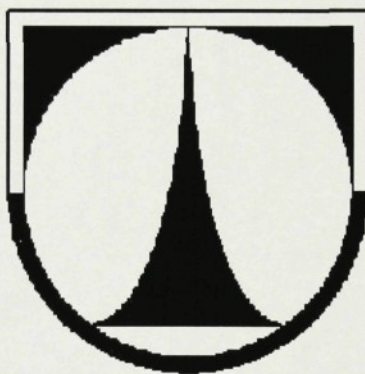


TECHNICKÁ UNIVERZITA V LIBERCI
FAKULTA STROJNÍ



DIPLOMOVÁ PRÁCE

TOMÁŠ RAJTR

Laboratorní informační a řídicí systém
The laboratory information and control system

LIBEREC 2001

TECHNICKÁ UNIVERZITA V LIBERCI
FAKULTA STROJNÍ
KATEDRA APLIKOVANÉ KYBERNETIKY

Obor 23 – 40 – 8

Automatizované systémy řízení ve strojírenství

Zaměření

Automatizace inženýrských prací

Laboratorní informační a řídicí systém

The laboratory information and control system

Tomáš Rajtr

Vedoucí diplomové práce: Ing. Slavomír Němeček

Konzultant diplomové práce: Prof. Ing. Miroslav Olehla, Csc.

Rozsah práce a přílohy:

Počet stran textu: 54

Počet obrázků: 26

Počet příloh: 5

Datum: 20.05.2001



TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta strojní

Katedra aplikované kybernetiky

Studijní rok: 2000/01

ZADÁNÍ DIPLOMOVÉ PRÁCE

Jméno a příjmení **Tomáš R a j t r**

obor **23 – 40 - 8 Automatizované systémy řízení ve strojírenství**

zaměření **automatizace inženýrských prací**

Ve smyslu zákona č. 111/1998 Sb. o vysokých školách se Vám určuje diplomová práce na téma:

Laboratorní informační a řídicí systém

Zásady pro vypracování:

(uveďte hlavní cíle diplomové práce a doporučené metody pro vypracování)

1. Seznamte se s problematikou laboratoře aplikované kybernetiky FS TU Liberec a navrhňte architekturu informačního a řídicího systému laboratorního procesu tak, aby byla respektována modulární koncepce, snadná rozšiřitelnost, přizpůsobitelnost nejrozličnějším nárokům a možnost ovládání z jednoho místa.
2. Navrhňte technické a programové prostředky pro:
 - síť katedry aplikované kybernetiky,
 - připojení na akademickou síť LIANE.
3. Navrhňte a ověřte systém sběru dat a řízení laboratorního procesu z jednotlivých operátorských stanic s využitím vizualizačního software TOMPACK v. od a.s. ProjektSoft Hradec Králové.
4. Navrhňte metodiku laboratorních cvičení z předmětu Základy aplikované kybernetiky s využitím tohoto informačního a řídicího systému.

+ CD
123/01 S

KKY/AIP-ASR
540, 60. příl., 1 vol. příl. - CD

Forma zpracování diplomové práce:

- průvodní zpráva cca 40 stran textu + přílohy
- grafické práce dle potřeby

Seznam literatury (uveďte doporučenou odbornou literaturu):

- [1] HANUŠ, B. : Teorie automatického řízení.
- [2] DORNER, J. : Návrh architektury informačního a radiaceho systému y hl'adiska aplikácii operačných systémov. (Sbomík konference Process Control '99, Tatranské Matliare).
- [3] MARTINÁSKOVÁ, M., ŠMEJKAL, L. : Řízení programovatelnými automaty
- [4] Popis a použití vizualizačního systému TOMPACK.

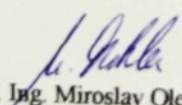
Vedoucí diplomové práce:

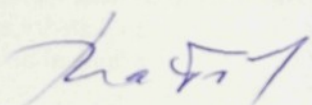
Ing. Slavomír Němeček

Konzultant diplomové práce:

Prof. Ing. Miroslav Olehla, CSc.,




Prof. Ing. Miroslav Olehla, CSc.
vedoucí katedry


Doc. Ing. Ludvík Prášil, CSc.
děkan

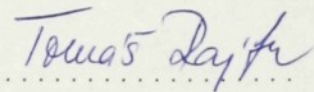
V Liberci dne 28. 02. 2001

Platnost zadání diplomové práce je 15 měsíců od výše uvedeného data. Termíny odevzdání diplomové práce jsou určeny pro každý studijní rok a jsou uvedeny v harmonogramu výuky.

MÍSTOPŘÍSEŽNÉ PROHLÁŠENÍ:

Prohlašuji, že jsem diplomovou práci vypracoval samostatně s použitím uvedené literatury pod vedením vedoucího a konzultanta.

V Liberci dne 20.05.2001

A handwritten signature in blue ink, reading "Tomáš Rajtr", written over a dotted line.

Tomáš Rajtr

Anotace:

Tato diplomová práce se zabývá problematikou laboratorních informačních a řídicích systémů, v našem případě modernizací laboratoře na katedře aplikované kybernetiky. V úvodu se nejprve seznámíme s nutností podrobného zkoumání prostředí, kde budeme náš systém provozovat, aby byla respektována modulární koncepce, snadná rozšiřitelnost, přizpůsobitelnost nejrůznějším nárokům a možnost ovládání z jednoho místa a jeho praktická aplikovatelnost v laboratoři. Ve druhé části se seznámíme s řešením technických a programových prostředků pro využití systému v síťovém provedení. Další část dává konkrétní pohled na systém sběru dat a řízení laboratorního procesu při řešení jednoduchých laboratorních úloh. V závěrečné části práce je uveden způsob využití navrženého systému při výuce předmětu Základy aplikované kybernetiky s použitím vizualizačního prostředí TomPack.

Annotation:

This dissertation deals with the problem areas concerning laboratory and control system with regard to the specific case of the modernization of the laboratory at the Faculty of Applied Cybernetics. The introduction examines the necessity of detailed research into the environment, where the system is to be operated, in order to respect the system's modular conception, its ease expandability, its adaptability to various requirements, the option of controlling the system from a single point and its practical application within the laboratory. The second section deals with the solutions for the technical and programming environments for the use of the system in its network form. The next section provides a specific view of the data collection system and the control of the laboratory processes during the resolution of the individual laboratory tasks. The final section of the dissertation sets out the methods for using the proposed system during the teaching of the Principles of Applied Cybernetics using the TomPack visual environment.

OBSAH:

1.	ÚVOD.....	9
2.	PŘÍPRAVA AUTOMATICKÉHO ŘÍZENÍ TECHNOLOGICKÉHO PROCESU ...	9
2.1	Identifikace systému	9
2.2	Simulace systému.....	10
3.	DRUHY A ZPŮSOB AUTOMATICKÉHO ŘÍZENÍ	11
3.1	Řízení v otevřeném obvodu.....	11
3.2	Řízení v uzavřeném obvodu	12
4.	LABORATOŘ NA KKY	14
5.	LOKÁLNÍ POČÍTAČOVÁ SÍŤ	15
5.1	Topologie LAN	16
5.1.1	Sběrnice	17
5.1.2	Hvězda	18
5.1.3	Strom (hvězdice).....	18
5.1.4	Kruh	19
5.2	Přenosová média	19
5.2.1	Koaxiální kabel	20
5.2.2	Kroucená dvoulinka	21
5.2.3	Optické kabely	23
5.3	Architektura komunikačních funkcí.....	23
5.3.1	Architektura ISO OSI	24
5.3.2	Architektura TCP/IP	25
5.4.	Návrh počítačové sítě v laboratoři KKY	27
5.4.1	Vyjádření potřeb pro realizaci laboratoře.....	28
5.4.2	Připojení laboratoře do akademické sítě LIANE	28
6.	VIZUALIZACE TECHNOLOGICKÉHO PROCESU	29
6.1	Stručná charakteristika dostupných vizualizačních systémů.....	30
6.1.1	Control Panel.....	30
6.1.2	Trend Accord	30
6.1.3	Citect.....	31
6.1.4	TomPack	32
6.2	Vizualizační software TomPack	32
6.2.1	Jednotlivé komponenty systému	33
6.2.1.1	DDE server	34
6.2.1.2	TPServer	36

6.2.1.3	TPView	37
6.2.1.4	TPConfig	37
6.2.2	Možné konfigurace pro realizaci vizualizace.....	38
6.2.2.1	Triviální konfigurace	38
6.2.2.2	Složitější konfigurace	39
6.2.3	Typy proměnných v TomPacku.....	39
6.2.3.1	Složky proměnných	40
7.	TVORBA APLIKACE V PROSTŘEDÍ TOMPACK.....	42
7.1	Vizualizační SW vyvinutý pro laboratoř na KKY	42
7.2	Matematický popis použitých soustav	47
8.	NÁVRH VÝUKY S VYUŽITÍM VYVINUTÉ APLIKACE	48
9.	ZÁVĚR	51
	LITERATURA:	52
	SEZNAM OBRÁZKŮ :	53
	SEZNAM TABULEK :	53
	SEZNAM GRAFŮ :	54
	SEZNAM PŘÍLOH :	54

SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ:

PC	-	Personal Computer
LAN	-	Local Area Network
SW	-	software
HW	-	hardware
PLC	-	Programmable Logic Controller
DDE	-	Dynamic Data Exchange
SCADA	-	Supervisory Control And Data Acquisition

1. ÚVOD

Řízení technologických procesů a zařízení je jednou částí širší oblasti řízení výrobních procesů. Jestliže je proces nebo zařízení automatizováno, má ucelenou koncepci, vyjasněné cíle a návaznosti mezi jednotlivými částmi, mluvíme o automatizovaném řízení technologických procesů, resp. zařízení. Počítače a další automatizační prostředky mohou pracovat s cílem buď koordinovat, kontrolovat, eventuálně ovlivňovat (v tzv. režimu „ON-LINE“), nebo mohou bezprostředně řídit průběh procesů (v tzv. režimu „IN-LINE“). V obou případech jde o řízení automatické, v němž člověku přísluší jen úloha dohlížecí a rozhodovací [1].

Zavedením automatizace se odstraňují chyby způsobené nepozorností nebo nedbalostí obsluhy, a tím se výrazně zvýší produktivita a kvalita výroby. Před zavedením automatizace do praxe musí dojít k pečlivé analýze problému, která by nám měla stanovit podmínky, při kterých zavedením automatizace do provozu dosáhneme stanovených požadavků.

2. PŘÍPRAVA AUTOMATICKÉHO ŘÍZENÍ TECHNOLOGICKÉHO PROCESU

V přípravě pro automatické řízení technologického procesu je nutné nejprve provést několik základních operací, kterými získáme potřebné informace nezbytně nutné pro správné nastavení celého systému automatického řízení před jeho spuštěním.

2.1 *Identifikace systému*

Identifikací soustavy se rozumí zjištění vlastností této soustavy. Výsledkem identifikace jsou pak obvykle statické a dynamické charakteristiky soustavy, matematický či logický popis chování soustavy apod. Na základě výsledků identifikace je možno stanovit model soustavy nebo celého zařízení a řešit potřebné problémy mimo vlastní zařízení. Je možno vyzkoušet všechny možné varianty provozu bez nákladného a často i nebezpečného experimentování a vybrat z hlediska technického a ekonomického nejvýhodnější variantu řešení. Např. výběr vhodného regulátoru a správné nastavení jeho parametrů závisí na chování regulované soustavy [2].

