová aplikácia ponúka operátorovi všetky možnosti riadenia a plnej vizualizácie na rozdiel od aplikácií klientov s obmedzenými možnosťami. V aplikácii je zavedené väčšie množstvo ovládačov, kanálov a vyznačuje sa niektorými špecifickými nastaveniami. Aplikáciu tvoria tri hlavné panely, z ktorých prvý predstavuje vizualizáciu práce modelu výťahu za pomoci riadiaceho programu vytvoreného v programovacom prostredí PG5. Druhý panel prezentuje riadenie modelu systémom Control Web 6 a tretí panel znázorňuje prácu s modelom fontány.

V nastaveniach aplikácie, v sekcii *Časování záložky Datové inspektory* sa nachádza parameter *independent_procedure_execution*, ktorý nadobúda hodnotu 'true'. Plní významnú úlohu pri práci s procedúrami. Umožňuje používanie príkazov *pause*;, *yield*; a *wait*;, ktoré sú štandardne prístupné len procedúram *OnActivate()*, i v ostatných užívateľských procedúrach.

Sekcia *Vzdálený přístup* slúži k nastaveniu zdieľania modulu, teda aplikácie. Parameter *scope* rozlišuje medzi lokálnym a vzdialeným zdieľaním. Zdieľanie v sieti zaručí hodnota 'shared_remotely'. Parameter *client_access* umožňuje nastavenie rôznych obmedzení pre prístup klientov. Uvedená hodnota 'allow' nestanovuje obmedzenia prístupu, ale umožňuje prístup všetkým klientom. Ak chceme obmedziť množinu klientov na danú sieť, uvedieme za touto hodnotou masku podsiete, v našom prípade /255.255.254.0.

V aplikácii sú zavedené tri ovládače pre dve rôzne PLC SAIA. Každý z ovládačov má presne definovaný svoj mapovací a parametrický súbor. Tieto tri ovládače využívajú následne definované skupiny kanálov.

Datové elementy typu *var* obsahujú skupinu skalárnych premenných, využívaných vo vnútri aplikácie, týkajúcich sa modelu fontány a výťahu riadeného systémom Control Web 6. Významy premenných sú uvedené pri popisoch funkcií jednotlivých častí aplikácie.

Dátový typ *channel* s názvom *vytah_server* obsahuje skalárne premenné využívané výhradne serverovou aplikáciou bez možnosti zdieľania. Serverová aplikácia ďalej využíva i kanály určené pre aplikácie klientov, uvedené pod názvom *vytah_klient*. Táto skupina je na rozdiel od predošlej zdieľaná vzdialene, čo definuje hodnota 'shared_remotely' parametra *scope* tejto sekcii. Parameter *client_access* definuje obmedzenia pre prístup

klientov k tejto sekcii. Obe skupiny kanálov využívajú pre komunikáciu s modelom výťahu ovládač *Vytah*. Názov ovládača je uvedený ako hodnota parametra *driver* u oboch sekcii.

Pod názvom *fontana* sa nachádzajú dva definované kanály pre prístup k modelu fontány. Model poskytuje informáciu zo senzora o úrovni loptičky a ovládaný je výkon čerpadla. Parameter *driver* definuje použitie ovládača s názvom *Fontana*.

Na reguláciu úrovne loptičky nepostačuje samotná hodnota dodávaná zo senzora modelu. Preto je v aplikácii zavedený ďalší dátový element, *expression*, s názvom *p_fontana*. Obsahuje tri premenné, ktorých hodnoty sú získavané prepočtom z aktuálnych hodnôt iných premenných. Patrí medzi ne výpočet regulačnej odchýlky, regulovanej veličiny a vyjadrenie výkonu čerpadla v percentách.

Kanály spadajúce do skupiny pod názvom *vytah_cw* sú využívané v prípade riadenia modelu výťahu systémom Control Web 6 bez nutnosti zavedenia ďalšieho riadiaceho programu. Viazu sa na ovládač *Vytah_CW*. Čas čakania systému na ukončenie komunikácie je predĺžený o čas definovaný za parametrom *timeout*.

6.3.1.1 Panel Výťah

Časť aplikácie slúžiaca na vizualizáciu modelu výťahu. Hlavný riadiaci program sa nachádza v PLC, systém Control Web 6 sa na riadení nepodieľa, preto sa dokumentácia nezaobrá jeho popisom. Panel využíva kanály sekcii *vytah_server* a *vytah_klient*. Pre funkčnosť programu je nutné aby premenná *podm* nadobúdala hodnotu 'false'. Stará sa o to neviditeľný prístroj typu *switch* s názvom *podmienka*.

V ľavej časti panela sa nachádza vyobrazenie kabíny výťahu. Obsahuje tri tlačidlá voľby poschodia a prístroje indikujúce polohu kabíny. Správanie prístrojov typu *switch* ako tlačidiel definuje hodnota 'set_true_on_press' parametra *logic*. Prístroje zobrazujúce stavy premenných sú časované časovačom len v prípade viditeľnosti panela. Umiestnenie zobrazovaných údajov prístroja *multi_label* definuje parameter *justify*. Umožňuje nastaviť zarovnanie vľavo, vpravo alebo centrovanie textu.

Pravá časť panela predstavuje jednotlivé poschodia s tlačidlami privolania výťahu a prístrojmi *smer_1*, *smer_2* a *smer_3* zobrazujúcimi nasledovnú zastávku kabíny.

Dolná časť panela reprezentuje servisné tlačidlá ako zastavenie výťahu, nastavenie kabíny na úroveň prvého poschodia a tlačidlo opätovného uvedenia výťahu do prevádzky.

6.3.1.2 Panel *Výtah_CW*

Panel predstavuje ukážku riadenia modelu pomocou systému Control Web 6. Obsahuje skryté prístroje zabezpečujúce samotné riadenie modelu a prístroje slúžiace na zobrazenie hodnôt niektorých veličín. Prvou fázou návrhu aplikácie bolo vytvorenie simulačného modelu výťahu, ktorý bol v druhej časti návrhu napojený na reálny model.

V ľavej časti panela sa nachádzajú indikátory zobrazujúce reálne hodnoty snímačov prítomnosti kabíny na jednotlivých poschodiach. Tieto prístroje, *real_1*, *real_2* a *real_3* sa nijak nezúčastňujú na riadení modelu. Prístroj *prpravuje* slúži na upozornenie užívateľa počas inicializácie a vyhodnotenia stavu výťahu po spustení aplikácie. Po vyhodnotení nasledujúcich krokov sa výstražný symbol deaktivuje. Prístroje *obr_1* až *obr_3* predstavujú obrazy reálnych snímačov. V aplikácii je možné sledovať čas od zmeny hodnoty reálneho snímača po vyhodnotenie tohto stavu s následnou zmenou odpovedajúcich premenných. Hodnoty premenných *snimac_1p* až *snimac_3p* sa podieľajú na riadení modelu. Prístroje typu *indicator* sú časované len v prípade viditeľnosti panela *Výtah_CW*. Rovnako tiež prístroj *ch_mt*, ktorý indikuje chod motora výťahu.

Zvyšná časť panela predstavuje tri poschodia. Na každom poschodí sa nachádza tlačidlo privolania kabíny výťahu. Samotná kabína obsahuje tri tlačidlá pre voľbu ďalšieho pohybu kabíny. Časť prístrojov typu *panel* predstavuje steny príslušnej trojposchodovej budovy a steny kabíny. Časť panelov tvorí dvere kabíny. Tieto prístroje neplnia žiadnu úlohu v procese riadenia, slúžia pre názornú ukážku. Hierarchia vlastníkov zabezpečuje správnu viditeľnosť a správne prekryvanie panelov. Panel *kabinka* tvorí vizuálnu podobu samotnej kabíny. Je časovaný periodicky, nielen počas viditeľnosti panela *Vytah_CW*. Jeho vlastníkom je panel *pozadie_poschodi*. Všetky prístroje, ktorých vlastníkom je panel *kabinka* menia svoju polohu podľa zmeny polohy jeho samého. Tento panel má definované dve procedúry. V procedúre *OnStartup()* sa vykoná skrytie tohto panela. O jeho opätovné zobrazenie sa stará prístroj *ready*, ktorý okrem iného informuje užívateľa o pripravenosti výťahu k použitiu. V procedúre *OnActivate()* je definované správanie panela v čase pohybu kabíny. V prípade, že je motor v činnosti, vykoná sa príkaz *dvere.Show()*; Tým sa zobrazí panel predstavujúci dvere kabíny. Následne sa podľa hodnoty premennej *m_smer* vyhodnotí smer pohybu. Hodnota *true* predstavuje pohyb v smere nahor, hodnota *false* pohyb v smere nadol. Príkaz *Move(x,y)*; udáva, o koľko sa prístroj pri svojej aktivácii posunie v smere príslušnej osi. Ak motor nie je v činnosti, dvere sa skryjú a následne sa vyhodnocujú

hodnoty obrazov reálnych snímačov. Poloha kabíny sa zároveň pomocou príkazu *MoveTo(x,y)*; nastaví do odpovedajúcej polohy príslušného poschodia.

Neviditeľný prístroj typu *switch* s názvom *priprav* sa stará o pripravenie výťahu na použitie. Časovaný je jednorázovo po štarte aplikácie, o čo sa stará hodnota 'infinite' parametra *activity - period*. Nie je prístupný užívateľovi. Po spustení aplikácie čaká tri sekundy na nadviazanie komunikácie a zistenie skutočných hodnôt snímačov modelu. Ak sa kabína výťahu nachádza v pozícii na prvom poschodí, prístroj nevykoná žiadnu akciu. Ak kabína nie je na prvom poschodí, teda hodnota premennej *snimac_1p* je *false*, zakáže sa obsluha prístroja *snimanie_2* príkazom *Disable()*; Tento prístroj nie je užívateľovi prístupný, preto je tento riadok príkazu z pozície funkcie nepodstatný. Následne sa nastaví premenná *m_smer* na hodnotu *false*, teda kabína sa bude pohybovať smerom nadol. Táto premenná predstavuje simulačný smer pohybu. Skutočné nastavenie smeru zabezpečuje príkaz *ovladanie_smeru_motora.SetValue(false)*; Výstupnou hodnotou prístroja *ovladanie_smeru_motora* je premenná *smer_motora*, ktorá je priamo komunikovaná s modelom. Vo vnútri aplikácie je motor uvedený do pohybu pomocou premennej *motor*, nastavenej na hodnotu *true*. Príkaz *ovladanie_pohybu_motora.SetValue(true)*; nastaví prepínač riadiaci pohyb reálneho motora do polohy *true*, čím uvedie skutočný motor do pohybu. Následne čaká, kým sa kabína výťahu nedostaví do prvého poschodia. Po dosiahnutí tejto úrovne sa simulovaný i reálny motor vypnú. Tým končí úloha prístroja *priprav*, ktorý už viac nie je aktivovaný.

Prístroje *ovladanie_smeru_motora* a *ovladanie_pohybu_motora* sprostredkovávajú ovládanie reálneho modelu. Ich výstupné hodnoty, *smer_motora* a *pohyb_motora*, sú prepojené priamo s výstupmi PLC a riadia pohyb reálneho motora modelu.

Prístroje *snimanie_1*, *snimanie_2* a *snimanie_3* predstavujú simuláciu snímačov kabíny na jednotlivých poschodiach. Ich výstupné hodnoty *snimac_1p*, *snimac_2p* a *snimac_3p* ovplyvňujú rozhodnutia iných prístrojov o nasledovnom pohybe kabíny. Prístroje nie sú periodicky časované. K ich aktivácii dochádza v prípade zápisu do ich výstupnej hodnoty. V prípade, že kabína dorazí do polohy, kedy aktivuje jeden zo snímačov polohy, prístroje *real_snimacX* zabezpečia zmenu výstupnej hodnoty jedného z týchto prístrojov a vyvolajú prevedenie procedúry *OnOutput()* odpovedajúceho prístroja. Príslušná procedúra sa vykoná len v prípade prechodu snímača z polohy *false* do polohy *true*, teda len v prípade príchodu kabíny na dané poschodie, nie pri jej odchode z daného poschodia.

Zabezpečuje to podmienka *if snimac_Xp=false*. Následne prístroj čaká na uskutočnenie zápisu výstupnej premennej a volá procedúru *ProcedureStartX()* prístroja *pristroj_X*. Tieto prístroje, *snimanie_1*, *snimanie_2* a *snimanie_3*, nie sú užívateľovi k dispozícii. O ich prepínanie sa starajú prístroje *real_snimac1*, *real_snimac2* a *real_snimac3*.

Prístroje *pristroj_1*, *pristroj_2* a *pristroj_3* rozhodujú o nasledovnom pohybe kabíny po príchode na prvé, druhé alebo tretie poschodie. Prístroje nie sú časované a neplnia svoju funkciu indikátorov. Každý z týchto prístrojov obsahuje užívateľskú procedúru *ProcedureStartX()* a *ProcedurePristrojX()*. Po príchode na prvé poschodie sa vykoná procedúra *ProcedureStart1()*. V úvode tejto procedúry systém počká krátky čas na dokončenie požadovaných zápisov z predošlého kroku komunikácie. Následne nastaví premennú *m_smer* na hodnotu *true*, čo znamená, že ďalší pohyb kabíny bude smerom nahor. Rovnako túto hodnotu nastaví priamo na výstupe PLC. Pohyb motora naopak nastaví na hodnotu *false*. Vďaka tomu dôjde po príchode kabíny na prvé poschodie k jej zastaveniu. Premenné *go_to_1* a *pojdes_do1* sú po príchode na prvé poschodie nastavené na hodnotu *false*. Hodnota *true* týchto premenných určuje ďalší pohyb kabíny na prvé poschodie. Vo vykonávaní procedúry sa uskutoční krátke prerušenie na dokončenie komunikácií a následne sa volá procedúra *ProcedurePristroj1()* toho istého prístroja. Na začiatku tejto procedúry sa opäť čaká krátky čas na dokončenie predošlých komunikácií a dochádza k samotnému rozhodnutiu o nasledujúcom pohybe kabíny. Ak premenná *go_to_2*, *go_to_2up*, *go_to_2down* alebo *go_to_3* nadobúda hodnotu *true*, motor sa uvedie do pohybu. Ak žiadna z týchto premenných nenadobúda hodnotu *true*, kabína čaká na prvom poschodí.

Po príchode kabíny na druhé poschodie prístroj *snimanie_2* volá procedúru *ProcedureStart2()* prístroja *pristroj_2*. V úvode procedúry príkaz *pause 0*; zabezpečí krátky čas na dokončenie predošlej komunikácie a zápisy zmien premenných. Následne, ak nadobúdala premenná *go_to_2* hodnotu *true*, dôjde k zastaveniu motora výťahu a k zápisu hodnoty *false* do tejto premennej. Zároveň sa vymedzí krátky čas na dokončenie zápisov zmien všetkých premenných a na ukončenie komunikácií. V ďalšej časti sa testuje hodnota premennej *pojdes_do1*. Ak nadobúda hodnotu *true*, to znamená, že výťah má pokračovať do prvého poschodia, testuje sa hodnota premennej *go_to_2down*. V prípade, že jej hodnota je *true*, dôjde k zastaveniu motora výťahu, k nastaveniu hodnoty *false* tejto premennej a výťah čaká tri sekundy na nástup a výstup osôb. Ak *go_to_2down* nenadobúda hodnotu

true, nedôjde k pozastaveniu výťahu. Dôjde k zápisu hodnoty *false* do premennej *m_smer* pre pohyb nadol a rovnako tiež k zápisu na výstupy PLC. Podobne dôjde k uvedeniu motora do chodu. Ak premenná *pojdes_do1* nenadobúda hodnotu *true*, testuje sa hodnota premennej *pojdes_do3*. Ak je jej hodnota *true* a hodnota premennej *go_to_2up* nadobúda taktiež hodnotu *true*, dôjde k pozastaveniu kabíny na druhom poschodí na uskutočnenie nástupu a výstupu osôb. Premenná *m_smer* sa pre pohyb smerom nahor nastaví na hodnotu *true*. Rovnako dôjde k spusteniu motora zápisom do premennej *motor* a priamo na výstupy PLC príkazom *ovladanie_pohybu_motora.SetValue(true);*. Ďalšími vyhodnocovanými premennými sú *go_to_1* a *go_to_3*. U týchto premenných sa vykonáva rovnaký postup ako u premenných *pojdes_do1* a *pojdes_do3*. Premenné *pojdes_do1* a *pojdes_do3* sú však vo vyhodnocovaní uprednostnené. Ak žiadna z týchto premenných nenadobúda hodnotu *true*, dôjde k zastaveniu motora a k zápisu hodnoty *false* do premenných *go_to_2up* a *go_to_2down*.