Předběžné určení chování (na základě výpočtů a konstrukčních a projektových dat) nebývá pro účely optimalizace a automatizace dostatečně přesné a proto je nutné zjistit vlastnosti soustavy přesněji identifikací. Vlastnosti soustavy se však mnohdy v čase mění v důsledku změn vnějších podmínek (změny zatížení, jakosti vstupního zařízení, atd.). V těchto případech je třeba identifikaci provádět opakovaně nebo i nepřetržitě. Hovoříme proto o identifikaci jednorázové (off – line) a o identifikaci průběžné (on – line).

➤ ***jednorázová identifikace (off – line)***

Nejprve se provádí celé měření, potom jejich vyhodnocení a nakonec následuje využití výsledků.

➤ ***průběžná identifikace (on – line)***

Měření, vyhodnocování i využívání všech výsledků se provádí současně.

Různé vypracované metody identifikace jsou pak vhodné pro použití on – line nebo off – line.

2.2 Simulace systému

Simulací systému rozumíme realizaci modelu systému na základě jeho popisu (získaného například identifikací). Provádí se buď na fyzikálních modelech, vybudovaných pomocí fyzikálního zařízení (např. modely letadel, vodních přehrad) nebo na modelech matematických (např. na analogových či číslicových počítačích).

Simulace systému se používá nejčastěji pro:

- Syntézu systému v případě, že originál jako celek ještě neexistuje (je zatím ve stádiu projektování). Účelem je pak stanovit nejvhodnější chování a strukturu systému podle zadaných cílů. Na základě výsledků simulace se originál navrhuje.
- Analýzu systému v případě, že originál již existuje. Účelem je zjistit chování za nejrůznějších situací, které není žádoucí experimentálně ověřovat přímo na originálu (např. havarijní situace, odstavování, ...). Cílem je zajištění bezpečnosti originálu a jeho optimalizace.
- Identifikaci systému v případě, že originál již existuje. Srovnává se chování originálu a modelu za určitých situací a podle výsledku se model koriguje. Účelem je získat další vyhovující model systému.

- Imitaci systému v případě, že originál sice existuje, ale jeho použití pro daný účel by nebylo vhodné (např. bezpečnostní nebo ekonomické hledisko). Jedná se obvykle o trenažéry nejrůznějších typů.

3. DRUHY A ZPŮSOB AUTOMATICKÉHO ŘÍZENÍ

Přes velkou rozmanitost úloh automatického řízení lze najít některá obecná kritéria, podle nichž můžeme automatické řídicí systémy klasifikovat.

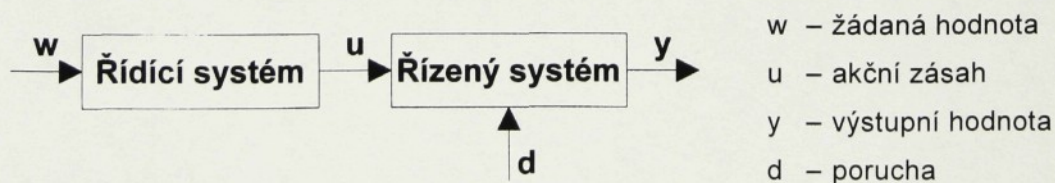
Úlohy, které lze formulovat jako dodržování parametrů procesu na žádané hodnotě, bez ohledu na změny zatížení systému, se řeší pomocí prostředků **automatické regulace**. Přitom není podstatné, zda je žádaná hodnota řízeného parametru konstantní nebo zda se mění podle předem známého časového programu.

V minulosti byla převážná část automatických systémů realizována klasickými technickými prostředky – automatickými regulátory a řídicími automaty, resp. jejich kombinacemi. V dnešní době jsou tyto prostředky nahrazovány mikroprocesory či řídicími počítači, avšak charakter principu regulace se nemění.

Řídicí systémy mohou pracovat buď v otevřeném nebo uzavřeném obvodu řízení. Při řízení v otevřeném obvodu se k řízení využívá jen apriorní informace o řízeném objektu a nekontroluje se jeho skutečný stav. Takovému způsobu řízení říkáme **ovládání** (může být ruční nebo automatické). Při řízení v uzavřeném obvodu se vedle apriorní informace zpracovává také informace o stavu řízeného objektu. Takové řízení se jmenuje **regulace**.

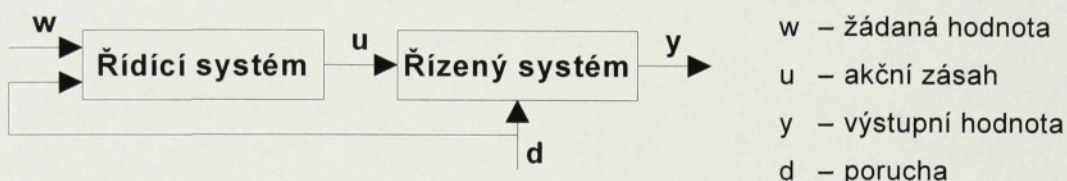
3.1 Řízení v otevřeném obvodu

Systémy řízení v otevřeném obvodu (ovládání) můžeme použít tam, kde vnitřní stav systému a okolní vlivy jsou konstantní nebo se mění podle předem daného průběhu (obr. 1). Na systém působí náhodné vnější vlivy **d** (poruchy). Pokud je vliv poruch malý, bude se výstupní veličina měnit v souladu s řídicí veličinou **w**.



Obr. 1: Řízení v otevřeném obvodu

Jestliže je systém vystaven silným vnějším vlivům, může se použít řízení s měřením poruchy (obr. 2). Řídicí systém zpracovává jak řídicí veličinu w , tak průběh poruchy d . Uvedené blokové schéma řízení však není schopno eliminovat změny vnitřního stavu systému, ani není schopno řídit systém nestabilní.

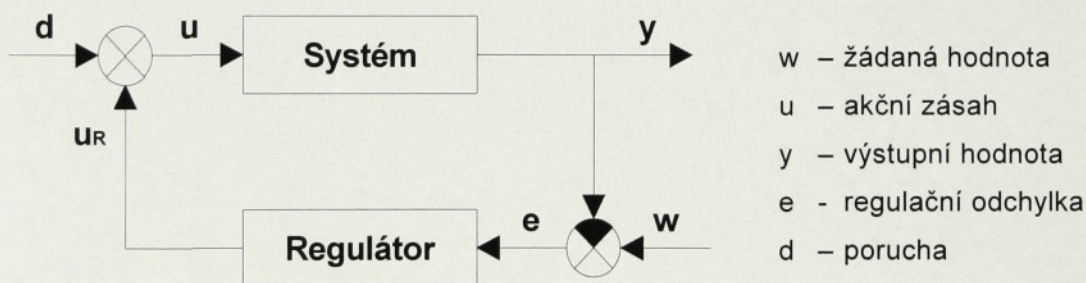


Obr. 2: Řízení v otevřeném obvodu s měřením poruchy

Otevřené obvody řízení se používají např. při řízení tvářecích strojů, jednoduchých kopírovacích strojů, pro ovládání signálních světel (semaforů), atd.

3.2 Řízení v uzavřeném obvodu

Systémy řízení v uzavřeném obvodu (regulace) se od předcházejících liší tím, že k řízení využívají principu zpětné vazby (obr. 3).



Obr. 3: Řízení v uzavřeném obvodu

Řídicí systém dostává informace o žádané hodnotě výstupního signálu a porovnává je s dosaženým výsledkem činnosti, tj. se skutečnou hodnotou y . Jestliže existuje regulační odchylka $e = w - y$, zasahuje akční veličinou u do řízeného systému tak, aby odchylku odstranil. Poruchy d mohou působit jak v místě působení akční veličiny, tak i v jiných místech řízeného systému. V teorii regulace se řídicí systém nazývá regulátor, řízený systém regulovaná soustava. Soustava s regulátorem tvoří (uzavřený) regulační obvod.

$$Y(p) = G_S(p) * U(p) \quad /3.2.1/$$

$$U(p) = D(p) + U_R(p) \quad /3.2.2/$$

$$U_R(p) = G_R(p) * E(p) \quad /3.2.3/$$

$$E(p) = W(p) - Y(p) \quad /3.2.4/$$

Z rovnic /3.2.1/ až /3.2.4/ získáme rovnici výslednou, která zahrnuje regulovanou hodnotu závislou na vnější poruše a žádané hodnotě.

$$Y(p) * [1 + G_S(p) * G_R(p)] = G_S(p) * D(p) + G_S(p) * G_R(p) * W(p) \quad /3.2.5/$$

Podle obrázku 3 můžeme snadno definovat dynamické vlastnosti regulované soustavy, tj. přenos soustavy $G_S(p)$ a přenos poruchy soustavy $G_{Sdi}(p)$, vyjádřené v oblasti komplexní proměnné (v prostoru obrazů)

➤ **přenos regulované soustavy**

$$G_S(p) = \frac{Y(p)}{U(p)} \text{ za předpokladu, že } d_i(t) = 0, i = 1 \dots n, \text{ potom } u_R(t) = u(t)$$

/ 3.2.6 /

➤ **přenos poruchy regulované soustavy**

$$G_{Sdi}(p) = \frac{Y(p)}{D_i(p)}, i = 1 \dots n, \text{ za předpokladu, že } u_R(t) = 0 \quad / 3.2.7 /$$

Existuje celá řada modifikací schémat regulačních obvodů, které zajišťují vyšší kvalitu řízení i při velkých vnějších poruchách.

V dnešní době se automatizace vyskytuje téměř ve všech oborech, jako jsou např. průmyslové technologie, zdravotnictví či energetika.

4. LABORATOŘ NA KKY

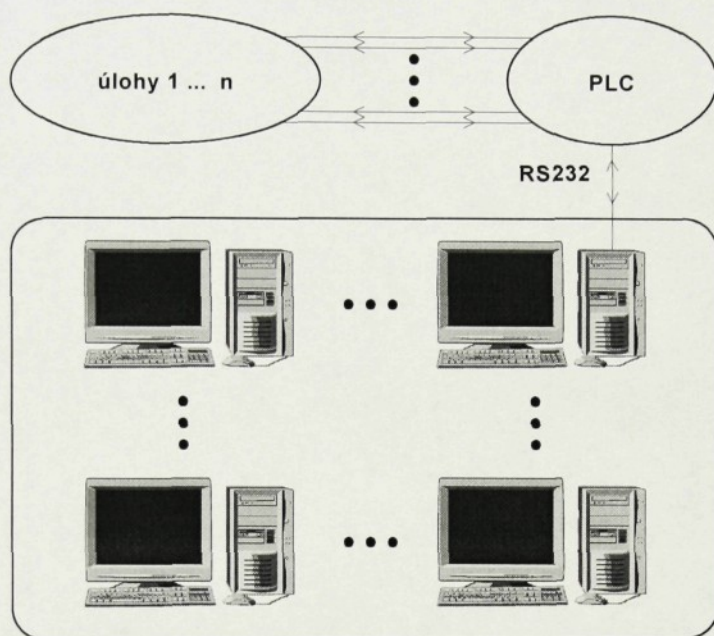
Laboratoř na katedře aplikované kybernetiky Technické univerzity v Liberci slouží kromě jiného také k výuce předmětu „Základy aplikované kybernetiky“ pro posluchače II. ročníku fakulty strojní. V tomto předmětu se studenti seznamují (jak název předmětu napovídá) se základy aplikované kybernetiky.

Při návrhu modernizace celé laboratoře musíme mít na paměti spoustu aspektů, které jsou méně či více důležité. Asi nejdůležitějším aspektem při návrhu laboratoře, je snadná modifikace a rozšiřitelnost podle aktuálních požadavků výuky a možností katedry. V současné době jsou k dispozici 3 úlohy, v dohledné době se plánují úlohy další.

Z technického hlediska návrhu struktury laboratoře je jedno, zda automatické řízení bude prováděno pomocí PLC nebo jiným způsobem. Struktura laboratoře bude vždy dělena do následujících sekcí:

- vlastní úloha,
- přenos dat mezi úlohou a zpracováním,
- počítačová síť,
- zpracování a vizualizace přenesených dat.

Podrobný rozbor a návrh technických prostředků pro vlastní realizaci automatizace úloh a pro přenos dat mezi úlohou a PC, zpracovává student Jindřich Diesner ve své diplomové práci – Využití moderních technických prostředků a programových prostředků pro řízení laboratorního procesu, Liberec 2001. Při mém návrhu síťového prostředí a sběru dat a jejich realizaci – vizualizaci, využívám PLC, které komunikuje s vizualizačním serverem po sériové lince RS232. Návrh struktury laboratoře v blokovém vyobrazení je na obr. 4.



Obr. 4: Návrh struktury laboratoře na KKY

5. LOKÁLNÍ POČÍTAČOVÁ SÍŤ

Pojmem *lokální počítačová síť* (dále jen LAN) zpravidla označujeme komunikační systém schopný propojit desítky až stovky počítačů na vzdálenost stovek metrů až jednotek kilometrů. Síť LAN je vlastně několik počítačů (standardu PC) navzájem propojených takovým způsobem, aby byla umožněna jejich vzájemná komunikace. Takováto síť pak může poskytnout uživatelům uvedené služby:

- sdílení technických prostředků sítě (HW),
- sdílení dat,
- elektronická pošta (E-MAIL),
- monitorování připojených uživatelů,
- obrazová (hlasová) komunikace v rámci sítě.

Není však možné vycházet z předpokladu, že všechny typy sítí musejí poskytovat všechny (výše uvedené) služby. Jednotlivé typy sítí se běžně liší (mimo jiné) právě v množství a kvalitě poskytovaných služeb. Dá se říci, že první dva uvedené druhy služeb poskytují všechny typy sítí [3].

LAN je typická následujícími znaky:

- přenosové médium (kabeláž) je ve vlastnictví uživatele (umístěno v uživatelském objektu),
- pro přenos se používají poměrně vysoké přenosové rychlosti,
- mají relativně nízkou chybovost, která vyplývá z používání poměrně kvalitních kabelů a ze skutečnosti, že síť neprochází otevřeným prostorem s výraznějším rušením.

Sítě LAN se uplatňují především v dále uvedených aplikacích:

➤ ***integrující prostředí***

V těchto aplikacích sítě LAN propojují heterogenní prostředky výpočetní techniky. Jejich pomocí jsou tedy propojené jednotlivé počítače, střediskové počítače, případně mohou být tyto prostředky propojené pomocí LAN na veřejnou datovou síť.

➤ ***informační systémy s integrovanými službami***

Sítě těchto aplikací poskytují svým uživatelům celou řadu služeb. Většinou se jedná o telex, teletext, videotext a přístup na veřejnou datovou síť.

➤ ***místní sítě v průmyslových aplikacích***

V průmyslu se sítě LAN nejvíce uplatňují při řízení nejrůznějších technologických procesů a spojují jednotlivé výrobní procesy s řídicím střediskem.

➤ ***místní sítě pro spojení osobních počítačů***

V této oblasti došlo k největšímu nasazení a rozvoji sítí LAN. Sítě propojují jednotlivé počítače (PC) za účelem sdílení dat a technických prostředků (HD, tiskárny, CD-ROM, ...).

5.1 Topologie LAN

Topologie sítě je způsob vzájemného propojení uživatelů sítě, kterým se LAN liší od rozsáhlých počítačových sítí. Volba topologie má vliv na řadu vlastností lokální sítě:

➤ ***rozšiřitelnost***

Možnost a snadnost doplňování stanic do existující sítě.

➤ ***rekonfigurovatelnost***

Možnost modifikovat síť při závadě komponenty.

➤ **spolehlivost**

Odolnost sítě proti výpadkům komponent.

➤ **složitost obsluhy**

➤ **výkonnost**

Využití přenosové kapacity média, zpoždění zprávy.

Propojení může být dvoubodové nebo na sdíleném kanálu. Některé sítě jednotlivé topologie kombinují (např. ARCNet nebo dnešní Ethernet). U sítí PC-LAN (sít' LAN na PC) se nejčastěji setkáváme s následujícími topologiemi:

- sběrnice,
- hvězda,
- strom (hvězdice),
- kruh.

Uvedené dělení sítí na sběrnice, hvězdové, stromové a kruhové je opřeno o *elektrickou topologii* (signálovou topologii), tedy o způsob vzájemného propojení stanic. Z hlediska vlastností sítě má velký vliv i *topologie fyzická* (způsob vedení kabelů) a *topologie logická* (metoda spolupráce stanic u deterministických metod).

5.1.1 Sběrnice

Základním prvkem sběrnice je úsek přenosového média – *segment* sběrnice, ke kterému jsou připojeny stanice sítě. Přenosovým médiem je nejčastěji koaxiální kabel nebo symetrické vedení (kroucená dvoulinka). Vlastnosti sběrnice sítě lze shrnout do těchto bodů:

- pasivní sdílené médium,
- snadné připojování stanic,
- odolnost proti výpadkům stanic,
- sít' padá s výpadkem sběrnice (při porušení sběrnice mezi stanicemi dochází ke vzniku dvou nekomunikujících sítí).

Pro řízení sběrnice je využívána řada deterministických i nedeterministických metod, které využívají faktu, že signál vysílaný jednou stanicí je přijímán ostatními stanicemi s velmi malým zpožděním.

5.1.2 Hvězda

Stanice sítě jsou připojeny k centrálnímu uzlu samostatnými linkami. Centrální uzel, označovaný jako *hub* (v překladu „střed loukoťového kola“), signál přicházející z jedné linky rozděluje do ostatních linek hvězdy. Rozlišujeme *pasivní hub*, ve kterém je signál pouze dělen (odporovým děličem), a *aktivní hub* (vícestupňový opakovač), ve kterém je přijatý signál upravován tak, aby měl na výstupních linkách požadovanou úroveň a časování. Vlastnosti topologie „hvězda“ lze shrnout takto:

- dvoubodové spoje mezi stanicemi a centrálním uzlem lze snadno realizovat,
- síť je odolná proti výpadku jednotlivých stanic a linek,
- síť je citlivá na poruchu centrálního uzlu.
- větší koncentrace kabeláže v okolí centrálního uzlu.

5.1.3 Strom (hvězdice)

Stromová topologie je přirozeným rozšířením topologie typu „hvězda“. Setkáváme se s ní u širokopásmových sítí a u sítí využívajících pro přenos světlovody. Vlastnosti stromové topologie jsou podobné jako u sítí typu „hvězda“:

- odolnost sítě proti výpadkům jednotlivých stanic a linek,
- citlivost na výpadky uzlů (hubů),
- snadná rozšiřitelnost,
- dvoubodové spoje.

Jedná se o zobecněnou hvězdu, kdy jednotlivé počítače jsou propojeny pomocí tzv. rozvětvovačů (hubů). Výhodou této sítě je jednoduché rozšiřování a poměrně veliký dosah. Důsledky výpadku sítě, způsobené výpadkem kabelu nebo hubu, nelze předem určit. Záleží na místě poruchy. Otázka výkonu sítě závisí na poloze serveru v topologii.

Stromové (hvězdicové) sítě používají podobných metod řízení jako sítě sběrníkové.

5.1.4 Kruh

U kruhových sítí jsou komunikační stanice propojeny spoji, které jsou využívány pouze jednosměrně. Signál vyslaný jednou stanicí je postupně předáván ostatním stanicím kruhu (základním prvkem stanice je krátký posuvný registr) a po oběhu sítí se vrací ke stanici, která jej odeslala. Vlastnosti kruhových sítí lze shrnout do těchto bodů:

- dvoubodové jednosměrné spoje lze snadno realizovat i na světlovodech,
- v síti lze kombinovat různá média (pro krátké spoje elektrická vedení, pro dlouhé spoje světlovody),
- síť je citlivá na výpadek libovolného prvku (stanice nebo spoje).

U kruhových sítí jsou pravidelně používány deterministické metody řízení.

5.2 Přenosová média

Jednou z odlišností lokálních a rozlehlých počítačových sítí je vztah jejich provozovatelů k prostředkům, které slouží k propojování uzlových počítačů. V případě rozlehlých sítí jde vesměs o přenosové cesty, resp. okruhy či kanály, pouze pronajaté od spojové organizace, která je jejich vlastníkem. V případě lokálních počítačových sítí si nezbytné prostředky propojení zřizují provozovatelé sítě sami, a tyto pak zůstávají jejich majetkem.

Kromě malého počtu historických sítí, které používaly paralelní přenos po vícevodičových kabelech (např. sběrníková síť Cluster One nebo kruhová síť Twentenet), jde u většiny dnešních sítí o přenos sériový. Nejčastěji se setkáváme s nesymetrickým (*koaxiální kabel*) a symetrickým (*kroucená dvoulinka – twisted pair*) vedením. Řada sítí se opírá o optická vlákna a ta jsou alternativním médiem i pro klasické technologie. Začínají se objevovat LAN využívající vysokofrekvenčních rádiových a vzdušných světelných spojů [4].

Možnosti využití jednotlivých druhů kabelů jsou určeny především jejich kvantitativními parametry. K nejdůležitějším patří:

- **přenosová rychlost**

Rychlost, které lze na daném kabelu dosáhnout.

- **útlum**

Představuje zeslabení přenášeného signálu. Je způsoben odporem, který kabel klade přenášenému signálu – bývá větší pro vyšší

frekvence přenášeného signálu (a tedy pro vyšší přenosové rychlosti), a roste také se zmenšováním průměru kabelu. Celková hodnota útlumu je přímo úměrná délce kabelu a je jedním z rozhodujících faktorů, které určují maximální použitelnou délku souvislého úseku (segmentu) kabelu.

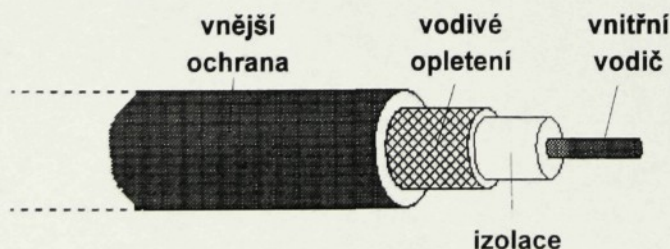
➤ **odolnost vůči rušení**

V okolí kabelu může docházet k různým jevům, které mají nepříznivý vliv na přenášený signál – jde např. o provoz různých elektrických spotřebičů apod. Každý druh kabelu vykazuje obecně jinou odolnost vůči rušení (jednou ze specifických forem rušení je také tzv. **přeslech**, kdy signál, přenášený jedním kabelem, ovlivňuje průběh signálu v jiném kabelu – nejčastěji se projevuje u souběžně vedených vodičů).