Procedúra *ProcedurePristroj2()* sa vykoná v prípade, ak kabína čaká v druhom poschodí a dôjde k stlačeniu jedného z tlačidiel privolania alebo k stlačeniu voľby v kabíne výťahu. Procedúra vyhodnocuje premenné *go_to_1* a *go_to_3*, podľa čoho sa rozhoduje o tom, ktorým smerom sa kabína vydá. Podobne ako u prístroja *pristroj_1* dochádza k nastaveniu smeru pohybu a ku spúšťaniu, prípadne zastaveniu motora.

Prístroj *pristroj_3* sa stará o rozhodovanie v prípade, ak sa kabína nachádza v treťom poschodí. Obdobne ako *pristroj_1* a *pristroj_2*, i *pristroj_3* obsahuje dve procedúry, *ProcedureStart3()* a *ProcedurePristroj3()*. Dochádza k rovnakému rozhodovaniu ako u prístroja *pristroj_1* s voľbami opačného smeru pohybu kabíny.

Procedúry *ProcedureStartX()* sa vykonávajú v momente, keď kabína na dané poschodie dorazí. Riadia zastavenie kabíny a vyhodnotenie nasledujúceho smeru kabíny. Následne procedúry *ProcedurePristrojX()* vyhodnocujú ďalší pohyb kabíny, podľa volieb tlačidiel na chodbách a v kabíne výťahu.

Prístroje *real_snimacX* sú aktivované jednorázovo po spustení aplikácie, ich procedúra sa však vykonáva neustále cyklicky vďaka príkazu *loop*. Vyhodnocovaným výrazom je hodnota premennej *poschodie_X*, čo je aktuálna hodnota snímačov modelu. Prístroje zabezpečujú správne prepínanie prístrojov *snimanie_1*, *snimanie_2* a *snimanie_3*, ktoré predstavujú obrazy reálnych snímačov. V prípade, ak kabína dorazí na určité poschodie

a snímač vyhodnotí jej prítomnosť, dôjde k prepnutiu jedného z prístrojov *snimanie_X*. To, do akej polohy bude prepínač prepnutý záleží od stavu jeho výstupnej premennej. K prepnutiu dôjde len raz. Tým je rozlíšené správanie v prípade príchodu kabíny na dané poschodie, od jej odchodu. Medzi každým cyklom sa vykoná krátke čakanie, aby nedošlo k zahltenu procesora.

Prístroje *chodba_1*, *chodba_2up*, *chodba_2down* a *chodba_3* predstavujú tlačidlá privolania výťahu na jednotlivých poschodiach. Tieto prístroje obsahujú rovnakú procedúru *OnOutput()*, v každej je však vyhodnocovaná iná výstupná premenná príslušného prístroja. Prvá podmienka v týchto procedúrach zabezpečuje vykonanie procedúry len v prípade stlačenia tlačidla, teda pri prechode z úrovne *false* do úrovne *true*. Pri uvoľnení tlačidla sa nevykonáva žiadna akcia. Prístroje vyhodnocujú stavy snímačov polohy kabíny. Ak sa kabína nachádza na nejakom poschodí, volá sa procedúra *ProcedurePristrojX()* príslušného prístroja. Dôjde tým k vyhodnoteniu ďalšieho pohybu kabíny. Ak sa kabína nenachádza na žiadnom poschodí, teda ak je niekde medzi poschodiami, príkazom *send medzi*; dôjde k aktivovaniu prístroja *medzi*. Prístroje *chodba_1* a *chodba_3* plnia rovnakú úlohu ako prístroje *tlacidlo_1_kabina* a *tlacidlo_3_kabina*. Preto sú tieto odpovedajúce dvojice na seba naviazané uvedením názvu odpovedajúceho prístroja za parametrom *receivers*. Hodnota 'true' parametra *auto_update* zabezpečuje rovnaký vzhľad oboch prístrojov pri aktivácii jedného z nich. Parameter *logic* nadobúda hodnotu 'set_true'.

Prístroj *medzi* slúži k vyhodnoteniu pohybu kabíny počas stlačenia volieb výťahu v prípade, ak sa kabína nachádza niekde medzi poschodiami. Ak dôjde k stlačeniu voľby príchodu do prvého poschodia, teda ak *go_to_1* nadobúda hodnotu *true*, zisťuje sa, či už nebola predtým stlačená voľba príchodu do tretieho poschodia. Ak nie, nastaví sa uprednostnená premenná *pojdes_do1* na hodnotu *true*. Ak však bola vykonaná voľba príchodu kabíny do tretieho poschodia, hodnota *go_to_1* zostáva v úrovni *true*, no najskôr bude vyhodnocovaná uprednostnená premenná *pojdes_do3*. Platí to i presne naopak, ak je výťah privolaný do tretieho poschodia, kontroluje sa, či nebola predtým stlačená voľba príchodu do prvého poschodia.

Na dverách kabíny sa v prípade jej pohybu zobrazuje prístroj *ind_dvere*. Vyhodnocuje premennú *m_smer* a svojím vzhľadom informuje užívateľa o aktuálnom, prípadne ďalšom smere pohybu.

6.3.1.3 Panel Fontána

Panel umožňuje užívateľovi nastavovať úroveň loptičky na modeli fontány. Na paneli sa nachádza niekoľko prístrojov typu *label*, slúžiacich k popisu jednotlivých častí. Prístroje tohto typu nie sú časované, sú zobrazované počas celého chodu aplikácie. Výnimkou sú prístroje *pozor* a *alert*, ktorých zobrazenie riadi prístroj typu *meter* s názvom *snimac*.

Prístroj typu *control* s názvom *cerpadlo* slúži k nastaveniu žiadanej hodnoty úrovne. Prístroj tohto typu má za parametrom *mode* uvedenú hodnotu 'horizontal_potentiometer'. Jeho rozsah je uvedený v rozmedzí od 0 do 1020 s krokom 1. Tieto hodnoty sa nachádzajú za parametrami *range_from*, *range_to* a *real_step*. Parameter *dec_places* určuje počet zobrazovaných desatinných miest. Rozsah je zvolený podľa desaťbitového prevodníka, na ktorý je snímač pripojený. V horných polohách je zistenie hodnoty problematické, preto je rozsah obmedzený.

Prístroj *snimac* zobrazuje výkon čerpadla vyjadrený v percentách. Typ prístroja špecifikuje hodnota 'horizontal_bar' parametra *mode*. Rozsah zobrazovanej hodnoty je od 0 do 100. Prístroj má v procedúre *OnActivate()* definované správanie prístrojov *pozor* a *alert*. Ich zobrazovanie je závislé na hodnote vyhodnocovanej veličiny tohto prístroja. V procedúre *OnStartup()* sú príkazmi *alert.Hide()*; a *pozor.Hide()*; oba prístroje skryté.

Hlavným prístrojom panela je prístroj typu *boolean_regulator* s názvom *dvojpolohovy_regulator*. Prístroj sa stará o samotnú reguláciu. Za parametrom *condition* je uvedená podmienka regulácie. K regulácii dochádza ak je regulačná odchýlka väčšia ako dvadsať a ak nie je čerpadlo preťažené. Parameter *wish_value* predstavuje žiadanú hodnotu. Jeho hodnotou je reálna premenná *ziadana_hodnota*. Parameter *expression* predstavuje regulovanú veličinu. Je ňou výraz s názvom *regulovana_velicina*. Parameter *output* predstavuje akčný zásah. Keďže sa jedná o dvojpolohový regulátor, predstavuje túto hodnotu premenná *akcny_zasah* typu *boolean*. Nie je však priamo riadiacou veličinou výkonu čerpadla. Hodnota 'true' parametra *show_description* zabezpečuje zobrazenie popisov jednotlivých sekcií regulátora. Skupiny parametrov *wish_value_section*, *value_section* a *output_section* predstavujú možnosti zobrazenia týchto hodnôt v jednotlivých častiach regulátora. O samotné riadenie výkonu čerpadla sa stará procedúra *OnActivate()*. V prípade nulovej hodnoty žiadanej veličiny je čerpadlo vypnuté. Ak nie je nulová, vyhodnocuje sa výstupná hodnota dvojpolohového regulátora. Ak regulátor

vykonáva akčný zásah, porovnáva sa regulačná odchýlka s jednotlivými odstupňovanými hodnotami a podľa toho sa nastavuje premenná *vykon_cerpadla*, ktorá ovláda priamo výstup PLC. Čerpadlo fontány je pripojené na osembitový D/A prevodník. Porovnávanie hodnôt regulačnej odchýlky prebieha v oboch smeroch, pri stúpaní i klesaní loptičky. Je tak zabezpečený plynulý chod čerpadla a loptička sa na vodnom stĺpci udrží.

6.3.2 Aplikácia modely_klient

Aplikácia predstavuje klienta k serverovej aplikácii *modely_server*. Klient slúži k diaľkovému ovládaniu modelov výtahu a fontány. Po svojom spustení sa prihlási na presne definovaný server. V aplikácii klienta nie je zavedený žiadny ovládač. Klient nekomunikuje priamo s modelom, ale ovláda prístroje serverovej aplikácie. Až tieto vykonávajú kroky k ovládaniu modelov. V sieti Ethernet sa môže nachádzať väčší počet spustených aplikácií klientov na jednotlivých staniciach. Všetky sa pri svojom spustení prihlásia na rovnaký server.

V hlavných nastaveniach aplikácie sa v sekcii *Časování* nachádza parameter *independent_procedure_execution*. Hodnota 'true' tohto parametra umožňuje používanie príkazov *pause;*, *yield;* a *wait;* vo všetkých procedúrach. Štandardne sú tieto príkazy povolené len v procedúre *OnActivate()*.

Aplikácia nemá definované žiadne komunikačné kanály. Jedinými zadefinovanými premennými sú skalárne dátové elementy typu *var* – *ActivePanel* a *vyst*. Premennú *ActivePanel* využíva systém pri určení aktívneho panela, od čoho sa odvíja časovanie viditeľných prístrojov. Reálna premenná *vyst* je použitá na zobrazenie žiadanej hodnoty pri voľbe úrovne vznášajúcej sa loptičky modelu fontány.

Naviazanie na serverovú aplikáciu je definované v sekcii *Moduly*. V sekcii je zavedený jeden modul s názvom *server_vytah*. Parameter *scope* tohto modulu je nastavený na hodnotu 'imported_remotely', čo znamená, že vzdialený modul bude hľadaný v sieti Ethernet. Hodnota 'on_startup' parametra *required* znamená, že modul klienta sa napojí na modul servera hneď po spustení aplikácie. Stanica, na ktorej je spustená serverová aplikácia je definovaná za parametrom *server*. Definícia zahŕňa IP adresu stanice a názov spustenej serverovej aplikácie.

Aplikácia je zložená z dvoch prekrývajúcich sa panelov, *Výtah_CW* a *Fontána*, ktorých prístroje sú časované vzhľadom na viditeľnosť panela. Keďže na paneli *Výtah_CW* sa nenachádza žiadny prístroj, ktorý by bolo nutné časovať, časovač vyhradený pre tento panel nie je v aplikácii využitý. Využitý je len časovač panela *Fontána*, ktorý aktivuje jeden prístroj, zobrazujúci žiadanú hodnotu úrovne loptičky.

V aplikácii sa nachádza jeden skrytý prístroj s názvom *keyb*. Umožňuje prepínanie medzi panelmi pomocou kláves. Definícia klávesových skratiek je uvedená v skupine *item* jeho parametrov. Tento prístroj dokáže spolu s presne definovanými parametrami vytvoriť sprievodca novou aplikáciou.

6.3.2.1 Panel *Výtah_CW*

Panel je rozdelený na dve časti. Ľavá časť predstavuje tlačidlá privolania výťahu na jednotlivých poschodiach. Táto časť aplikácie je vystavaná na paneli s názvom *patra*. Obsahuje ďalšie dva panely slúžiace na názorné oddelenie poschodí. Okrem vizuálnej stránky neplnia žiadnu úlohu. Dôležitými sú prístroje *cl_1*, *cl_2up*, *cl_2down* a *cl_3*, slúžiace k privolaniu výťahu. Tieto prístroje nie sú časované, sú to prepínače, ktorých parameter *logic* obsahuje hodnotu 'set_true_on_press', vďaka čomu sa chovajú ako tlačidlá. Úloha tlačidiel je definovaná v ich procedúrach *OnOutput()*. Táto procedúra je vykonaná v momente stlačenia tlačidla, kedy by došlo k zápisu do výstupnej premennej. Obsahuje jeden príkaz, ktorý vykoná stlačenie odpovedajúceho tlačidla v serverovej aplikácii. Napríklad tlačidlo na treťom poschodí obsahuje príkaz *server_vytah.chodba_3.SetValue(true)*; Toto tlačidlo navyše zabezpečuje správanie aplikácie po spustení. V procedúre *OnStartup()* je definované skrytie všetkých tlačidiel po dobu piatich sekúnd a následné zobrazenie. Deje sa tak z dôvodu neinformovanosti aplikácie klienta o tom, či je serverová aplikácia pripravená na použitie. Tento čas je dostatočný na uskutočnenie všetkých potrebných komunikácií v prípade takmer súčasného spustenia oboch aplikácií.

Pravá časť panela predstavuje kabínu výťahu. Panel *kabina* opäť obsahuje niekoľko panelov tvoriacich vizuálnu stránku aplikácie. Dôležitými sú tlačidlá *cl_1k*, *cl_2k* a *cl_3k*. Ich nastavenia sa zhodujú s nastaveniami tlačidiel umiestnených na paneli *patra*. Rozdiel spočíva v prepínaní iných odpovedajúcich tlačidiel v serverovej aplikácii.

6.3.2.2 Panel Fontána

Na paneli sa nachádza prístroj typu *control* s názvom *nast_ziad_hod*, slúžiaci k nastaveniu žiadanej hodnoty. Výstupná hodnota uvedená za parametrom *output* je využívaná len miestne, vyhodnocuje ju prístroj typu *meter* s názvom *ziad_hod*. Prístroj *control* pri zmene svojej výstupnej hodnoty túto hodnotu nastavuje na prístroji *cerpadlo* v serverovej aplikácii, pomocou príkazu *server_vytah.cerpadlo.SetValue(vyst)*; . Po nastavení tejto hodnoty dochádza k aktivovaniu prístroja *cerpadlo* pomocou príkazu *send server_vytah.cerpadlo*; , čo zabezpečí zápis nastavenej hodnoty do jeho výstupu.

6.3.3 Aplikácia výťah_klient

Aplikácia predstavuje klienta k prvému panelu aplikácie *modely_server*. Jedná sa o ovládanie modelu výťahu s použitím priloženého programu na disku CD pre PLC SAIA. V aplikácii nie je zavedený žiadny ovládač. Taktiež nie sú definované žiadne premenné. Aplikácia sa po svojom štarte pripojí na serverovú aplikáciu a využíva jej premenné.

Na nadviazanie spojenia je nutné v aplikácii vytvoriť nový dátový element. Keďže ide o premenné komunikované s modelom, element s názvom *vzdialene* je typu *channel*. Musí zostať prázdny. V hlavných nastaveniach elementu je za parametrom *scope* uvedená hodnota 'imported_remotely'. To znamená, že premenné použité v rámci aplikácie budú vyhľadávané v sieti Ethernet. Parameter *server* definuje IP adresu servera a názov modulu, serverovej aplikácie, a tiež názov dátovej sekcie, na ktorú sa klient pripojí. Názov modulu a sekcie sa nemení, no IP adresu je potrebné špecifikovať podľa stanice, na ktorej je serverová aplikácia spustená. Pri vývoji bola použitá stanica s IP adresou 10.5.16.203.