V důsledku útlumu a rušení pak dochází ke **zkreslení** přenášeného signálu.

5.2.1 Koaxiální kabel

Koaxiální kabely mohou přenášet data přenosovými rychlostmi až 350 Mb/s. Samotný kabel (obr. 5) je tvořen vnitřním (měděným) vodičem obklopeným vrstvou izolačního materiálu. Tato vnitřní izolační vrstva je obklopena vodivou vrstvou tvořenou drátěným opletením. Vnější vrstva slouží jako vnější sekundární vodič a je-li uzemněna, slouží též jako stínění. Celá právě popsaná soustava je navíc chráněna vnější ochrannou izolací z plastiku.

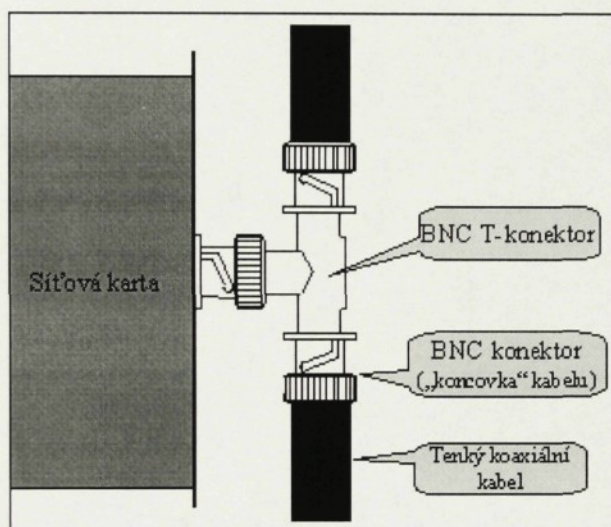


Obr. 5: Koaxiální kabel

V praxi se můžeme u LAN nejčastěji setkat s tenkým koaxiálním kabelem, nazývaným obvykle tenký Ethernet, nebo se silnější variantou, nazývanou silný (thick) Ethernet.

➤ **tenký Ethernet**

Jedná se o kabel rozměrově podobný koaxiálnímu kabelu běžně používanému pro připojování televizních antén, ale s jinými elektrickými vlastnostmi. Jedná se o kabel s impedancí 50 ohmů [5]. Připojování stanic k počítačové síti s tenkým koaxiálním kabelem je jednoduchá záležitost, která vyžaduje kompatibilní síťovou kartu, již zmiňovaný tenký koaxiální kabel a BNC (Bayonet–Neill–Concelman) konektor (koncovku na kabel) a BNC T-konektor jako propojovací prvek. Výsledné připojení vyjadřuje obr. 6.



Obr. 6: Připojení stanice s použitím BNC a tenkého koaxiálního kabelu

➤ **silný Ethernet**

Tento kabel umožňuje díky svým elektrickým vlastnostem realizaci rozsáhlejší sítě než kabel tenký, ale pro svou vyšší cenu a obtížnou instalaci se používá méně často. Jeho impedance je rovněž 50 ohmů.

5.2.2 Kroucená dvoulinka

Vedle koaxiálních kabelů připadají v úvahu pro propojování uzlových počítačů také dvojice běžných jednožilových vodičů. Aby se co možná nejvíce minimalizoval vliv rušení, používá se pro tyto vodiče tzv. diferenciální buzení - což znamená, že přenášenou informací vyjadřuje rozdíl signálů na obou vodičích.

Dojde-li pak vlivem vnějších rušení ke stejné změně signálu na obou vodičích, na rozdíl obou signálů se to neprojeví.

Dva paralelně vedoucí vodiče však vždy fungují jako jednoduchá anténa. Při souběžném vedení více takovýchto dvojic pak může mezi nimi docházet k nežádoucímu přeslechu. Lze jej výrazně omezit zkroucením jednotlivých dvojic vodičů, což minimalizuje jejich chování jako antény – tak vzniká kroucená dvoulinka (obr. 7).



Obr. 7: Kroucená dvoulinka

Kroucená dvoulinka patří do skupiny symetrického vedení. Známe ho z telefonních kabelů a je nejlevnějším přenosovým médiem. Ve většině případů jde o stíněnou (STP – Shielded Twisted Pair) nebo nestíněnou (UTP – Unshielded Twisted Pair), jednoduchou nebo dvojitou dvoulinku, která dovoluje bez problémů přenášet signály rychlých sítí, jako jsou Ethernet, 100Base-T, FDDI a ATM na vzdálenost 100m, přenosové rychlosti jsou zde do 155 Mb/s. Rozdíl mezi STP a UTP je v tom, že v případě STP jsou jak vnitřní dvoulinky, tak celý kabel opatřeny stíněním z hliníkové fólie. Toto stínění sice poněkud zvyšuje cenu kabelu, ale získáme tak kabel s výrazně vyšší odolností proti vnějším poruchám, což umožní spojit počítače nacházející se ve větších vzdálenostech.

Kabely z kroucené dvoulinky jsou zakončovány konektory řady RJ-45 (obr. 8 a 9), které jsou 8-pólové. Využity jsou jen 4 piny:

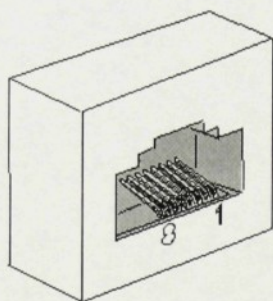
pin č. 1: TransmitData+

pin č. 2: TD-

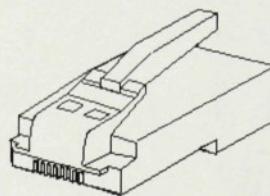
pin č. 3: Receive Data+

pin č. 6: RD-

Pro připojení každého uzlu jsou nutné 2 páry zkroucených vodičů (jeden pro vysílání dat, druhý pro příjem).



Obr. 8: Zásuvka typu RJ-45



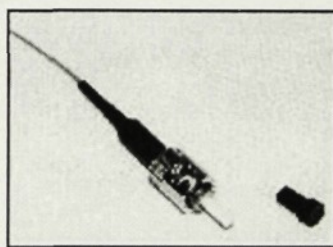
Obr. 9: Zástrčka typu RJ-45

5.2.3 Optické kabely

Optické kabely (obr. 10), světlovodná vlákna, využívají infračervené a viditelné oblasti světelného spektra pro přenos dat rychlostmi do 2,5 Gb/s na kilometrové vzdálenosti. Výhodou optických kabelů je vysoká přenosová kapacita při nízké ceně média a velká odolnost proti rušení, nevýhodou je vysoká cena prvků rozhraní, konektorů a náročné spojování kabelů. S optickými vlákny se setkáváme v lokálních sítích s kruhovou nebo stromovou topologií. K zakončení optických kabelů se využívá ST konektorů (obr. 11).



Obr. 10: Optický kabel



Obr. 11: ST konektor

5.3 Architektura komunikačních funkcí

Současné LAN se opírají o technologie, které jsou vesměs definovány standardy normalizačních organizací jako jsou IEEE (Institute of Electrical and Electronics Engineers), ANSI (American National Standards Institute) a ISO (International Organization for Standardization). Patří sem několik variant Ethernetu, kruhová síť IBM Token Ring a sítě pro technologické řízení.

5.3.1 Architektura ISO OSI

Pro přepojovací počítačové sítě, ze kterých se na počátku osmdesátých let vyvinuly dnes provozované veřejné datové sítě, byl vytvořen standardní model síťové architektury označovaný jako ISO/OSI (Open System Interconnection). Model OSI popisuje komunikaci zajišťovanou počítači, jako hierarchii sedmi vrstev technických a/nebo programových prostředků, kde každá z vrstev zajišťuje funkce potřebné pro vrstvu vyšší a využívá služeb vrstvy nižší. Mezi jednotlivými vrstvami jsou (formou standardů a doporučení) definována rozhraní (mezivrstvové protokoly), mezi prvky stejné vrstvy jsou definována pravidla komunikace (vrstvé protokoly). Architekturu vrstev modelu OSI ilustruje obr. 12.



Obr. 12: Architektura vrstev ISO OSI

➤ **Fyzická vrstva** - (Physical Layer)

Fyzická vrstva přenáší bitové proudy mezi odesílatelem a příjemcem. Definuje připojení síťových zařízení po mechanické a elektrické stránce, určuje typy konektorů a jednotlivých pinů a determinuje napěťové, resp. proudové úrovně přenášených signálů.

➤ **Prezentační vrstva** - (Presentation Layer)

Prezentační vrstva určuje formát, v němž jsou data přenášena a je odpovědná za jejich komprimaci resp. za jejich kódování.

➤ **Relační vrstva** - (Session Layer)

Relační vrstva vytváří, ukončuje a synchronizuje spojení mezi uživateli.

➤ **Transportní - (Transport Layer)**

Transportní vrstva se soustřeďuje pouze na komunikaci mezi koncovými účastníky spojení. Na základě požadavku vytváří spojení, rozčleňuje data od paketů a z přijatých paketů vytváří data původní.

➤ **Síťová - (Network Layer)**

Síťová vrstva přeměňuje rámce na pakety, pro které zřizuje vhodnou trasu. Uplatňuje se především tam, kde odesílatel a příjemce nejsou přímo spojeni a zajišťuje směrování datových paketů přes mezilehlé uzly sítě.

➤ **Linková (spojová) vrstva - (Data Link Layer)**

Linková vrstva řídí proud dat. Pracuje s tzv. datovými rámci, složenými z různého počtu bytů (bajtů), které sestavuje a k nimž přidává adresu příjemce a kontrolní informace.

➤ **Aplikační vrstva - (Application Layer)**

Aplikační vrstva tvoří rozhraní k uživatelské aplikaci a umožňuje uživateli přístup ke službám sítě (E-MAIL, terminálová emulace, vzdálený přístup, ...).

5.3.2 Architektura TCP/IP

Normalizačního úsilí v oblasti lokálních sítí se ujala organizace IEEE, jejíž pracovní skupiny si vzaly za úkol definovat univerzální standard pro lokální datové komunikace, označen jako IEEE 802.

Specifikace IEEE 802 popisují způsob, jak přenést konkrétní lokální síť bloky dat – rámce. Využití obsahu těchto rámců pro data aplikací a pro řízení vyšších síťových služeb je záležitostí vyšších vrstev architektury (síťové, transportní, relační, prezentační a aplikační).

	IPX/SPX	NetBIOS	TCP/IP	VINES	AppleTalk
Aplikační	Application Programs				
	Netwre Core Protocol (NCP)	Server Message Block (SMB)	Remote Procedure Call (RPC/XDR)	Remote Procedure Call (NetRCP)	AppleTalk Filling Protocol (AFP)
	NetBIOS	NetBIOS			AppleTalk Session Protocol (ASP)
	Transportní	Sequenced Packet Exchange (SPX)	NetBIOS Extended User Interface (NetBEUI)	Transmission Control Protocol (TCP/UDP)	VINES Interprocess Communication Protocol (VIPC)
Síťová	Internetwork Packet exchange (IPX)	Internet Protocol (IP)		VINES Internet Protocol (VIP)	Datagram Delivery Protocol (DDP)
Linková	Software Driver Network Interface Card				
Fyzická	Transmission Media				

Tab. 1: Architektura TCP/IP, IPX/SPX, NetBIOS, VINES a AppleTalk

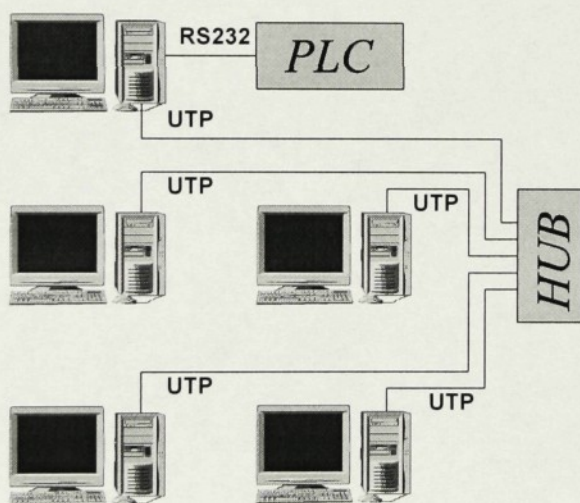
Tab. 1 uvádí protokolové sady typické pro architektury TCP/IP (dnes převažující), IPX/SPX, NetBIOS, VINES a AppleTalk. Tyto sady většinou zahrnují síťový protokol IP, IPX, VIP a DDP), transportní protokol (TCP, SPX, NetBIOS, VIPC a ATP) a aplikační rozhraní (RPC/XDR, NCP, NetBEUI, NetRPC a AFP).

5.4. Návrh počítačové sítě v laboratoři KKY

Problematika návrhu počítačové sítě v laboratoři KKY se sestává ze dvou částí. První část je vytvoření sítě v laboratoři a druhá část je pak připojení této sítě do akademické sítě LIANE.

Vzhledem k tomu, že dnes nemůžeme pevně stanovit počet počítačů v laboratoři, nebo jejich uspořádání, jeví se jako nejvýhodnější použít hvězdicovou topologii (všechny počítače mají svou linku od centrálního prvku – hubu). Charakteristika této topologie s jejími přednostmi je v kapitole 5.1.2.

Učebna je vybavena pracovními stanicemi typu PC se síťovými kartami pro Ethernet na kroucené dvoulince. Každá stanice má přidělenou vlastní IP adresu. Komunikace stanic při automatickém řízení technologického procesu využívá TCP/IP síťového protokolu a IP adres jednotlivých stanic. Jako přenosové médium je použita nestíněná kroucená dvoulinka - UTP. Návrh propojení stanic je na obr. 13.



Obr. 13: Návrh propojení stanic v laboratoři

Pro správný běh sítě v laboratoři je potřeba jedné stanice, která se bude chovat jako server (umístěna libovolně v prostorách katedry) a dalších PC, které budou přímo v laboratoři.

Na serveru poběží operační systém Windows 2000 Server, který se bude starat o ověřování uživatelů a hesel do sítě v laboratoři, popř. pro potřeby síťových aplikací (licence pro SW Matlab). Na stanicích v laboratoři bude nainstalovaný systém Windows 2000 Professional.

Po úspěšném přihlášení na server se stáváme právoplatnými uživateli sítě v laboratoři na KKY.

5.4.1 Vyjádření potřeb pro realizaci laboratoře

Pro správný chod laboratoře je zapotřebí následující:

- 5x PC jako pracovní stanice v laboratoři,
- 1x PC jako server,
- 6x síťová karta pro Ethernet na kroucené dvoulince,
- 1x hub s minimálně 5-ti připojeními,
- 35 m UTP kategorie 5,
- 5x SW Windows 2000 Professional,
- 1x SW Windows 2000 Server pro 15 stanic.

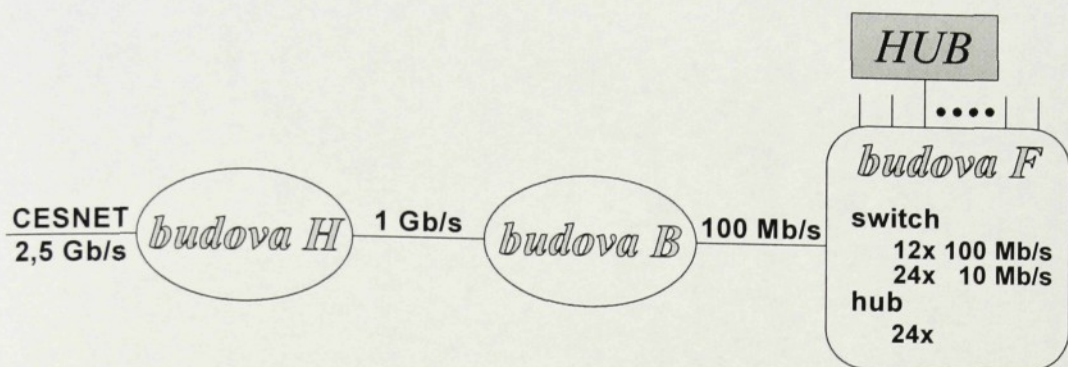
Ve skutečnosti jsou tyto potřeby redukovány na:

- 1x PC jako pracovní stanice,
- 1x PC jako server,
- 2x síťová karta pro Ethernet na kroucené dvoulince,
- 1x SW Windows 2000 Professional,
- 1x SW Windows 2000 Server,

Finanční vyčíslení potřeb je **145 697,- Kč** (Cena je s DPH). Tato cena je oficiální nabídkou od počítačové firmy OZO Czech Republic, s.r.o. Originální nabídkový list je v příloze 1.

5.4.2 Připojení laboratoře do akademické sítě LIANE

Pro připojení do akademické sítě LIANE využijeme strukturované kabeláže v budově F. V budově je switch, který má 12 kanálů 100 Mb/s a 48 kanálů 10 Mb/s a hub s 24 porty, ze kterých kabeláž vychází. Do tohoto místa je přivedena jedna odnož sítě LIANE. Tím je připojení do LIANE vyřešeno. Náhled na celou situaci je na obr. 14.



Obr. 14: Připojení laboratoře na akademickou síť LIANE

6. VIZUALIZACE TECHNOLOGICKÉHO PROCESU

V dnešní době existuje mnoho různých firem, které se specializují právě na automatizaci technologických procesů. Tyto firmy nabízejí široký sortiment řídicích systémů, snímačů, akčních členů, jiných prvků potřebných k automatizaci a v neposlední řadě i vizualizační software.

Důležitým faktorem, při chodu technologického procesu v automatickém režimu, je monitorování a vizualizace celého řízeného komplexu.

Hlavním úkolem monitorování a vizualizace řízeného technologického procesu je kontrola a oznámení zvláštních, kritických nebo havarijních stavů a jejich ošetření. K těmto základním vlastnostem monitorování a vizualizace ještě připadá možnost částečného zásahu obsluhy do řízeného komplexu podle okolností a stavů, v jakých se zrovna systém nachází. Vizualizaci provádíme většinou pomocí PC nebo speciálního počítače, který obsahuje software určený ke komunikaci s řídicími prvky a umožňuje vizualizaci činnosti řízené soustavy. Počítač může být spojen komunikačními médii s jedním nebo více řídicími prvky, s jejichž pomocí je celý technologický proces řízen. Nespornou výhodou monitorování a vizualizace je skutečnost, že obsluha má možnost dohlížet na celý proces z jednoho místa, velínu, za pomoci zobrazovacího zařízení, např. monitoru.

Hlavním požadavkem na monitorování a vizualizaci je skutečnost, že vše musí probíhat v reálném čase (všechny operace a výpočty, vztažené k jedné vzorkovací periodě, proběhnou v dané vzorkovací periodě). Zavedením automatizace se odstraňují chyby způsobené nepozorností nebo nedbalostí obsluhy, a tím se výrazně zvýší produktivita a kvalita výroby. Před zavedením automatizace do praxe musí dojít k pečlivé analýze problému, která by nám měla

stanovit podmínky, při kterých zavedením automatizace do provozu dosáhneme stanovených požadavků.

6.1 Stručná charakteristika dostupných vizualizačních systémů

6.1.1 Control Panel

Control Panel je objektově orientovaný systém, který slouží ke generování měřicích a regulačních programů pro PC. Systém může být využit v široké škále aplikací. Tento systém zajišťuje v reálném čase sběr dat, jejich zobrazení, archivaci a prezentaci, matematické zpracování a vyhodnocení dat a generování řídicích signálů pro řízený proces. Je distribuován firmou Alcor – Moravské přístroje, a.s. Zlín [6].

Návrh určité aplikace v prostředí Control Panel je vlastně vyšší formou programování. Uživatel je umožněno vytvářet programy dvojím způsobem, a to pomocí textového nebo grafického editoru (je umožněno přecházet z jednoho režimu do druhého dle potřeby a vlastního uvážení).

Výhody:

- Vývojové prostředí s překladačem a sadou užitečných nástrojů je integrováno v systému a může běžet se spuštěným aplikačním programem i ostatními nainstalovanými úlohami,
- Nemá v principu žádná omezení počtu užívaných I/O kanálů nebo zobrazitelných údajů. Jediným limitem jsou: velikost paměti, velikost disku a rychlost počítače (což záleží na uživateli).
- Pracuje ve víceúlohovém prostředí s grafickým uživatelským rozhraním.
- Spolupracuje s různými grafickými adaptéry.
- Během krátké doby lze v Control Panelu vytvořit funkční a výkonnou aplikaci.

6.1.2 Trend Accord

Trend Accord je vizualizační software pro operátorské a dispečerské stanice na bázi PC, který je distribuován firmou AutoCont, a.s. Ostrava. Jedná se o plně grafický, uživatelsky orientovaný SW pro podporu běžných požadavků na operátorské pracoviště. V systému jsou integrovány následující systémy:

- **Monitorovací systém** – umožňuje velmi rychle vytvořit obrazovky sledovaných hodnot pomocí sady předformátovaných obrazovek.
- **Alarmní systém** – stav alarmního systému má přímý vliv na monitorovací systém, takže operátor má neustále přehled o stavů alarmů na sledovaných signálech.
- **Aktuální a historické trendy** – umožňují sledovat průběh kritických veličin, popř. zjistit průběh hodnot z libovolného období z dat archivovaných na disku. Data lze dále exportovat do databázového formátu a statisticky vyhodnocovat.
- **Recepční systém** – umožňuje archivovat často používané sady dat (recepty). Tato data lze pak snadno vybrat a zaslat do technologie, čímž se dají rychle změnit parametry procesu.
- **Komunikace v síti** – systém lze navrhnout jako víceuživatelský, tzn., že technologii může ovládat několik operátorů z různých míst. Jednotlivé stanice se propojí sítí LAN

Nevýhoda:

- Systém má slabší grafické ztvárnění, což by mohl být problém u složitějších aplikací.

6.1.3 Citect

Vizualizační a řídicí programový systém Citect je distribuován firmou AutoCont, a.s. Ostrava. Dovoluje zpracovávat informace z velkého množství řídicích zařízení a existujících technologií. Lze ho použít pro malé i pro rozsáhlé aplikace různého druhu, protože je flexibilní (při použití běžné architektury a standardního HW).