Prevedenie aplikácie je podobné ako u serverovej verzie, niektoré voľby však nie sú užívateľovi prístupné. Popis nastavení prístrojov je totožný s popisom nastavení v serverovej aplikácii.

6.3.4 Popis zavedených kanálov v distribuovanej aplikácii

Zavedené kanály využíva serverová aplikácia *modely_server* na komunikáciu s modelmi. Aplikácie klientov, okrem aplikácie *výtah_klient*, tieto kanály nevyužívajú.

6.3.4.1 Kanály ovládača Vytah

- 101 – 103 Tri vstupné kanály typu *boolean* ktorým odpovedajú vstupy I 29, I 30 a I 31. Sú súčasťou kanálov *vytah_klient*, slúžia na zobrazenie aktuálneho poschodia.
- 105 – 112 Osem vstupných kanálov typu *boolean* uvedených v sekcii *vytah_server* a *vytah_klient*. Predstavujú hodnoty ôsmich výstupov začínajúc výstupom O 0. Informujú užívateľa o dosiahnutom a nasledujúcom poschodí.
- 115 – 117 Tri výstupné kanály typu *boolean*, zapisujúce do registrov F 1, F 2 a F 3. Nachádzajú sa medzi kanálmi *vytah_server* a *vytah_klient*. Slúžia na zastavenie výťahu, k príchodu na prvé poschodie a k opätovnému uvedeniu do prevádzky.
- 120 – 127 Osem výstupných kanálov typu *boolean*, zapisujúcich do registrov F, začínajúcich adresou F 23. Predstavujú tlačidlá na chodbách a v interiéri kabíny.

6.3.4.2 Kanály ovládača Vytah_CW

- 201 – 203 Tri vstupné kanály typu *boolean*. Do systému Control Web prinášajú informácie o stavoch vstupov I 29, I 30 a I 31, čomu odpovedajú premenné *poschodie_1*, *poschodie_2* a *poschodie_3*. Sú to hodnoty zo snímačov na jednotlivých poschodiach.
- 311 – 312 Dva výstupné kanály typu *boolean*. Slúžia na zápis do výstupov O 6 a O 7, čím sa určuje smer a pohyb motora výťahu.

6.3.4.3 Kanály ovládača Fontana

- 130 Jeden výstupný kanál typu *uint8*. Slúži na zápis reálnej hodnoty do registra R 10, čím je ovládaný výkon čerpadla fontány.
- 131 Jeden vstupný kanál typu *uint*, slúžiaci na získanie hodnoty zo snímača.

ZÁVER

Pre návrh a prácu so SCADA systémami je vhodné najskôr získať všeobecné informácie o zložení a funkcii takýchto systémov, ktoré ponúkla i táto práca. Základom úspechu je voľba vhodného komunikačného systému a výber komunikačného protokolu spĺňajúceho požiadavky užívateľov. V nemalej miere prispieva k úspechu voľba vhodného programového vybavenia.

Aplikácie, ktoré sú súčasťou tejto práce využívajú štandardné komunikačné linky, so zameraním na sieť Ethernet a komunikačný protokol TCP/IP. Prostredie systému Control Web 6 je užívateľsky veľmi prívetivé. Nové aktualizácie neustále odstraňujú i tie najmenšie nedostatky. Veľké množstvo pripravených virtuálnych prístrojov poskytne užívateľovi bezpochyby širokú paletu možností bez veľkých nárokov na ich vlastnú tvorbu. Podstatným uľahčením tvorby aplikácií je existencia množstva natívnych procedúr a možnosť tvorby rôznych užívateľských procedúr. Pre tvorbu aplikácií je taktiež nutnosťou znalosť a pochopenie rozdielov medzi dvoma ponúkanými typmi aplikácií, aplikáciou reálneho času a aplikáciou závislou na zmene dát. Pri tvorbe aplikácií sa užívateľ naučí definovať a rozpoznávať prácu v jednotlivých časovacích krokoch, na základe čoho sa môže vyvarovať mnohým prípadom takzvaného nezrozumiteľného konania aplikácie. Pri práci a komunikácii s rôznymi druhmi PLC je vhodné si uvedomiť časové nároky na uskutočnenie každej komunikácie. Preto je správny a bezproblémový chod aplikácií závislý na vhodnej voľbe a usporiadaní časovaní virtuálnych prístrojov, s kladením veľkého dôrazu na obmedzovanie zbytočného vyhodnocovania dát, so záujmom o znižovanie nárokov na procesor a samotný komunikačný systém.

Problémy v počiatkoch nadväzovania komunikácie sa podarilo s úspechom odstrániť. Je dobré mať prehľad o používaných ovládačoch komunikácie a klásť veľký dôraz na prehľad v komunikačných kanáloch. Pri tvorbe distribuovaných aplikácií musí užívateľ dbať na presné rozlíšenie medzi serverovou aplikáciou a aplikáciou klienta, správne zvoliť spôsoby zdieľania a pamätať na princíp, že pred spustením klientskej aplikácie musí byť server aplikácie už spustený. Výnimkou sú aplikácie, ktoré svoje dáta a moduly zdieľajú miestne, v rámci jedného PC.

Aplikácie staršej verzie systému Control Web sú s novou verziou kompatibilné po odstránení slova „window“, ktoré sa nachádza pred názvom a definíciou panela pozadia.

ZÁVER V ANGLIČTINE

In order to design and work with SCADA systems, it is useful to first obtain general information about the construction and functioning of these systems, such as that offered in this work. The key to success is the choice of the appropriate communication system and the communication protocol that fulfills the demands of the users. The choice of the appropriate software likewise contributes to success in no small way.

The applications which are part of this work rely on the use of standard communication links, with a focus on the Ethernet network and the TCP/IP communication protocol. The Control Web 6 system is very user-friendly. Updates constantly remove even the most minor errors. The vast amount of pre-made virtual devices available give the user a wide range of possibilities, while limiting the effort required in their creation. The existence of many native procedures and the possibility of creating diverse procedures make applications easier to create. As part of the process of creating applications, it is crucial to know and understand the differences between the two available types of applications: real-time applications and applications dependent on data changes. In the process of creating applications, the user learns to define and differentiate work in individual, sequential steps. This will help him to avoid creating an application that behaves in an unpredictable way. When working and communicating with different types of PLCs, it is important to take into account the demands on time in the realization of each communication. This is why the proper and error-free running of applications depends on the appropriate choice of and coordination in the timing of virtual devices, placing emphasis on the limiting of useless data analyses, and taking an interest in lowering the demands on the processor and the communication system itself.

The problems in the early stages of linking communication were able to be successfully overcome. It is good to have an overview of the communication drivers being used, and to greatly emphasize the orientation in communication channels. When creating distributed applications, the user must pay careful attention to the differences between the server and client applications, choose the correct sharing processes, and remember the principle that before the client application is run, the server application must already be running. The exceptions to this rule are applications that share their data and modules locally, in the framework of one PC.

Applications in the older version of the Control Web system are compatible with the new version after deleting the word "window" which is found before the name and definition of the background panel.

ZOZNAM POUŽITEJ LITERATURY

- [1] BÍLÝ, Radek, et al. *Control Web 2000*. 1. vyd. Praha : Computer Press, 1999. 384 s. ISBN 80-7226-258-0.
- [2] CLARKE, Gordon; REYNDERS, Deon; WRIGHT, Edwin. *Practical Modern SCADA Protocols: DNP3, 60870.5 and Related Systems*. Edited by Vivek Mehra. 1st ed. Great Britain : Newnes, 2004. 544 s. ISBN 978-0-7506-5799-0, ISBN 0-7506-5799-5.
- [3] BAILEY, David; WRIGHT, Edwin. *Practical SCADA for Industry*. Edited by Vivek Mehra. 1st ed. Great Britain : Newnes, 2003. 304 s. ISBN 978-0-7506-5805-8, ISBN 0-7506-5805-3.
- [4] AXELSON, Jan. *Serial Port Complete : Programming and Circuits for RS-232 and RS-485 Links and Networks*. Madison (USA) : Lakeview Research, 1998. 322 s. ISBN 978-0-9650819-2-4, ISBN 0-9650819-2-3.
- [5] KNOBLOCH, Tomáš. Ethernet s odezvou v reálném čase v pojetí Schneider Electric. *AUTOMATIZACE* [online]. červen 2004, 47, 6, [cit. 2010-04-15]. s. 408-409. Dostupný z WWW: < <http://www.automatizace.cz/article.php?a=152> >.
- [6] NÁVRAT, Petr. Nové vlastnosti systému PROMOTIC. *AUTOMATIZACE* [online]. Září 2007, 50, 9, [cit. 2010-04-19]. Dostupný z WWW: < <http://www.automatizace.cz/article.php?a=1875> >.
- [7] ZEZULKA, František; HYNČICA, Ondřej. Požadavky distribuovaných systémů řízení na průmyslovou síť Ethernet. *AUTOMATIZACE* [online]. únor 2006, 49, 2, [cit. 2010-05-19]. s. 86-89. Dostupný z WWW: < <http://www.automatizace.cz/article.php?a=1069> >.
- [8] BURGET, Pavel. Profibus-PA – řešení pro automatizaci procesů. *AUTOMA* [online]. 2001, 1, [cit. 2010-04-25]. Dostupný z WWW: < http://www.odbornecasopisy.cz/index.php?id_document=33429 >.
- [9] KOSEK, Rostislav. Profinet - řešení firmy Siemens pro průmyslový Ethernet v automatizaci. *AUTOMATIZACE* [online]. 2004, 47, 9, [cit. 2010-04-10]. s. 564-565. Dostupný z WWW: < <http://www.automatizace.cz/article.php?a=287> >.

- [10] PFEIFFER, Andreas. Průmyslová síť Ethernet - srovnání uklidňuje. *AUTOMATIZACE* [online]. únor 2006, 49, 2, [cit. 2010-04-12]. s. 99-101. Dostupný z WWW: < <http://www.automatizace.cz/article.php?a=1076> >.
- [11] ZEZULKA, František; HYNČICA, Ondřej. Průmyslový Ethernet III: Fyzické provedení sítě Ethernet. *AUTOMA* [online]. 2007, 6, [cit. 2010-05-05]. Dostupný z WWW: < http://www.odbornecasopisy.cz/index.php?id_document=34395 >.
- [12] ZEZULKA, František; HYNČICA, Ondřej. Průmyslový Ethernet VIII: Ethernet Powerlink, Profinet. *AUTOMA* [online]. 2008, 5, [cit. 2010-04-12]. s. 62-66. Dostupný z WWW: < <http://www.odbornecasopisy.cz/res/pdf/37288.pdf> >.
- [13] ČERVENKA, Zbyněk. Wonderware InTouch 10.0 a Wonderware System Platform 3.0. *AUTOMATIZACE* [online]. Září 2007, 50, 9, [cit. 2010-04-20]. s. 582-585. Dostupný z WWW: < <http://www.automatizace.cz/article.php?a=1872> >.
- [14] BÉLAI, Igor. *Priemyselny Ethernet*. [s.l.], 2006. 19 s. Oborová práce. Slovenská technická univerzita, Fakulta elektrotechniky a informatiky. Dostupné z WWW: < http://www.kar.elf.stuba.sk/predmety/pkom/Prednasky/Prednaska_11/PKOM_PR-11_05-06-a.pdf >.
- [15] MOLDA, Vojta. *Univerzální USB modul pro sběr dat a řízení fyzikálního experimentu*. [s.l.], 2009. 39 s. Bakalářská práce. Univerzita Karlova v Praze, Matematicko-fyzikální fakulta, Katedra fyziky nízkých teplot. Vedoucí práce Ing. Miloš Pfeffer, CSc.
- [16] BÉLAI, Igor. *Priemyselne komunikacie : Zbornica Profibus* [online]. [s.l.], 2006. 48 s. Oborová práce. Slovenská technická univerzita, Fakulta elektrotechniky a informatiky, Katedra automatizácie a regulácie. Dostupné z WWW: < http://www.kar.elf.stuba.sk/predmety/pkom/Prednasky/Prednaska_07/PKOM_PR-07_05-06.pdf >.
- [17] POUCHA, Pavel. *Automa* [online]. 2007 [cit. 2010-02-20]. Ethernet nahrazuje klasické průmyslové komunikační linky. Dostupné z WWW: < http://www.odbornecasopisy.cz/index.php?id_document=31237 >.

- [18] COUFAL, Tomáš. *Hw.cz* [online]. 21. Květen 2007 [cit. 2010-04-22]. TIRS.NET. Dostupné z WWW: < <http://hw.cz/teorie-praxe/art1910-tirs-net-vizualizacni-ridici-system-zalozeny-na-technologie-microsoft-net.html> >.
- [19] *TEDIA spol. s r. o.* [online]. 1994, poslední aktualizace: 09.02.2010 [cit. 2010-02-20]. TEDIA: Obecné vlastnosti komunikačních prostředků. Dostupné z WWW: < <http://www.tedia.cz/podpora/komunikace-obecne-vlastnosti.html> >.
- [20] PIŠL, Roman; BALDA, Pavel KOMUNIKACE ETHERNET POWERLINK PRO ŘÍDICÍ SYSTÉM REX. In *Studentská Vědecká Konference 2009* [online]. [s.l.] : [s.n.], 2009 [cit. 2010-02-25]. s. 2. Dostupné z WWW: < http://www.fav.zcu.cz/fakulta/akce/studentska-vedecka-konference/2009/cd/pdf/KKY/Mgr/pisl_roman.pdf >.
- [21] Rozhranie človek - stroj. In *Prehľadová štúdia* [online]. [s.l.] : [s.n.], 2009 [cit. 2010-02-07]. Dostupné z WWW: < <http://www.atpjournals.sk/> >.
- [22] Moravské přístroje a.s., firemná literatúra. Control Web 6 [online]. [cit. 2010-05-02]. Dostupné z WWW: < <http://www.mii.cz/> >.

ZOZNAM POUŽITÝCH SYMBOLOV A ZKRATIEK

CSMA/CD	Carrier Sense Multiple Access/Collision Detection
DCS	Distributed Control System
HMI	Human – Machine Interface
HRT	Hard Real – Time
HSE	High Speed Ethernet
IED	Intelligent Electronic Device
MMI	Man Machine Interface
PLC	Programmable Logic Controller
RTU	Remote Terminal Unit
SCADA	Supervisory Control and Data Acquisition
TCP/IP	Transmission Control Protocol/Internet Protocol

ZOZNAM OBRÁZKOV

<i>Obr. 1. Ukážka typického SCADA systému [3]</i>	13
<i>Obr. 2. Ukážka jednoduchého systému [3]</i>	14
<i>Obr. 3. Systém s využitím PLC/DCS [3]</i>	16
<i>Obr. 4. Systém s využitím IED [3]</i>	17
<i>Obr. 5. Architektúra peer-to-peer [2]</i>	18
<i>Obr. 6. Viacnásobné prepojenia [2]</i>	19
<i>Obr. 7. Využitie jednotky ako prostredníka v komunikácii [2]</i>	19
<i>Obr. 8. Použitie opakovacích zariadení [2]</i>	20
<i>Obr. 9. Prehľad architektúr systémov s použitím priemyselného Ethernetu</i>	24
<i>Obr. 10. Príklady prevedení priemyselných PC [21]</i>	28
<i>Obr. 11. Zmena stavu komunikácie v prostredí Mosaic</i>	34
<i>Obr. 12. Ukážka prostredia systému Control Web 6</i>	35
<i>Obr. 13. Rozšírená ponuka parametrov</i>	37
<i>Obr. 14. Ukážka inšpektora prístroja u panela s názvom „pozadie“</i>	38
<i>Obr. 15. Konfigurácia PLC</i>	82
<i>Obr. 16. Voľba komunikácie po sériovej linke</i>	82
<i>Obr. 17. Detail portu voľby Setup</i>	83
<i>Obr. 18. Voľba typu PLC</i>	83
<i>Obr. 19. Výber protokolu S-Bus</i>	84
<i>Obr. 20. Detaily komunikácie cez RS-232</i>	84
<i>Obr. 21. Výber TCP/IP protokolu a nastavenie IP adresy stanice</i>	85
<i>Obr. 22. Nahratie konfigurácie – voľba Download</i>	85
<i>Obr. 23. Potvrdenie nahrávania</i>	85
<i>Obr. 24. Voľby nahrávania programu</i>	86
<i>Obr. 25. Nahrávanie programu</i>	86
<i>Obr. 26. Voľby ovládania komunikácie</i>	86
<i>Obr. 27. Nastavenia komunikácie cez RS-232</i>	87
<i>Obr. 28. Výber komunikačného portu</i>	87
<i>Obr. 29. Výber typu PLC</i>	88
<i>Obr. 30. Voľba protokolu S-Bus</i>	88
<i>Obr. 31. Parametre komunikačnej linky RS-232</i>	89

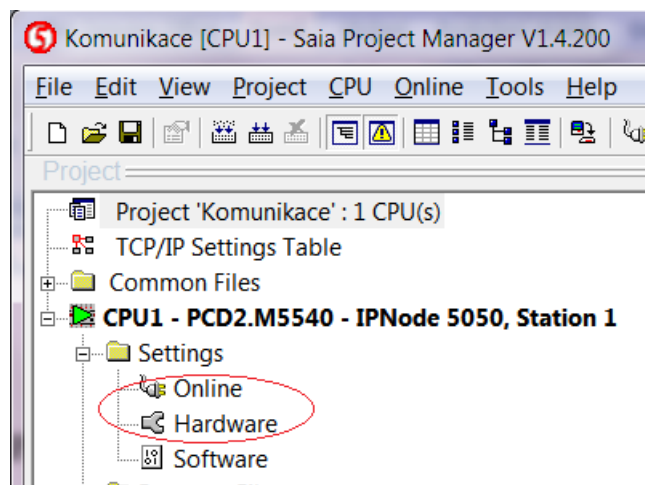
<i>Obr. 32. Nastavenie adresy stanice a výber protokolu TCP/IP</i>	89
<i>Obr. 33. Nahrávanie konfigurácie</i>	90
<i>Obr. 34. Potvrdenie nahrávania konfigurácie</i>	90
<i>Obr. 35. Nahrávanie prázdneho programu</i>	90
<i>Obr. 36. Manažér projektu</i>	91
<i>Obr. 37. Výber typu PLC</i>	91
<i>Obr. 38. Nastavenie parametrov sériovej komunikácie</i>	92
<i>Obr. 39. Preklad a spustenie programu</i>	92
<i>Obr. 40. Oznámenie prekladu</i>	92
<i>Obr. 41. Potvrdenie nahrávania</i>	93
<i>Obr. 42. Voľba typu reštartu</i>	93
<i>Obr. 43. Ukončenie komunikácie</i>	93
<i>Obr. 44. Výber komunikácie sériovým rozhraním USB</i>	94
<i>Obr. 45. Detail možností nastavení konfigurácie</i>	95
<i>Obr. 46. Vizualizácia výťahu riadeného pomocou PLC SAIA</i>	96
<i>Obr. 47. Klient aplikácie riadenia výťahu</i>	96
<i>Obr. 48. Serverová aplikácia riadenia modelu výťahu systémom Control Web 6</i>	97
<i>Obr. 49. Klient riadenia modelu systémom Control Web 6</i>	97
<i>Obr. 50. Serverová aplikácia ovládania modelu fontány</i>	98
<i>Obr. 51. Klient ovládania modelu fontány</i>	98
<i>Obr. 52. Indikácia vstupov a voľby stavov výstupov</i>	99
<i>Obr. 53. Ukážka práce s registrami</i>	99
<i>Obr. 54. Vyobrazenia stavov PLC, dátumu a času</i>	100
<i>Obr. 55. Indikácia vstupov a ovládanie výstupov PLC</i>	101
<i>Obr. 56. Ukážka práce s registrami</i>	102
<i>Obr. 57. Vyobrazenia niektorých údajov z PLC</i>	102

ZOZNAM PRÍLOH

- P I Konfigurácia PLC SAIA – Fontána.
- P II Konfigurácia PLC SAIA – Výt'ah.
- P III Konfigurácia PLC Tecomat.
- P IV Konfigurácia PLC SAIA.
- P V Ukážky aplikácií ovládania modelov.
- P VI Ukážka aplikácie komunikace_Teco.
- P VII Ukážka aplikácie komunikace_saia.

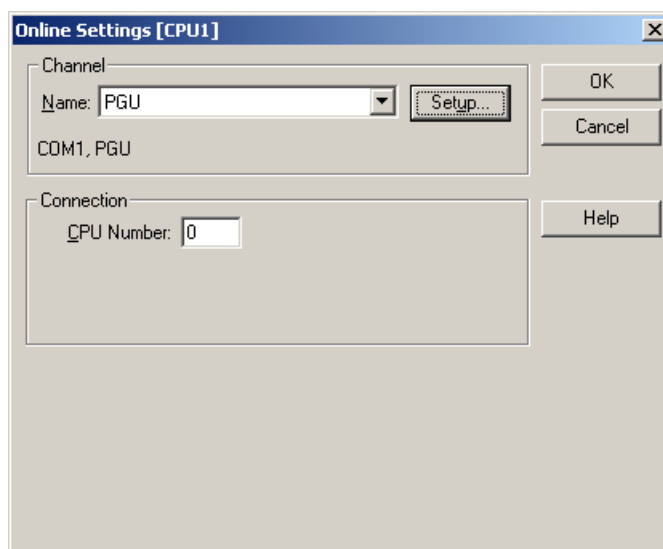
PRÍLOHA P I: KONFIGURÁCIA PLC SAIA – FONTÁNA.

Na priloženom CD sa nachádza pripravený program pre komunikáciu s modelom fontány. Po otvorení tohto programu v programovacom prostredí PG5 je nutné skontrolovať nasledovnú konfiguráciu. Počas konfigurácie je PLC SAIA pripojené k PC pomocou sériovej linky RS-232. Nastavenia konfigurácie sa týkajú sekcií *Online* a *Hardware*.

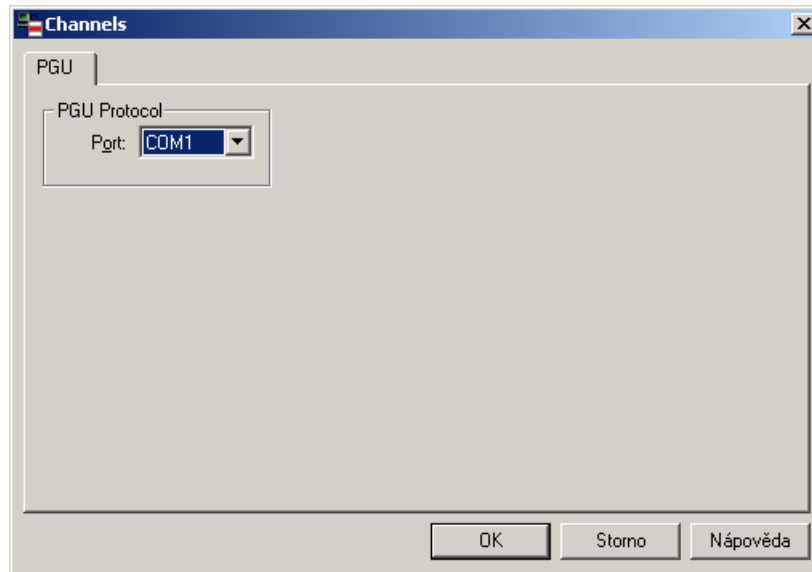


Obr. 15. Konfigurácia PLC

Online:

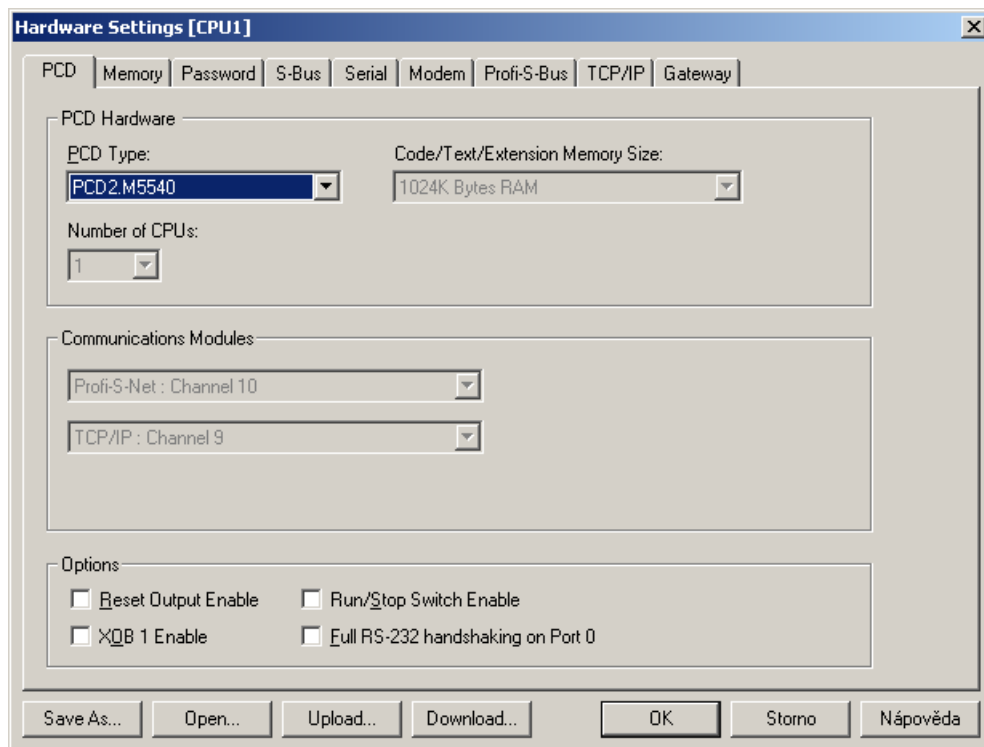


Obr. 16. Voľba komunikácie po sériovej linke

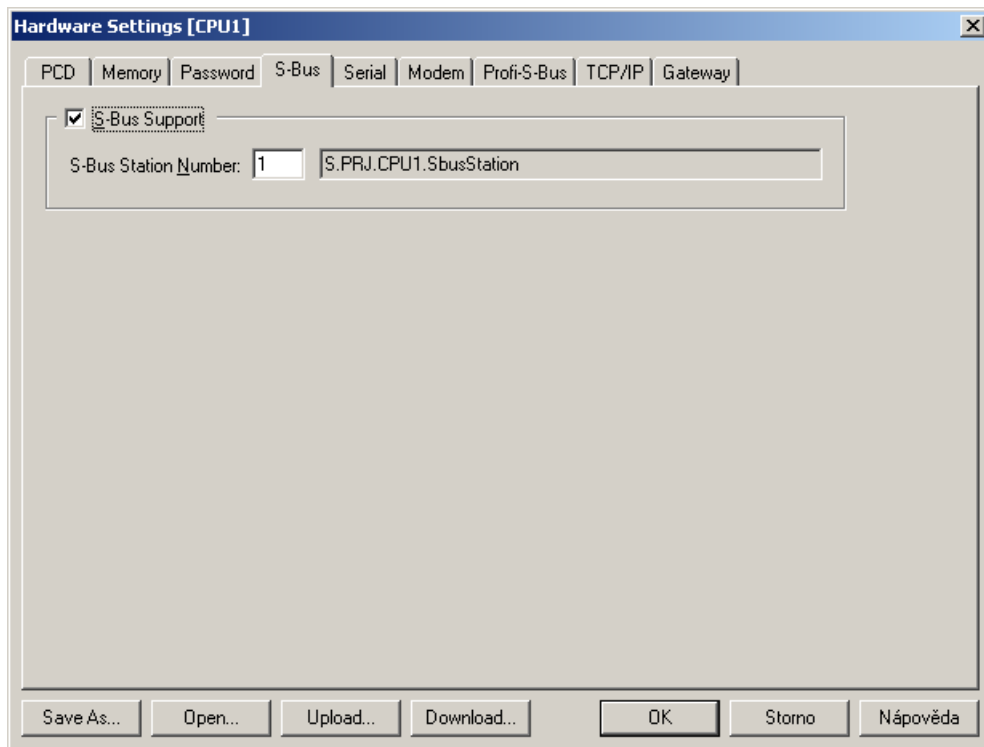


Obr. 17. Detail portu volby Setup

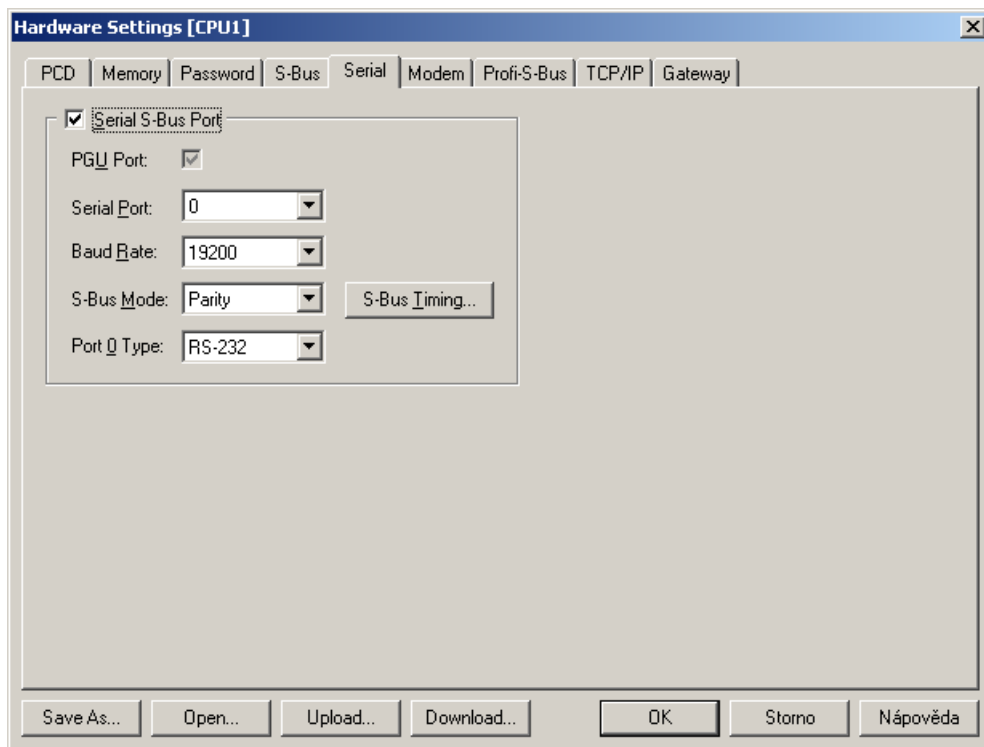
Hardware:



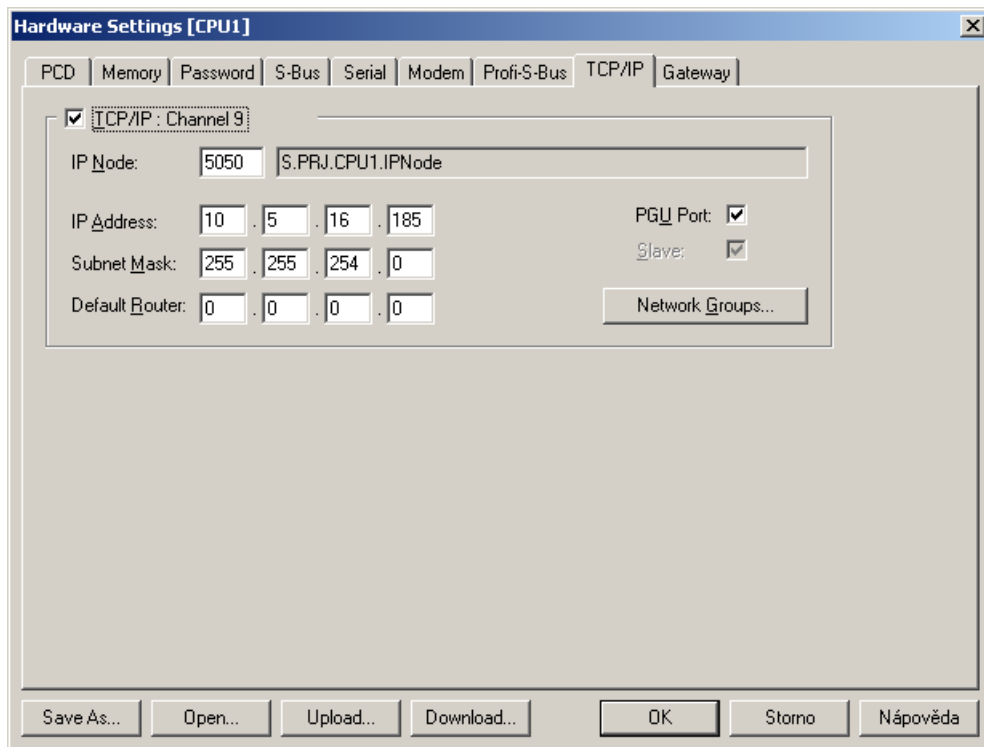
Obr. 18. Volba typu PLC



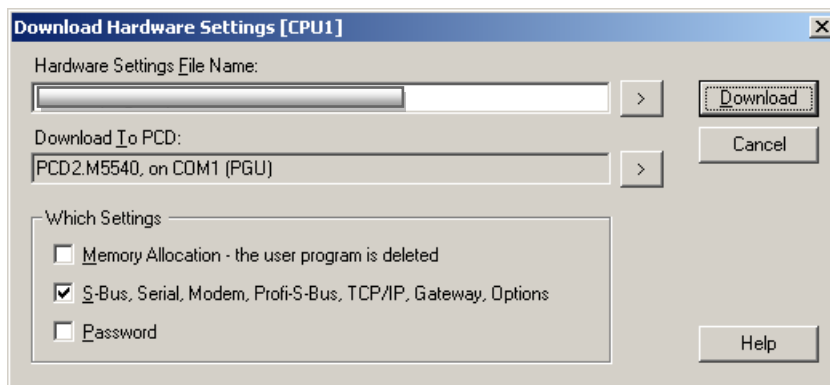
Obr. 19. Výber protokolu S-Bus



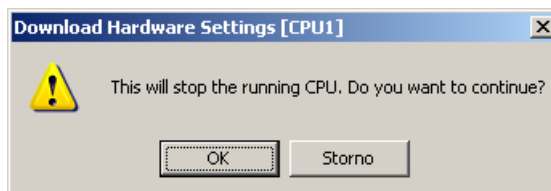
Obr. 20. Detaily komunikácie cez RS-232



Obr. 21. Výber TCP/IP protokolu a nastavenie IP adresy stanice

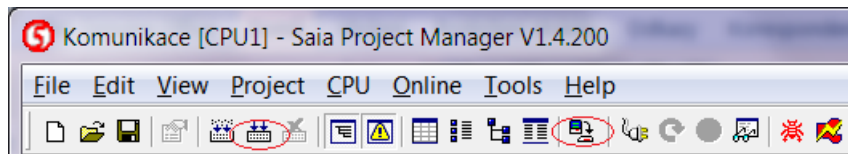


Obr. 22. Nahratie konfigurácie – voľba Download

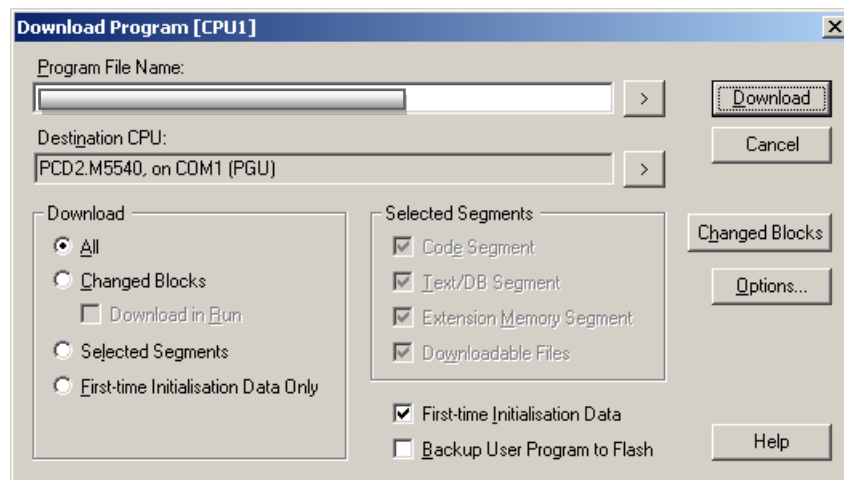


Obr. 23. Potvrdenie nahrávania

Zostávajúce okno potvrdíme stlačením tlačidla *OK*. Následne zvolíme z hornej nástrojovej lišty voľbu *Rebuild All Files* a voľbu *Download Program*.

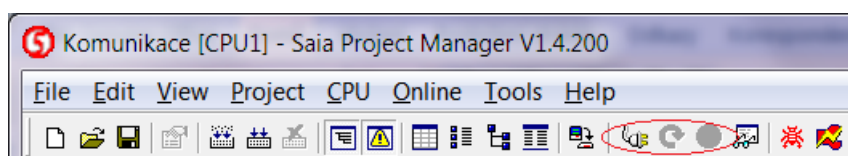


Obr. 24. Voľby nahrávania programu



Obr. 25. Nahrávanie programu

Voľbou *Download* spustíme nahrávanie programu do PLC. Následne sa voľbou *Go On/Offline* pripojíme na PLC a voľbou *Run* spustíme vykonávanie programu. PLC je pripravené na použitie. Opätovnou voľbou *Go On/Offline* odpojíme komunikáciu po sériovej linke a prácu v programovacom prostredí PG5 môžeme ukončiť.

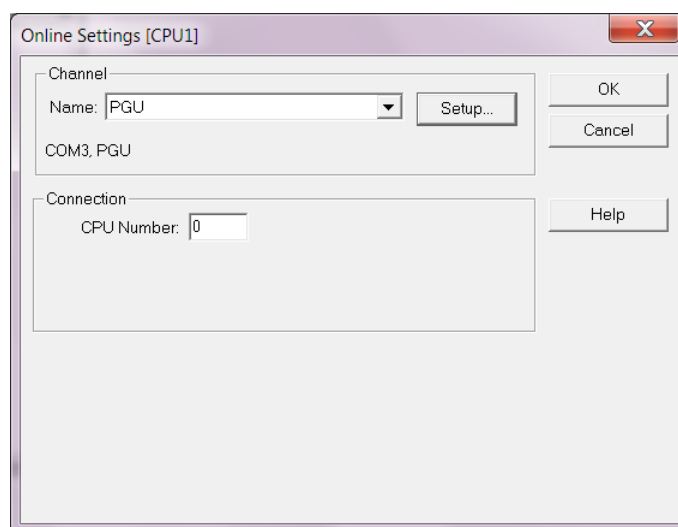


Obr. 26. Voľby ovládania komunikácie

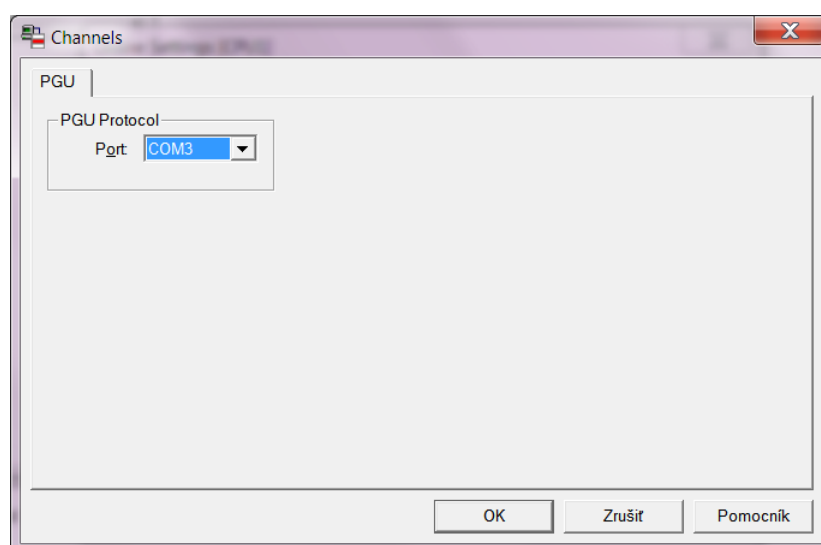
PRÍLOHA P II: KONFIGURÁCIA PLC SAIA – VÝŤAH.

V programovacom prostredí PG5 je nutné vytvoriť prázdny program, ktorý bude následne nahratý do PLC. V prípade prvého panela vizualizácie je nutné do PLC nahrať program nachádzajúci sa na disku CD. Nastavenia sa opäť týkajú sekcie *Online* a *Hardware*, obdobne ako pri konfigurácii PLC pri práci s modelom fontány, uvedenej v prílohe P I. Konfigurácia prebieha pomocou sériovej linky RS-232.

Online:



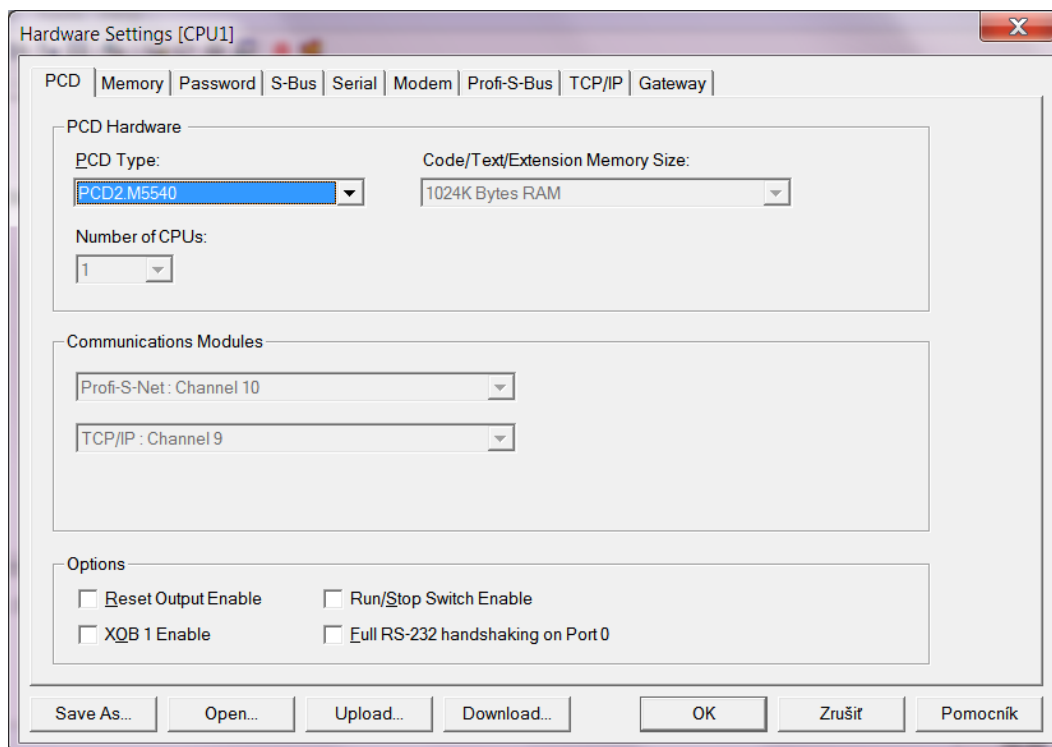
Obr. 27. Nastavenia komunikácie cez RS-232



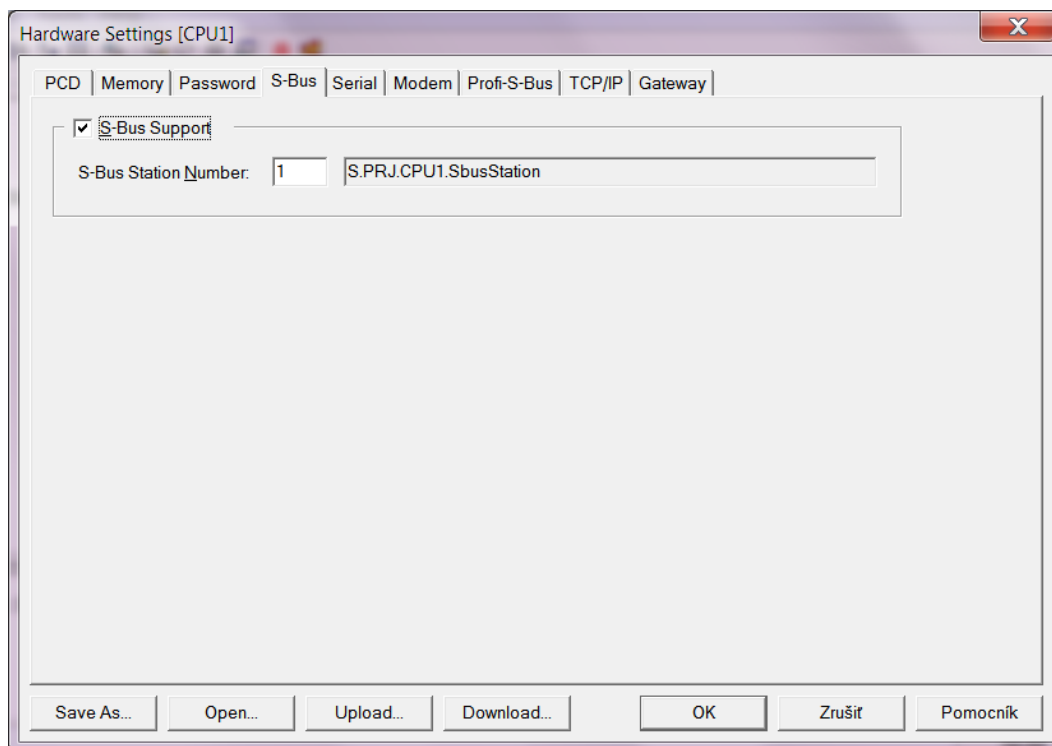
Obr. 28. Výber komunikačného portu

Vo väčšine prípadov je nutné vybrať z ponuky port COM1. Hodnota sa môže líšiť podľa spôsobu prepojenia.