- **Komunikace** – Citect komunikuje přímo s řídicími a monitorovacími jednotkami (nejčastěji PLC) v případě, že tuto komunikaci nabízejí.
- **Připojení** – u jednotlivých aplikací se provede pomocí instalovaného sériového portu počítače a RS232 kabelu bez potřeby dalšího HW. Pro zvýšení výkonu a rychlosti lze použít speciální kartu.
- **Grafické zpracování aplikace** – systém Citect má velké možnosti grafického vyjádření. Lze použít předdefinované grafické stránky podle požadavků aplikace.

- **Použití v síti** – SW lze snadno použít v síťové modifikaci připojením do LAN (možnost současně přihlásit až 256 uzlů).

Výhoda:

- Vzhledem k dobrým síťovým vlastnostem vhodné pro rozsáhlé aplikace.

Nevýhoda:

- Poměrně vysoké investiční náklady – nutnost podrobné analýzy pro využití SW.

6.1.4 TomPack

Systém TomPack je souborem programů pro vizualizaci technologických procesů typu SCADA (Supervisory Control And Data Acquisition) na počítačích typu PC pod operačním systémem Windows NT 4.0 a Windows 2000. SW je distribuován firmou ProjectSoft Hradec Králové, a.s.

Vzhledem k tomu, že tento software bude použitý k vizualizaci technologických procesů v laboratoři aplikované kybernetiky, věnujeme TomPacku samostatnou podkapitolu.

6.2 Vizualizační software TomPack

SW je určen jak pro malé systémy s jedním počítačem, tak pro rozsáhlé systémy mnoha počítačů propojených sítí. Umožňuje redundanci jednotlivých částí, tj. při výpadku jednoho PC lze technologii řídit pomocí záložního počítače.

K řídicím prvkům (nejčastěji PLC automatům) je TomPack připojen pomocí DDE serverů, přičemž je možné použít i DDE servery třetích stran. Základní DDE server, UniServer, je též produktem firmy ProjectSoft.

Součástí TomPacku je i skriptovací jazyk, který je podmnožinou jazyka C (není tedy nutné učit se další programovací jazyk). TomPack obsahuje řadu funkcí, mezi jinými i pro databázový přístup pomocí ODBC, regulátory, apod. Pomocí těchto funkcí a jazyka je možné vytvořit například živou demonstrační aplikaci, která nepotřebuje žádné řídicí a ovládací prvky.

TomPack umožňuje zabezpečit počítač tak, že obsluha bez znalosti hesla nemůže vstoupit do operačního systému.

Datové soubory generované TomPackem jsou ve formátu Microsoft Access 2000, takže není problém je dále zpracovávat.

Architektura TomPacku je typu client/server, jednotlivé části mohou běžet na různých počítačích, které jsou propojeny sítí. TPServer a DDE servery musí běžet stále, neboť zaznamenávají změny vybraných hodnot, alarmy atd. Naproti tomu TPView pouze zobrazuje a může být spuštěn jen tehdy, je-li to potřeba [7].

6.2.1 *Jednotlivé komponenty systému*

Samotný systém TomPack je složen (kromě DLL knihoven) ze tří programů – TPServer, TPConfig a TPView. K nim je nutné ještě připočíst DDE server, v našem případě UniServer. Tyto programy mají následující funkci:

- **DDE server**
 - Musí stále běžet.
 - Na základě požadavků klientů, připojených DDE komunikací, vyčítá data z PLC a posílá je klientovi.
 - Při požadavku klienta o zápis do PLC, tento povел provede.
- **TPServer**
 - Musí stále běžet.
 - Na základě požadavků od TPView atd. žádá o informace z DDE serveru (serverů).
 - Spravuje vnitřní proměnné, zapisuje si do trendových souborů a do souboru alarmů.
 - Poskytuje TPView veškerá data a při změně dat v TPView předává změny do DDE serveru, potažmo PLC.
- **TPView**
 - Běží pouze v případě, je-li to potřeba (vizuální kontrola systému).
 - Zobrazuje data, která dostane od TPServeru a zároveň do něj posílá změny.
- **TPConfig**
 - Slouží pro tvorbu projektu, který poté posílá programům TPServeru a TPView.
 - Konfigurace jak malého tak rozsáhlého projektu je uložena pouze v jednom souboru, což usnadňuje jeho správu.

6.2.1.1 DDE server

Pro komunikaci s automaty je nutné spustit jeden nebo více DDE serverů. TPServer je s DDE servery spojen standardní nebo rychlou DDE (FastDDE) komunikací. TPServer se vždy nejprve pokusí navázat komunikaci ve FastDDE. Nepodporuje-li ho DDE server, dále komunikuje pomocí normálního DDE.

Součástí TomPacku je DDE server – UniServer (funguje i s jinými DDE servery).

Seznámení s DDE

DDE server (Dynamic Data Exchange) slouží pro komunikaci mezi programy v prostředí Windows. Každá komunikovaná hodnota má svou adresu (viz níže) a formát. Nejobecnějším formátem je textový – ten by měly umět všechny programy.

Komunikace se odehrává mezi DDE serverem a DDE klientem, přičemž jeden server může obsluhovat více klientů a současně můžou být klienty jiných serverů.

Každá adresa v DDE komunikaci má tři složky (jejich znalost je pro spojení s DDE serverem podmiňující) - „Service name“ (někdy též označované jako „Application name“), „Topic name“ a „Item name“.

- **Service name** - někdy též označována jako „Application name“
- většinou je to jméno programu bez přípony .exe
(např. EXCEL)
- **Topic name** - jméno souboru hodnot
(např. TAB1 - jméno tabulky v EXCELU)
- **Item name** - jméno prvku
(např. R1C2 – první sloupec a druhý řádek v TAB1)

Při použití DDE komunikace s automaty je nepsanou úmluvou to, že Service name je jméno DDE serveru (nebo jméno definované v DDE serveru jako Service name), Topic name je označení automatu, definované v DDE serveru a Item name je označení proměnné dle zvyklostí DDE serveru.

Při navazování komunikace nejprve klient pošle požadavek všem aplikacím s adresou Service/Topic (INITIATE zpráva) a server, který má tuto adresu, klientovi odpoví (odpovědní INITIATE zpráva). Potom už klient sděluje přímo svému serveru, který Item chce posílat (ADVISE zpráva) a server mu posílá hodnotu Itemu při

každé změně (DATA zpráva). Klient naopak může poslat serveru novou hodnotu Itemu (POKE zpráva). Nepotřebuje-li klient dále nějaký Item, sdělí to serveru (UNADVISE zpráva). Při ukončení celé komunikace si klient a server ještě vymění TERMINATE zprávu o ukončení.

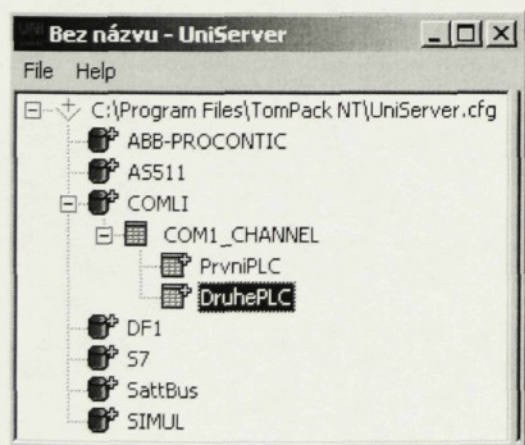
Příklad DDE adresace

Jako DDE server využijeme součást TomPacku – UniServer (obr. 15), ke kterému jsou připojeny 2 automaty s adresami 5 a 6, oba přes sériovou linku COM1. V serveru nejprve musíme vytvořit COM1_CHANNEL (obr. 16) pro komunikaci přes COMLI protokol. V nově vytvořeném kanálu potom vytvoříme dvě PLC – jedno s hodnotou Node 5 (obr. 17) a druhé s hodnotou Node 6. Jejich jména budou např. PrvniPLC a DruhePLC.

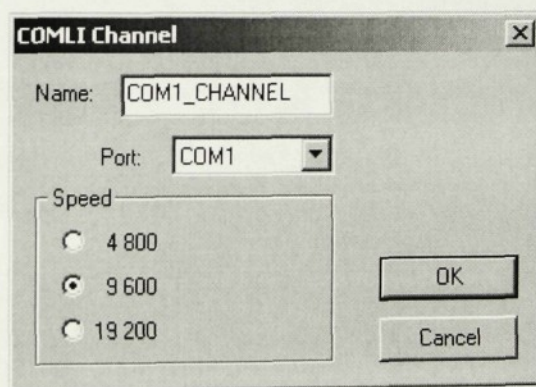
Nyní budeme chtít adresovat bit X34 v prvním automatu a registr R12 ve druhém. Jejich adresa (ve formátu SERVICE TOPIC ITEM) tedy bude:

- UniServer PrvniPLC X34
- UniServer DruhePLC R12

U DDE adresy nezáleží na velkých a malých písmenech, může se tedy zapsat i uNiSERVER atd.



Obr. 15: Prostředí UniServer



Obr. 16: Prostředí UniServer – vytvoření COM1_CHANNEL pro COMLI protokol

COMLI Topic Properties

Name:

Node:

Timeout:

Retries:

Cycle:

Dead cycle:

OK Cancel

Obr. 17: Prostředí UniServer - vytvoření spojení pro první PLC v COMLI protokolu

Skupiny proměnných jsou zavedeny kvůli adresaci DDE komunikace. Proměnné, přiřazené do jedné skupiny, jsou v automatu, který je určen touto skupinou. Teprve ve skupině je uvedeno jméno Service a Topic.

Skupiny journalů určují proměnné, které se budou zapisovat do jednoho souboru. Mají ještě společnou dobu, po kterou budou uchovávány v souboru.

6.2.1.2 TPServer

TPServer je jádrem systému TomPack. Po spuštění si nejprve načte z lokální kopie projektu (soubor `~~TPS~~.tp`) údaje (proměnné, skupiny proměnných a journalů, hesla a ServerCode). Poté zahájí komunikaci s DDE servery, začne ukládat zvolené proměnné do journalů a vyhodnocuje alarmy, jejichž změny ukládá do souboru s historií alarmů. Také spustí periodické vykonávání ServerCode a jedná-li se o redundantní server, spojí se s redundantním partnerem.

Ovládání i ukončení TPServeru je možné pouze po přihlášení uživatele s úrovní přístupu 999, což vylučuje možnost nekorektního zásahu obsluhy do sledovaného automatizovaného technologického procesu. Uživatelé jsou společně se svými hesly uloženi v projektu. Nemá-li tedy TPServer otevřen žádný projekt, nelze se k němu přihlásit. Je však samozřejmě možné poslat projekt z TPConfig.

Do pracovního adresáře si TPServer ukládá plnohodnotnou kopii projektu (soubor `~~TPS~~.tp`), soubor historie alarmů (Alarm.dat), soubor hodnot diskových

proměnných (DiskVar.dsk), logy (ServSys.log, ServSys.lo1, ServPipe.log) a soubory historických hodnot (s příponou .jou) a soubor .ldb vytvořené databázovými funkcemi.

6.2.1.3 TPView

TPView je jedinou částí systému, která komunikuje s uživatelem při běhu aplikace. TPView je spojen pouze s TPServerem, se kterým vede obousměrnou komunikaci (čtení dat, předávání změněných dat). Po spuštění si načte inicializační obrazovku a požádá TPServer o zaslání proměnných, které na ní jsou. Také si načte všechny ViewCode a zahájí jejich periodické vykonávání.