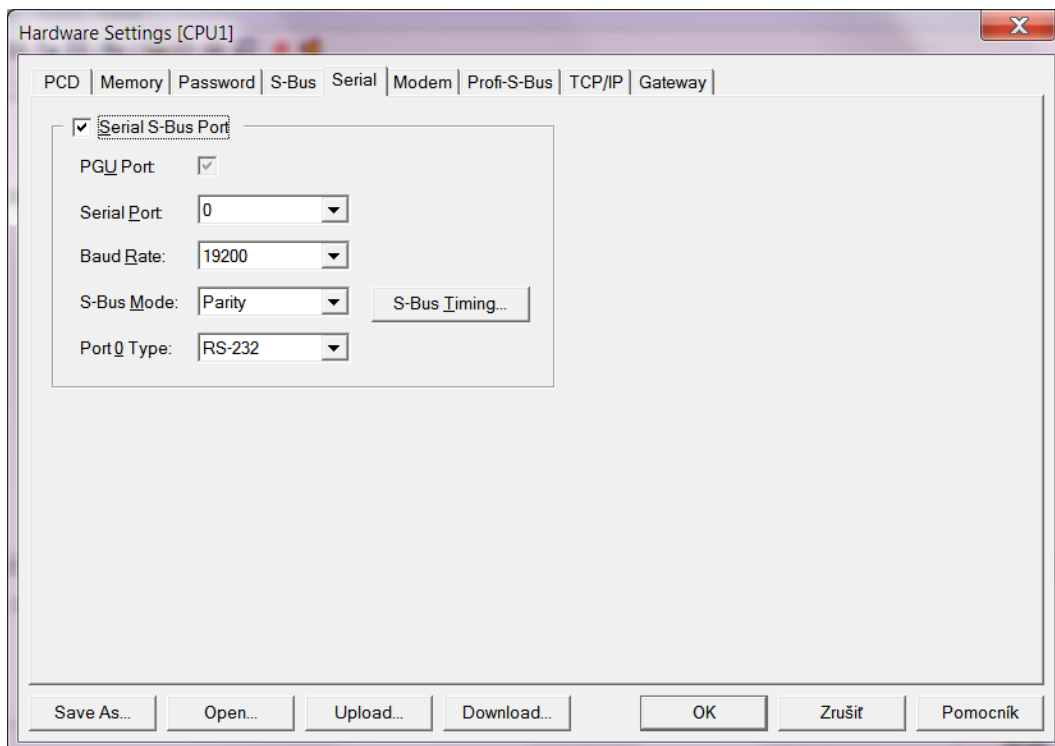
Hardware:



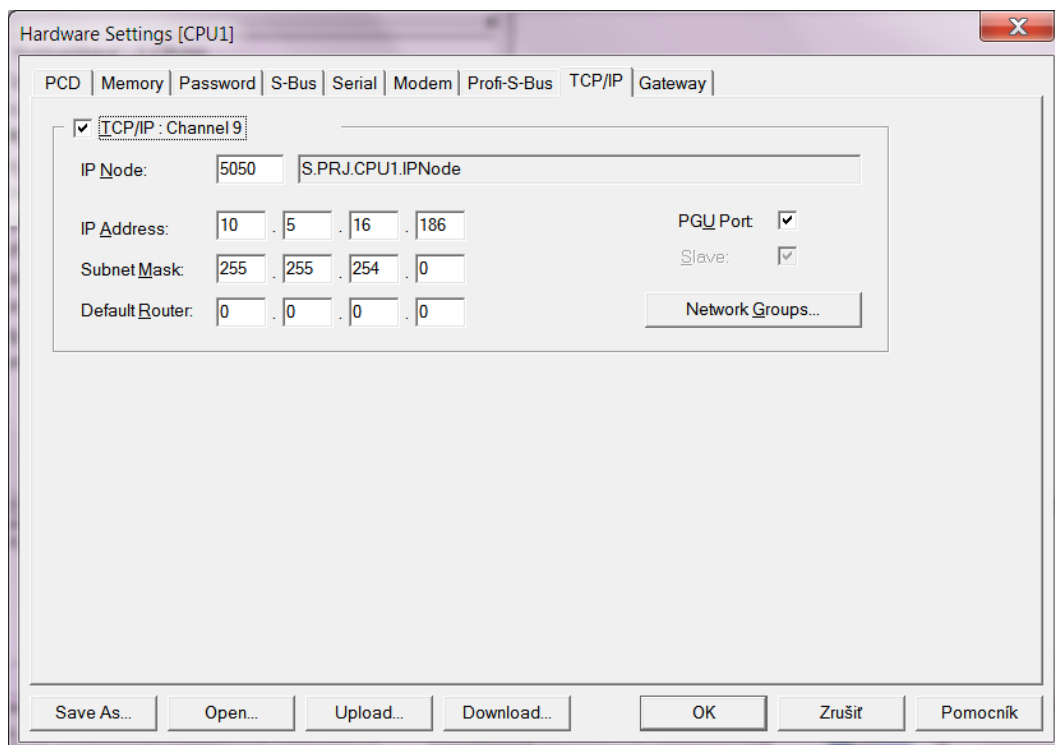
Obr. 29. Výber typu PLC



Obr. 30. Voľba protokolu S-Bus

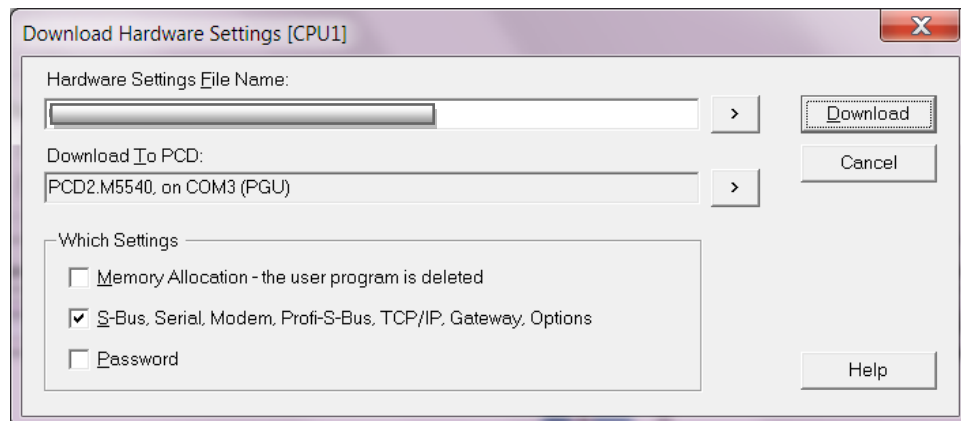


Obr. 31. Parametre komunikačnej linky RS-232

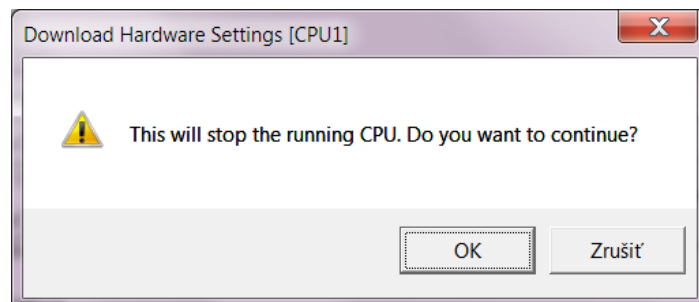


Obr. 32. Nastavenie adresy stanice a výber protokolu TCP/IP

Následne zvolíme možnosť *Download*.

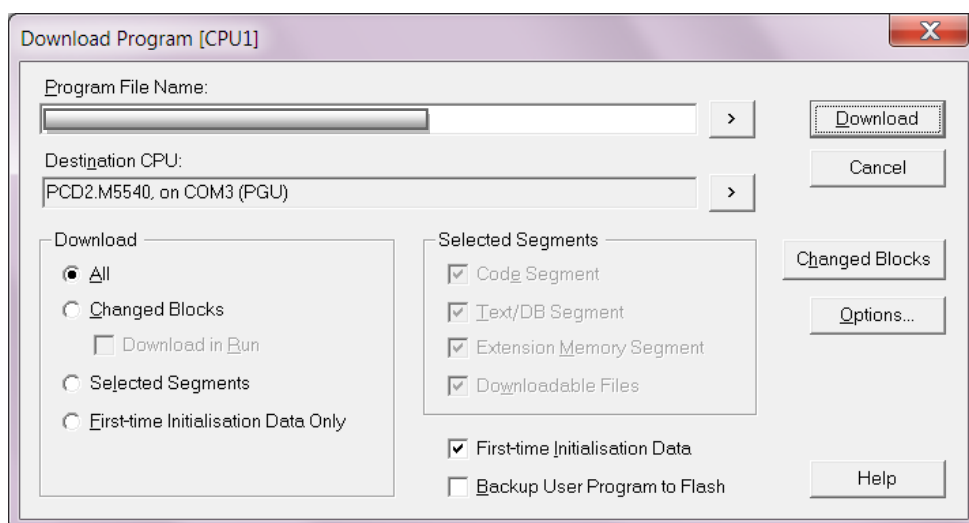


Obr. 33. Nahrávanie konfigurácie



Obr. 34. Potvrdenie nahrávania konfigurácie

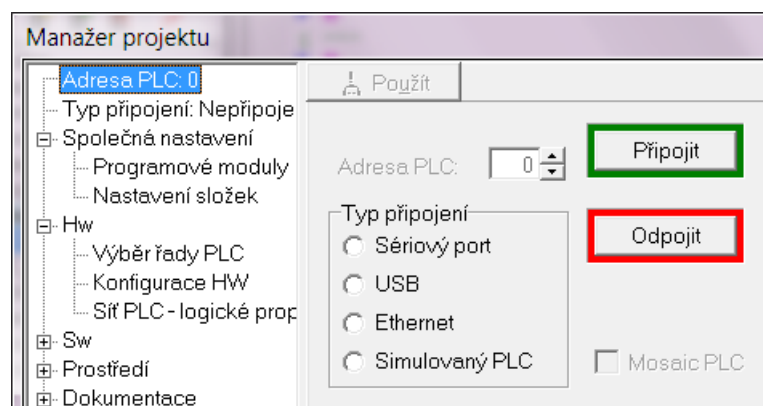
Po potvrdení a nahratí konfigurácie do PLC potvrdíme okno nastavení tlačidlom *OK*. Ďalší postup je totožný s postupom uvedeným v prílohe P I.



Obr. 35. Nahrávanie prázdneho programu

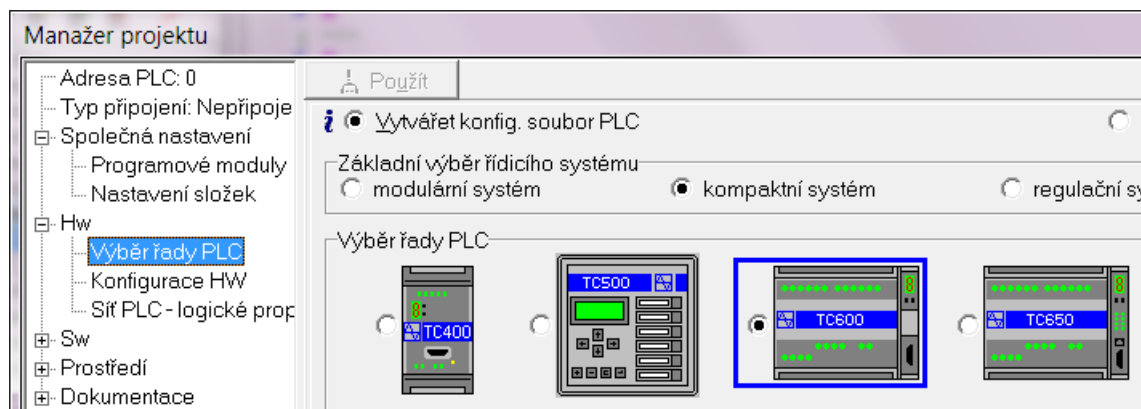
PRÍLOHA P III: KONFIGURÁCIA PLC TECOMAT.

Nastavenie konfigurácie PLC Tecomat pre komunikáciu s prostredím Control Web 6 je nutné vykonať v programovacom prostredí Mosaic. Prvým krokom je vytvorenie nového projektu s prázdny programom. Následne vyberieme voľbu *Projekt – Manažer projektu*.



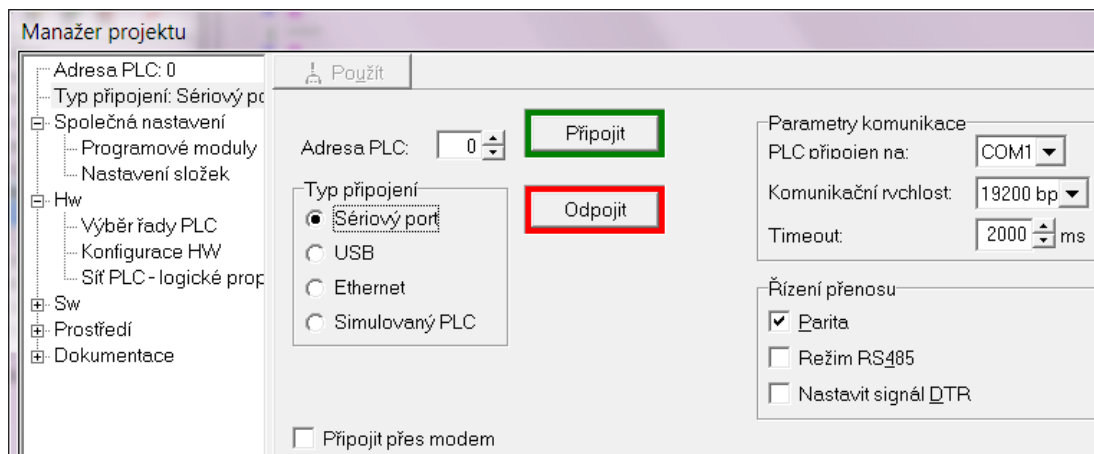
Obr. 36. Manažer projektu

V záložke *Výběr řady PLC* zvolíme typ TC600.



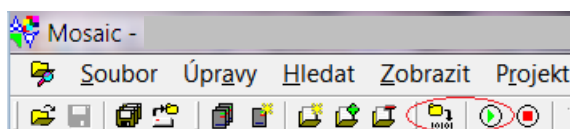
Obr. 37. Výber typu PLC

V ponuke *Typ připojení* vyberieme *Sériový port*, nastavíme parametre komunikácie a potvrdíme voľbou *Připojit*.

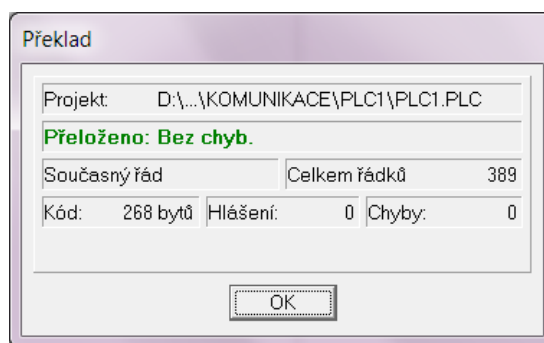


Obr. 38. Nastavenie parametrov sériovej komunikácie

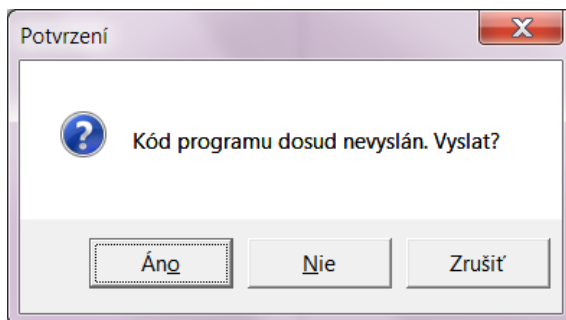
Pomocou kláves F9 a Ctrl+F9 program preložíme, nahráme do PLC a spustíme.



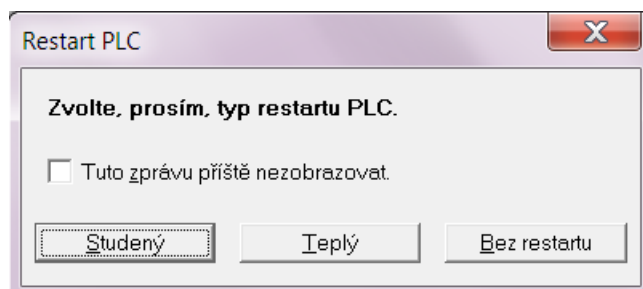
Obr. 39. Preklad a spustenie programu



Obr. 40. Oznámenie prekladu

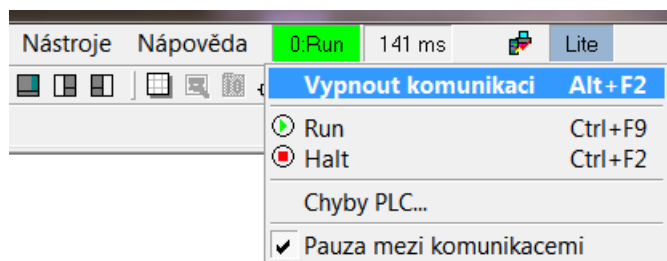


Obr. 41. Potvrdenie nahrávania



Obr. 42. Voľba typu reštartu

Pred začiatkom práce v prostredí Control Web 6 je nutné ukončiť komunikáciu s prostredím Mosaic.



Obr. 43. Ukončenie komunikácie

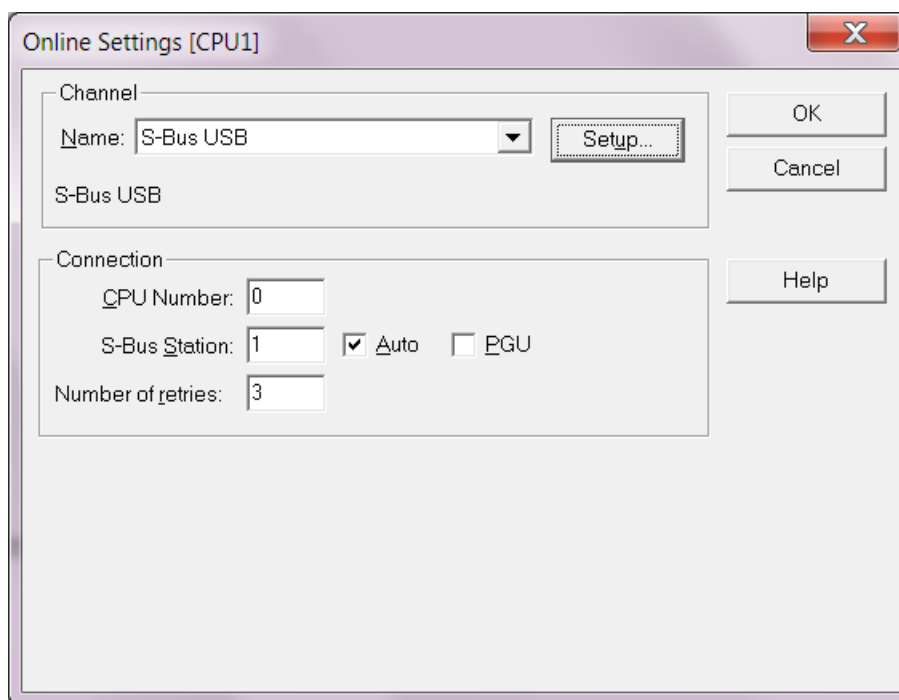
Po odpojení komunikácie je PLC pripravené pre ovládanie z prostredia Control Web 6. V ukázkovom programe *komunikace_Teco* je nutné dbať na správnu hodnotu komunikačného portu uvedenú v parametrickom súbore ovládača *Teco* za parametrom *ComDriver*. Číslo portu sa môže v jednotlivých prepojeniach líšiť.

PRÍLOHA P IV: KONFIGURÁCIA PLC SAIA.

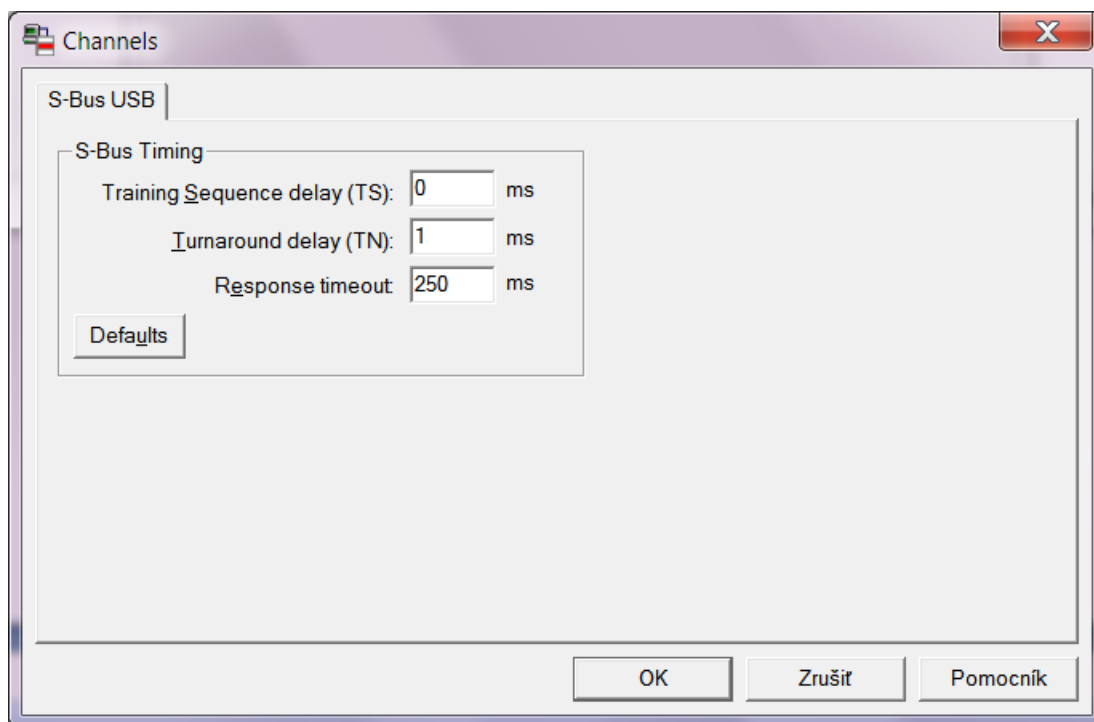
Súbor nastavení konfigurácie PLC SAIA pre prácu v prostredí Control Web 6 je totožný s popisom uvedeným v prílohe P II. S PLC SAIA môžeme komunikovať pomocou sériovej linky RS-232, USB alebo pomocou siete Ethernet. Vytvorená ukážková aplikácia *komunikace_saia* sa v jednotlivých prípadoch líši jedine nastaveniami uvedenými v parametrickom súbore.

Pri komunikácii pomocou sériovej linky RS-232 je potrebné presne uviesť číslo komunikačného portu v parametrickom súbore za parametrom *Device*. V konfigurácii nie je potrebné voliť ponuku protokolu TCP/IP.

Rozdiel v konfigurácii pre komunikáciu pomocou sériového rozhrania USB je vo voľbe *S-Bus USB* nastavení sekcie *Online*.



Obr. 44. Výber komunikácie sériovým rozhraním USB

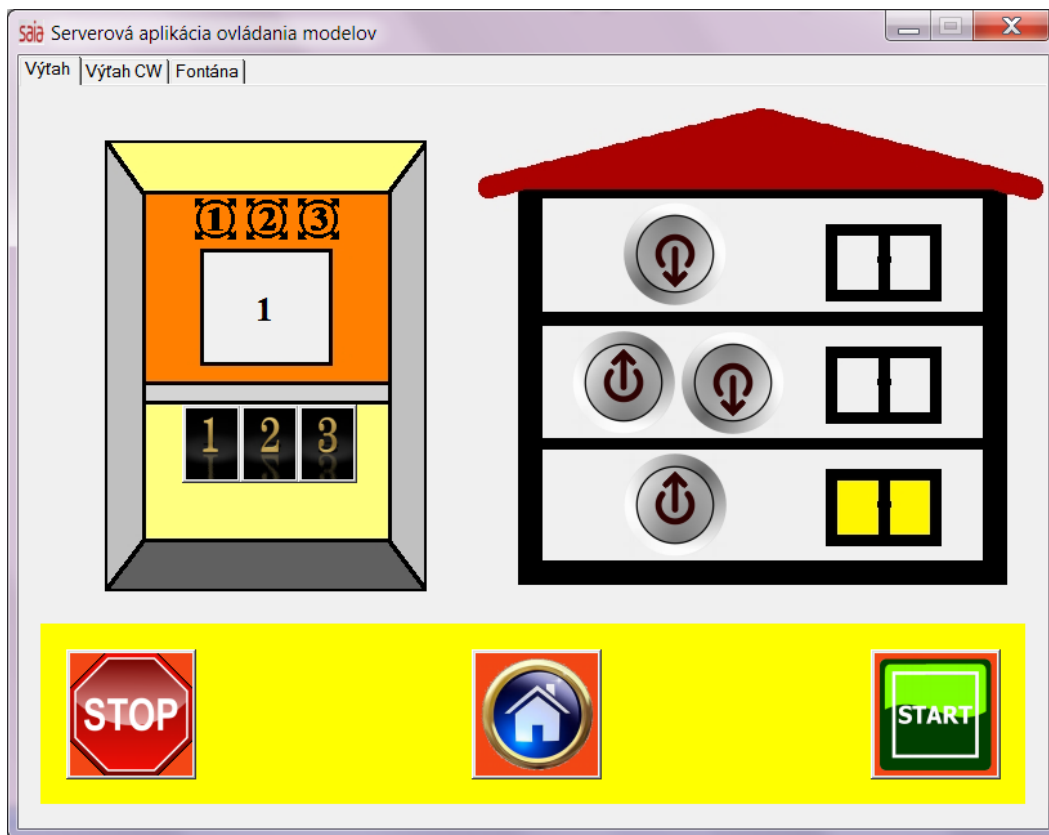


Obr. 45. Detail možností nastavení konfigurácie

V nastaveniach sekcie *Hardware* postačuje výber komunikácie protokolom S-Bus. Možnosti záložky *Serial* a voľba protokolu TCP/IP nie sú v tomto prípade potrebné.

Pri výbere komunikácie skrze sieť Ethernet prevedieme konfiguráciu presne podľa postupu uvedenom v prílohe P II.

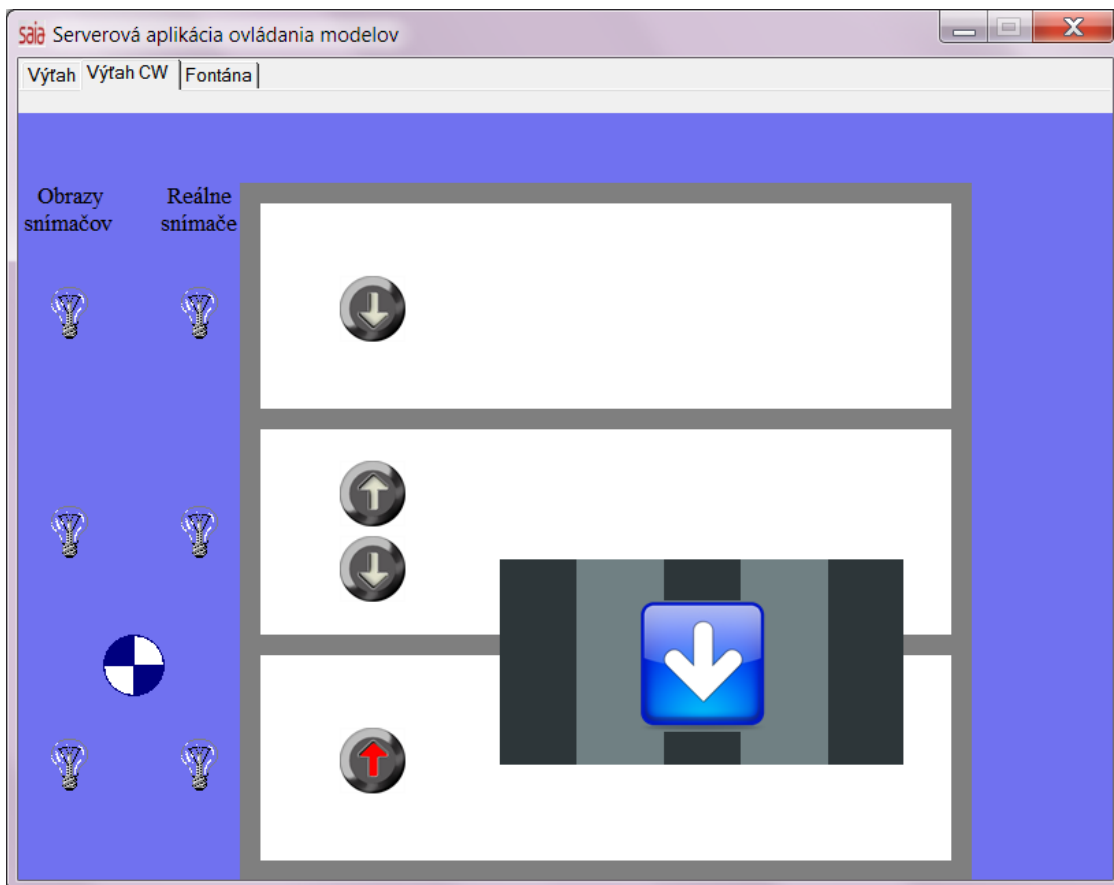
PRÍLOHA P V: UKÁŽKY APLIKÁCIÍ OVLÁDANIA MODELOV.



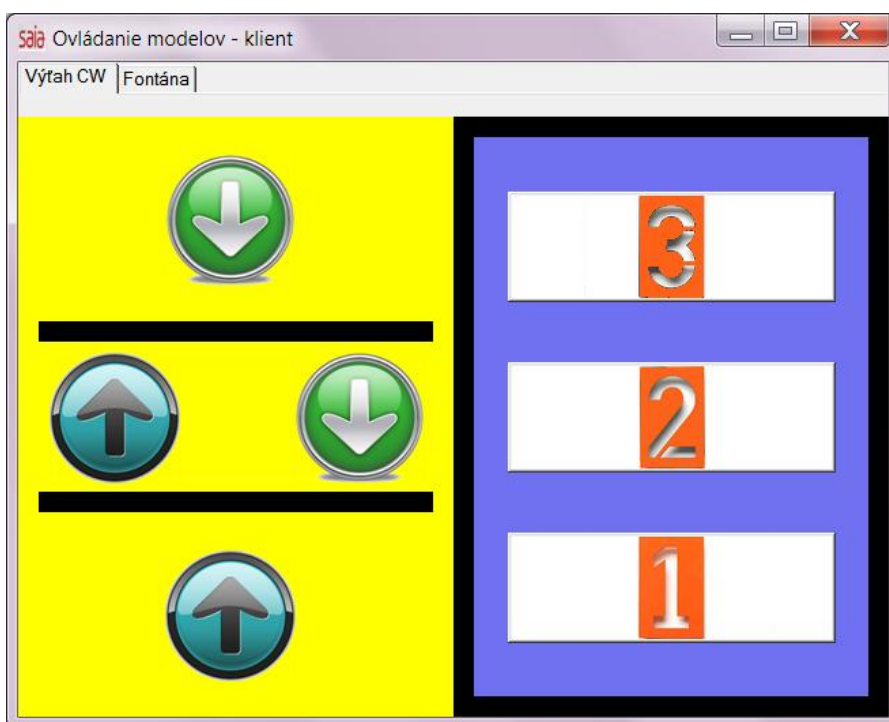
Obr. 46. Vizualizácia výťahu riadeného pomocou PLC SAIA



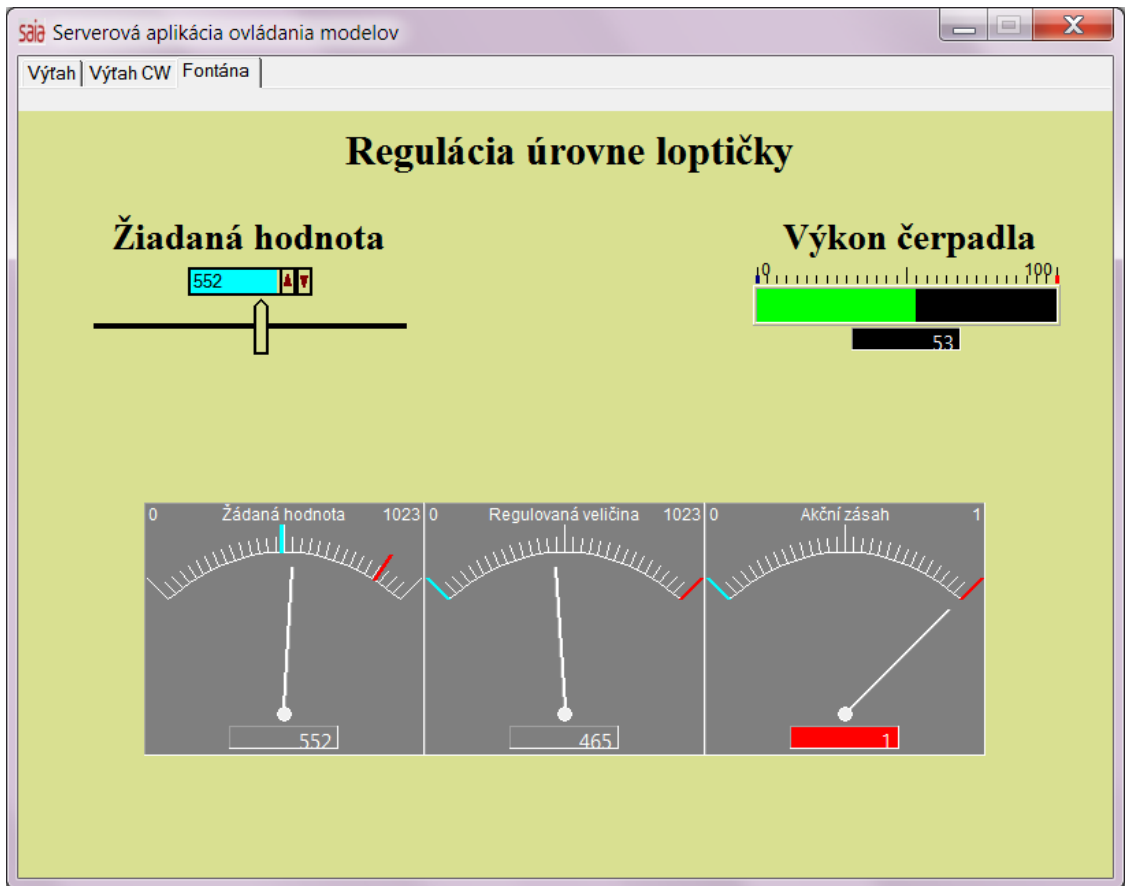
Obr. 47. Klient aplikácie riadenia výťahu