TPView může běžet na stejném či na jiném počítači, než je TPServer. Vzhledem k tomu, že neukládá žádná data, nemusí být spuštěn nepřetržitě.

Požadavky, které si TPView pro svůj bezchybný běh klade, jsou:

- Dostatečný HW, který musí umožnit spuštění síťového operačního systému Windows NT 4.0 nebo Windows 2000.
- Potřebná grafika pro ztvárnění technologického procesu.

Ovládání i ukončení TPView je možné pouze po přihlášení uživatele s úrovní přístupu 999, což vylučuje možnost nekorektního zásahu obsluhy do zobrazení technologického procesu. Uživatelé jsou společně se svými hesly uloženi v projektu.

Do pracovního adresáře si TPView ukládá plnohodnotnou kopii projektu (soubor ~~TPV~~.tp), logy (ViewSys.log, ViewSys.lo1) a soubor .ldb vytvořené databázovými funkcemi.

6.2.1.4 TPConfig

TPConfig je program, který vytváří projekt. Projektem je soubor s příponou tp, ve kterém je uložena celá konfigurace. Nemusí běžet, nemusí být ani na PC (z hlediska možné změny konfigurace TPView se jeho účast na PC doporučuje).

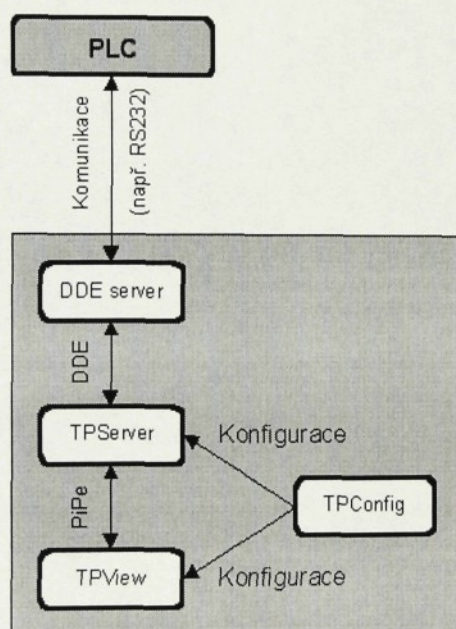
TPConfig rozešle hotový projekt do TPServer a do TPView. Ti si z projektu vytvoří plnohodnotnou kopii (~~TPS~~.tp a ~~TPV~~.tp), se kterou pracují. Při posílání projektu je nutné určit počítač, na kterém poběží příslušný TPServer, resp. TPView. Jinak se předpokládá, že běží na stejném počítači jako TPConfig.

6.2.2 Možné konfigurace pro realizaci vizualizace

Jak již bylo zmiňováno, programy TomPacku mohou, ale nemusí běžet na jednom počítači.

6.2.2.1 Triviální konfigurace

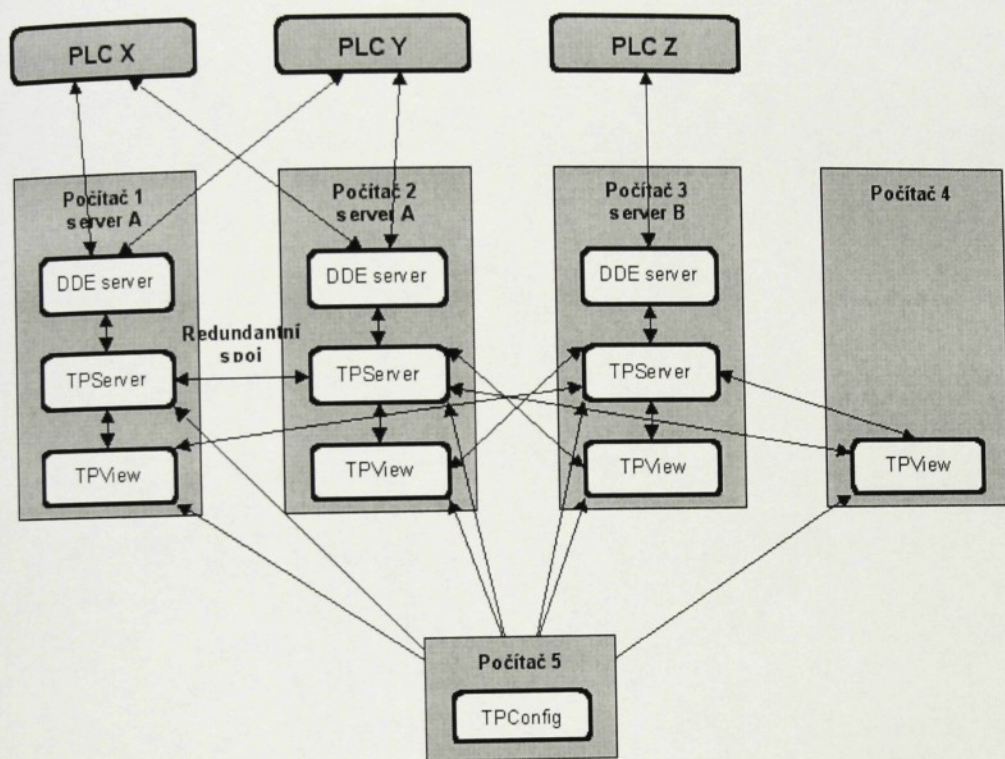
Celý projekt může být složen například z jednoho DDE serveru, TPServeru a jednoho TPView běžících na jednom počítači. TPConfig se samozřejmě spouští pouze v případě konfigurace TPServeru a TPView. Zobrazení takové struktury je na obr. 18.



Obr. 18: Zobrazení triviální konfigurace systému TomPack

6.2.2.2 Složitější konfigurace

Konfigurace systému může být podstatně složitější – viz obr. 19.



Obr. 19: Zobrazení složitější konfigurace systému TomPack

V tomto případě je aplikace složena ze dvou serverů. Server A běží na počítačích 1 a 2, ty jsou stejně propojeny s PLC. V případě havárie počítače 1 se všechny TPView přepojí na počítač 2 (redundantní spojení). Server B nemá zálohu, běží pouze na třetím počítači. Na čtvrtém počítači je pouze TPView, které se připojí na servery A a B. Počítač 5 se používá pouze ke konfiguraci celého systému.

6.2.3 Typy proměnných v TomPacku

Nezákladnější dělení proměnných je na typy: Process, Disk, Memory a View.

- **Process** - Proměnná je v automatu a TPServer uchovává její kopii.
- Změna proměnné se posílá do DDE serveru a do TPServer (při zápisu se hodnota proměnné změní ihned a nečeká se na potvrzení od DDE serveru, pouze se zablokuje změna od DDE serveru do přijetí potvrzení – kdyby se proměnná mezitím změnila v automatu). Provádí se proto, aby se na

- obrazovce změněný např. parametr ihned aktualizoval (zápis v automatu může trvat i několik sekund).
- Musí mít určenou Variable Group (určuje DDE server a Topic – konkrétní automat).
 - Musí mít určený Item Name (vlastní jméno proměnné – formát závislý na DDE serveru).
 - Mají složky proměnných.
- **Disk**
- Při startu se proměnné tohoto typu načtou ze souboru DiskVar.dsk, při změně se nová hodnota запиše zpět do tohoto souboru.
 - Neexistuje-li soubor DiskVar.dsk nebo proměnná v něm, přiřadí se proměnné inicializační hodnota z dialogu proměnné.
 - Mají složky proměnných.
- **Memory**
- Při startu se proměnná vždy nastaví na inicializační hodnotu. Po ukončení, nebo po reinicializaci TPServeru se na tuto hodnotu nastaví opět.
 - Hodnota proměnné je uchovávána v TPServeru - je-li několik TPView na jeden TPServer, proměnná má všude společnou hodnotu (hodnota uložená v TPServeru).
 - Mají složky proměnných.
- **View**
- chovají se podobně jako Memory, ale existují pouze v TPView – v každém TPView existuje vlastní instance proměnné a proměnná může mít jinou hodnotu (může se tedy do ní ukládat např. jméno přihlášeného uživatele, který je pro každý TPView jiný).
 - Nemají složky proměnných.

6.2.3.1 Složky proměnných

Každá proměnná (kromě typu View) je jakousi strukturou, k jejímž složkám je možné přistupovat – k názvu se připojí tečka a jméno složky. Pouze název proměnné označuje první složku, tedy přepočítanou hodnotu proměnné. Některé složky se ukládají do souboru DiskVar.dsk (a to nezávisle na tom, zda jsou typu Disk nebo ne), jiné jsou pouze virtuální (jsou určeny z výpočtu jiných složek). Přehled uvádějí následující tabulky 2, 3 a 4.

Proměnná Boolean

Složka	Typ	Atribut	Druh	Popis
.Raw	bool	R / W	process/disk/memory	hodnota přímo z DDE serveru, disku, atd.
.LoAlarm	bool	R	virtuální	nastal LoAlarm
.HiAlarm	bool	R	virtuální	nastal HiAlarm
.Valid	bool	R	virtuální	zda je proměnná platná

Tab. 2: Složky proměnné Boolean

Proměnná Real

Složka	Typ	Atribut	Druh	Popis
.Raw	double	R / W	process/disk/memory	hodnota přímo z DDE serveru, disku, atd.
.MinRaw	double	R / W	dle DiskParam	minimum Raw hodnoty
.MaxRaw	double	R / W	dle DiskParam	maximum Raw hodnoty
.MinEU	double	R / W	dle DiskParam	minimum přepočítané hodnoty
.MaxEU	double	R / W	dle DiskParam	maximum přepočítané hodnoty
.Units	string	R / W	dle DiskParam	jednotky
.LoLevel	double	string	disk	mez pro LoAlarm (alarm nastane při poklesu přepočítané hodnoty pod tuto mez)
.HiLevel	double	string	disk	mez pro HiAlarm
.LoAlarm	bool	R	virtuální	nastal LoAlarm
.HiAlarm	bool	R	virtuální	nastal HiAlarm
.Valid	bool	R	virtuální	zda je proměnná platná

Tab. 3: Složky proměnné Real

Proměnná String

Složka	Typ	Atribut	Druh	Popis
.Valid	bool	R	virtuální	zda je proměnná platná

Tab. 4: Složky proměnné String

7. TVORBA APLIKACE V PROSTŘEDÍ TOMPACK

Tvorba aplikace v prostředí TomPack je velmi jednoduchý proces, který je časově závislý na složitosti a požadavcích aplikace. Projekt se vytváří v programu TPConfig, poté se rozešle do TPServer a TPView (viz kapitola 6.2.2).