Obr. 48. Serverová aplikácia riadenia modelu výťahu systémom Control Web 6



Obr. 49. Klient riadenia modelu systémom Control Web 6



Obr. 50. Serverová aplikácia ovládania modelu fontány

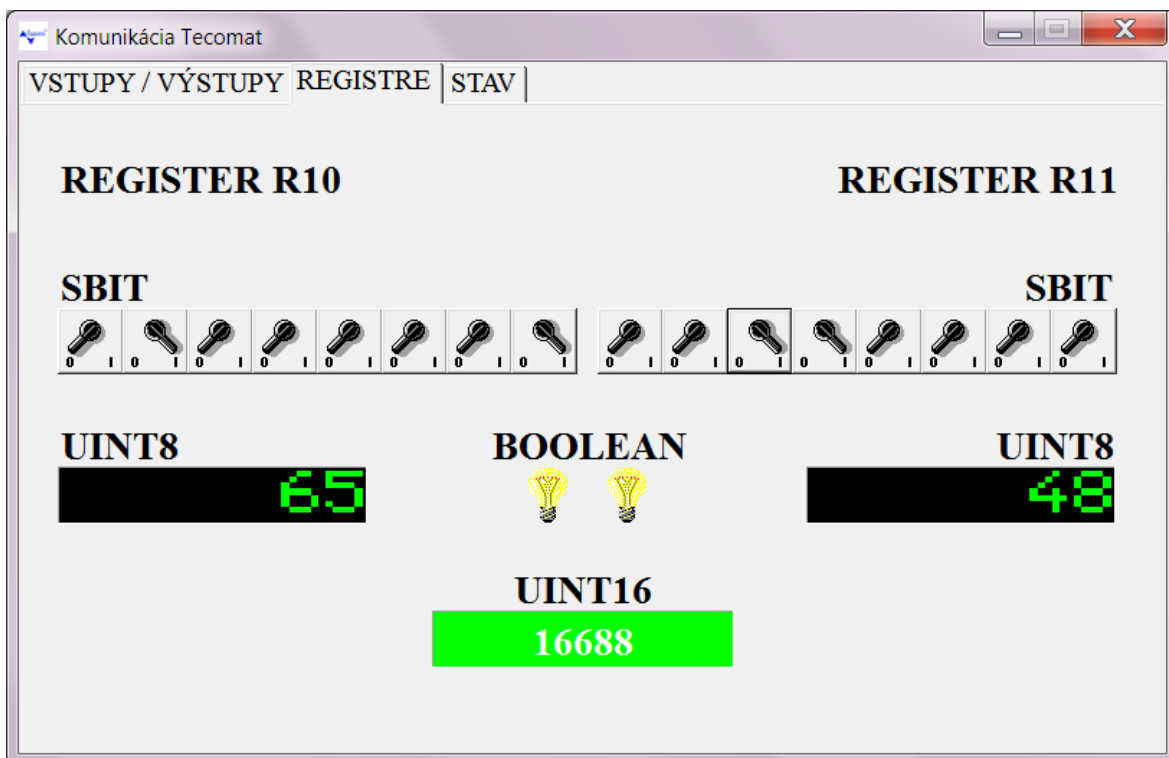


Obr. 51. Klient ovládania modelu fontány

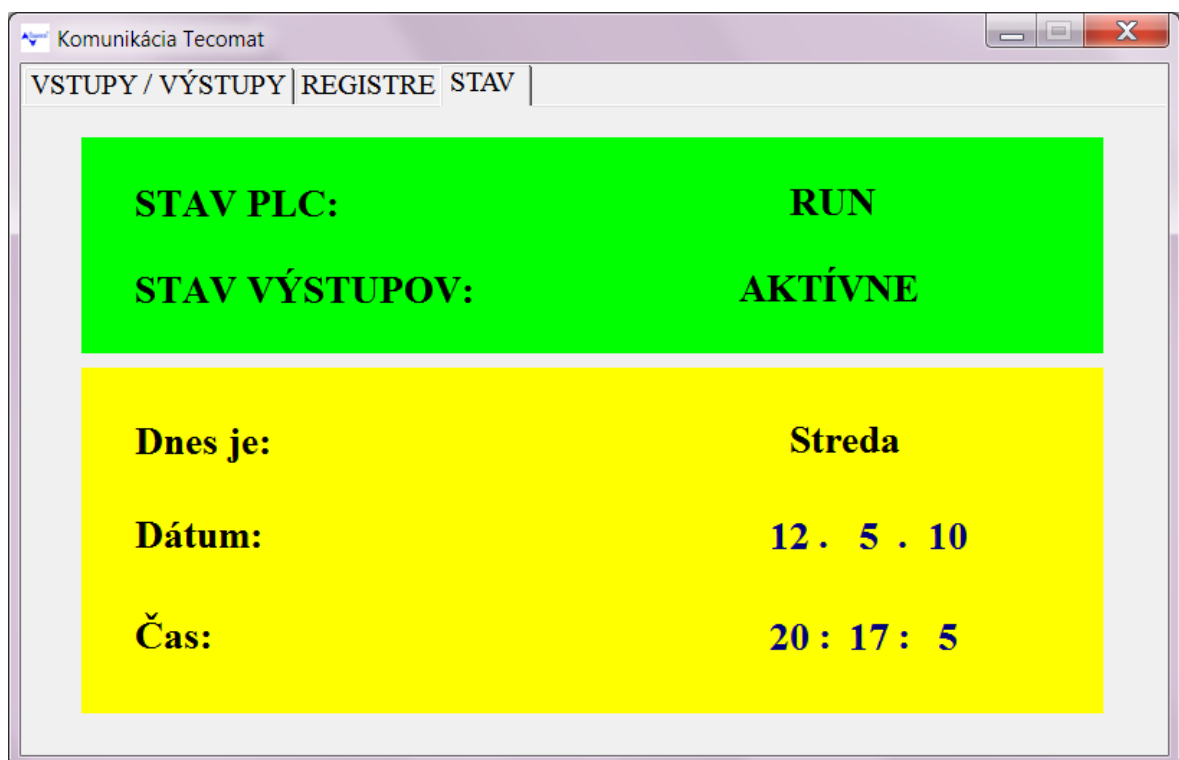
PRÍLOHA P VI: UKÁŽKA APLIKÁCIE KOMUNIKACE_TECO.



Obr. 52. Indikácia vstupov a voľby stavov výstupov

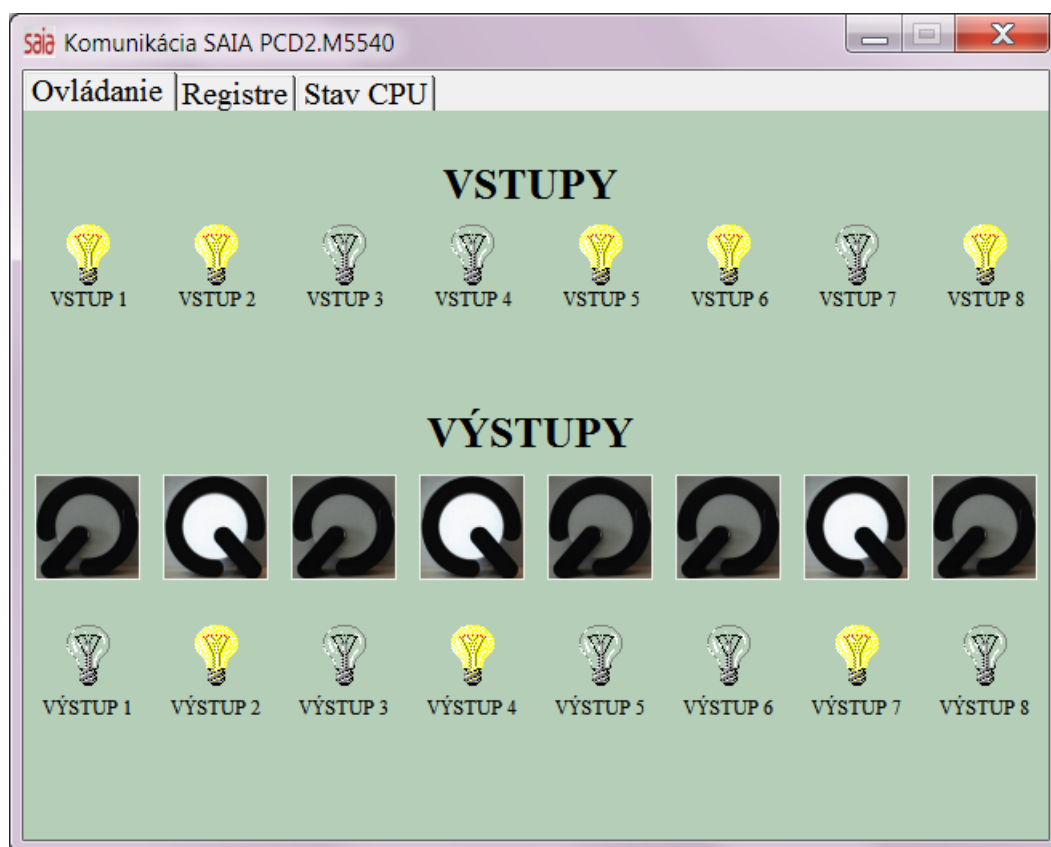


Obr. 53. Ukážka práce s registrami

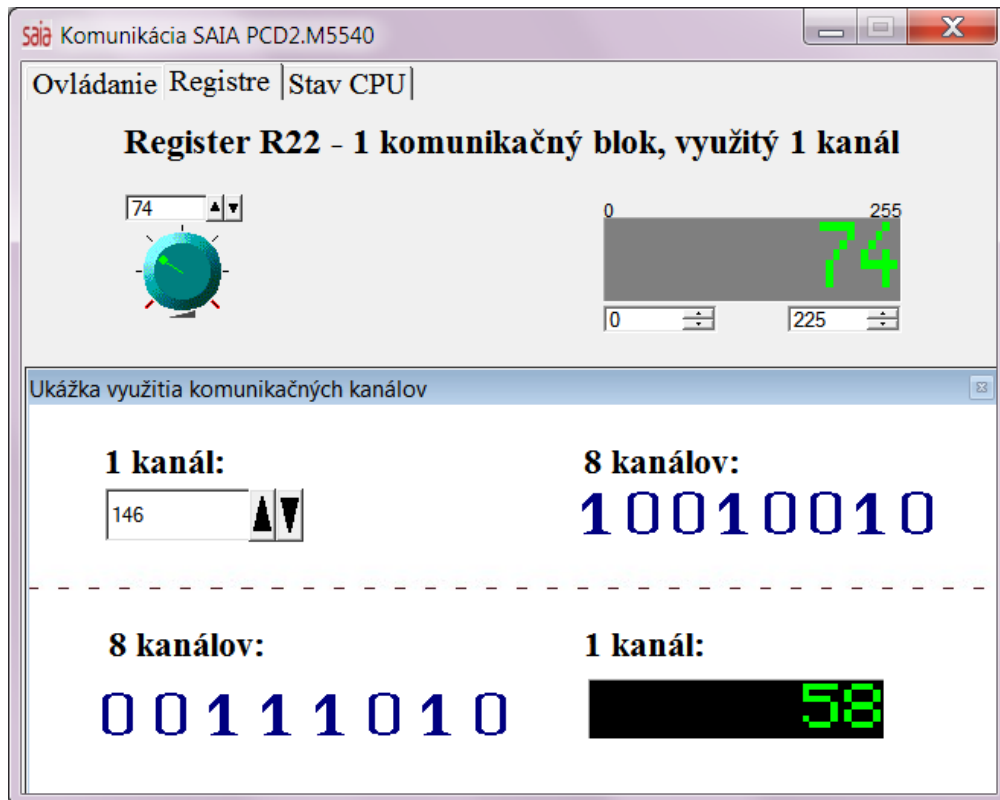


Obr. 54. Vyobrazenia stavov PLC, dátumu a času

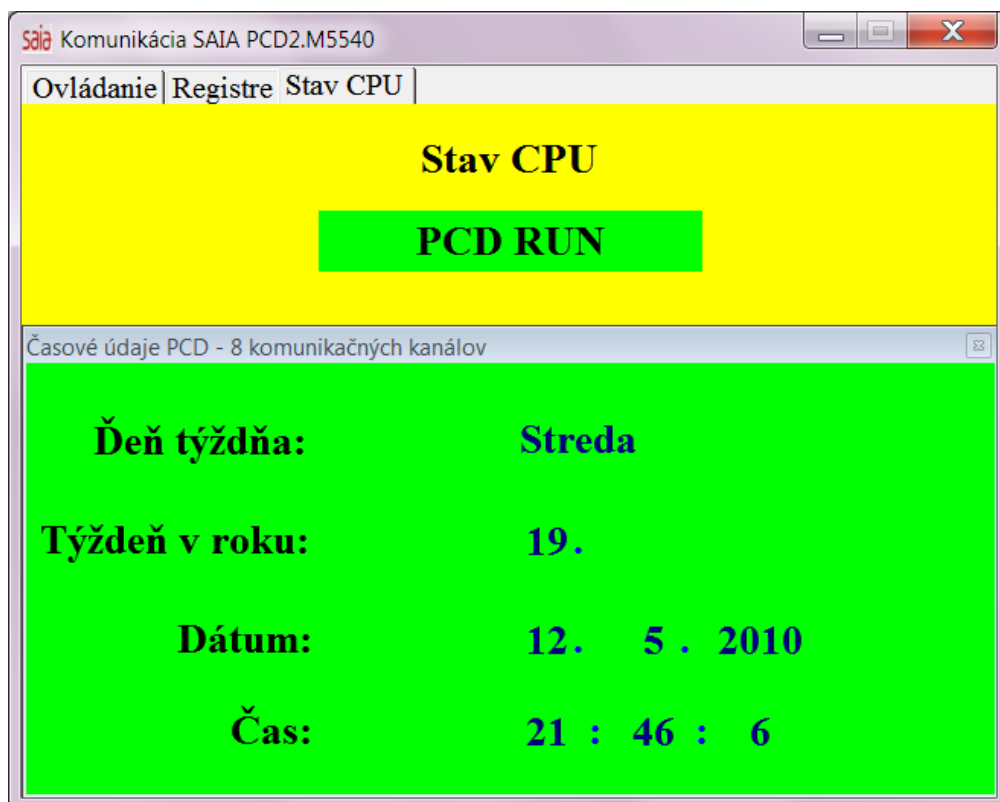
PRÍLOHA P VII: UKÁŽKA APLIKÁCIE KOMUNIKACE_SAIA.



Obr. 55. Indikácia vstupov a ovládanie výstupov PLC



Obr. 56. Ukážka práce s registrami



Obr. 57. Vyobrazenia niektorých údajov z PLC