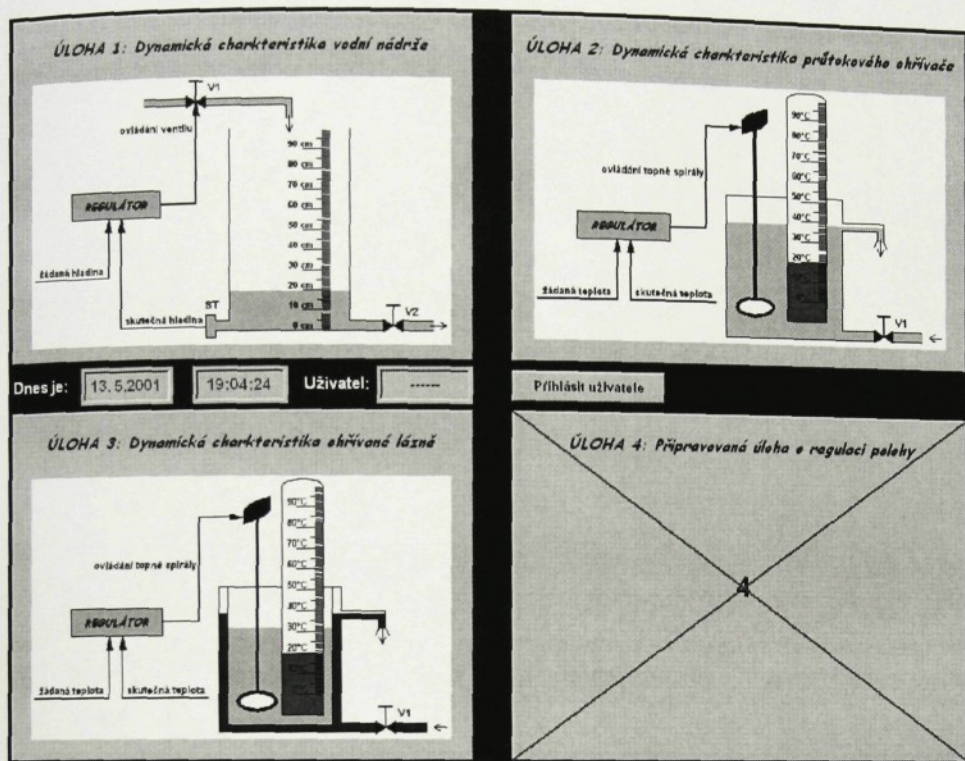
Vývojový prostředí TPConfig obsahuje kromě editoru proměnných také grafické prostředky pro tvorbu aplikace. Příkladem jsou editační okna proměnných, kreslení čar, elips a obdélníků a jiných geometrických tvarů, vkládání tlačítek, atd. Kombinací těchto prvků, proměnných a logických porovnání dostáváme požadovaný vizualizační SW.

7.1 Vizualizační SW vyvinutý pro laboratoř na KKY

Na začátku vývoje tohoto vizualizačního SW byly základní podmínky a požadavky pro realizaci projektu. Podmínkou byla schopnost vyvíjeného SW komunikovat s úlohami a regulovat je pomocí PLC přes RS232. Požadavkem byly potřeby ovládání kterékoliv ze tří úloh z libovolné stanice v laboratoři.

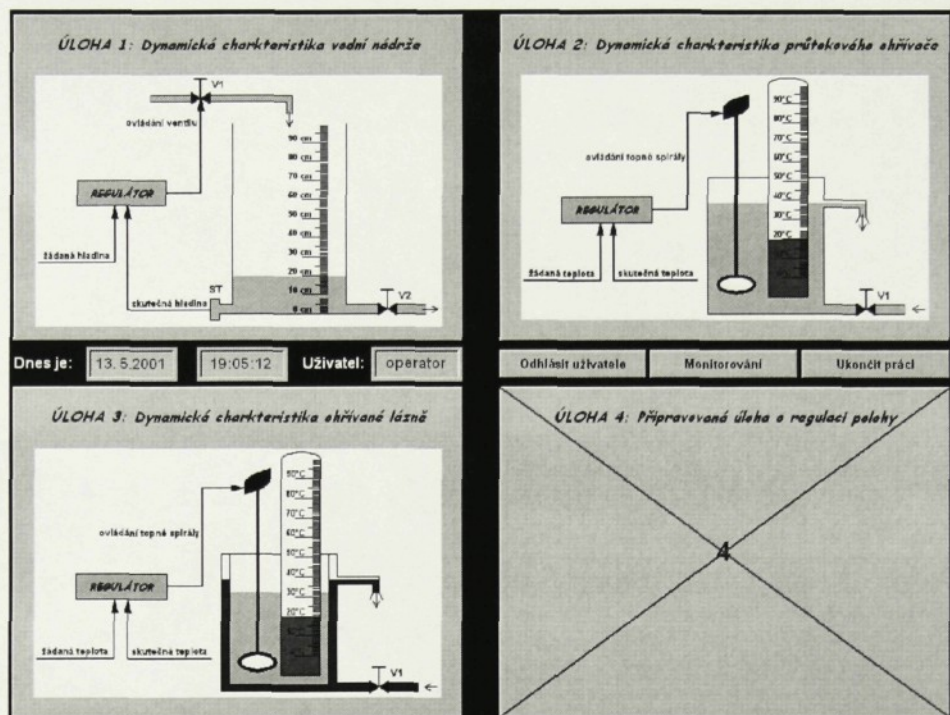
Vzhledem k okolnostem, ke kterým v průběhu vyvíjení tohoto SW došlo, muselo dojít ke změně ve vizualizaci a regulaci úloh a tudíž se modernizace a automatizace laboratoře dostala více k teoretickému řešení. Tato změna byla způsobena výpadkem PLC, což byl jeden ze základních kamenů původního návrhu modernizace laboratoře a tím i této diplomové práce. Náhradním řešením je vizualizace technologických procesů v paměti počítače s využitím regulátoru vestavěného v prostředí TomPack. V důsledku je tato situace s původním návrhem řízení pomocí PLC obdobná, protože není rozdíl, zda zavádím proměnnou do paměti PC či do registru PLC.

Pro správný chod modernizované laboratoře je na PC vytvořen uživatel **tompack** s přístupovým heslem **tompack**. Do spouštěcího skriptu stanic LAB1, LAB2, LAB4 a LAB5 je dán soubor TPView.exe. U stanice EAGLE je to kromě TPView.exe ještě TPServer.exe a příkaz pro vymazání všech souborů .dat z adresáře **e:\users\tompack** (důvody popsány dále v textu). SW je stavěn tak, že na PC EAGLE běží TPServer uchovávající stavy a hodnoty všech potřebných proměnných i naměřené hodnoty uložené právě do souboru .dat a bude využíván vyučujícím k případným korekturám (podrobněji dále v textu). Stanice LAB1, LAB2, LAB4 a LAB5 jsou připraveny pro studenty jako vizualizační stanice. Po spuštění TPView se zobrazí nabídka jako na obr. 20 (příloha 2).



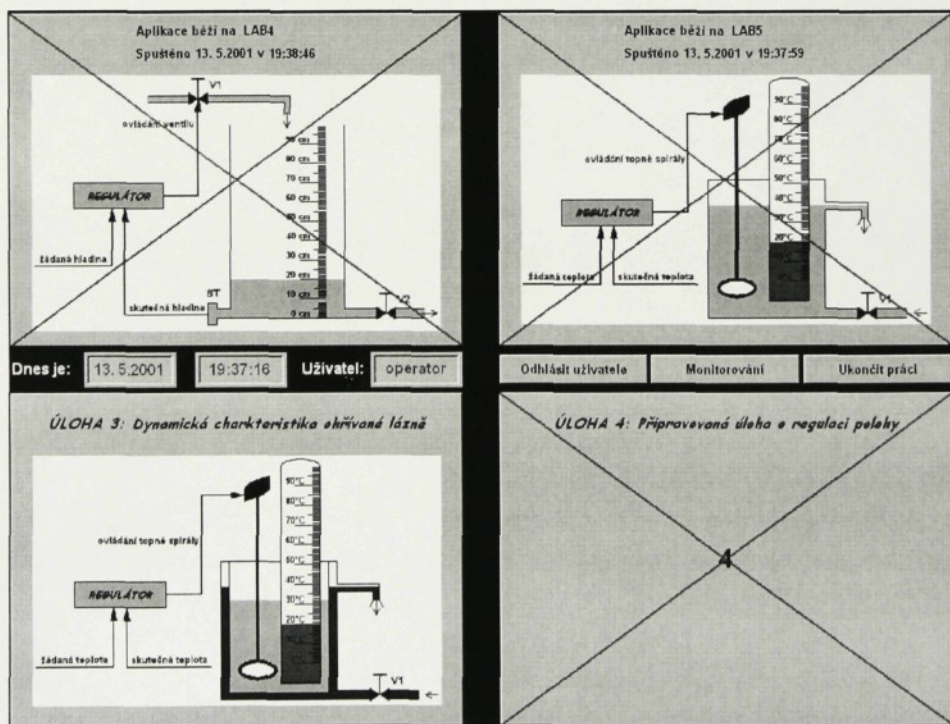
Obr. 20: Výběr úlohy bez přihlášeného uživatele

Tato nabídka dává možnost uživateli výběru libovolné úlohy, která ještě není obsazena. Po vyučujícího je v tomto okně důležité tlačítko „Přihlásit uživatele“ (uživatel: **operator**, heslo: **bez hesla**). Po úspěšném přihlášení se plocha změní o „Monitorování“ a o „Ukončit práci“ (obr. 21, příloha 3). Nutno ještě upozornit, že v nepřihlášeném stavu jsou blokovány řídicí klávesy (ALT-TAB, CTRL-ESC, CTRL-ALT-DELETE, ...), čímž je z části zamezeno studentské nekázní – nemohou se přepnout do jiné aplikace. Po přihlášení operátora jsou všechny tyto řídicí klávesy zpřístupněny.



Obr. 21: Výběr úlohy s přihlášeným uživatelem

Pokud nastane situace, že některá z úloh je již obsazena, potom bude vypadat úvodní strana podle obr. 22 (příloha 4).



Obr. 22: Omezený výběr úloh

Položka „Monitorování“ (obr. 23) slouží operátorovi k případným opravám kolizí vzniklých zhroucením některého z PC (kromě EAGLE) a také ke sledování obsazenosti úloh v přehledné formě.

Úloha 1 běží na stanici LAB4	Úloha 2 běží na stanici LAB5
Čas spuštění 19:31:12	Čas spuštění 19:30:28
Datum spuštění 13. 5.2001	Datum spuštění 13. 5.2001
Uvolnit a zpřístupnit úlohu 1	Uvolnit a zpřístupnit úlohu 2
Úloha 4 běží na stanici ----	
Čas spuštění ----	
Datum spuštění ----	
Stav TPView na LAB1	TPView není spuštěn
Stav TPView na LAB2	TPView je spuštěn
Stav TPView na LAB4	TPView je spuštěn
Stav TPView na LAB5	TPView je spuštěn
Stav TPView na EAGLE	TPView není spuštěn
Hlavní nabídka	
Ukončit práci	

Obr. 23: Monitorování

Položka „Ukončit práci“ (obr. 24) slouží pro vyučujícího jako nástroj pro vzdálené vypnutí TPView studentům pro další práci ve Windows, popřípadě pro ukončení TPView a vypnutí počítače. Zvýrazněná položka je počítač, na kterém pracuje přihlášená osoba.

Ukončení práce v TPView				
LAB1	LAB2	LAB4	LAB5	EAGLE
Ukončení práce ve Windows				
LAB1	LAB2	LAB4	LAB5	EAGLE
Hlavní nabídka				
Monitorování				

Obr. 24: Ukončení práce

Po výběru jedné z úloh se zobrazí ovládání pro konkrétní úlohu (ukázka jedné z nich jako příloha 5). Obsluha si může vybrat libovolný z dostupných regulátorů včetně nastavení jeho parametrů. Pro snadné sledování regulované veličiny poslouží zobrazení technologie s grafickým znázorněním již zmiňované žádané veličiny. Pro detailnější zobrazení změny žádané veličiny a akčního zásahu, slouží zobrazení grafů místo technologie. Toto přepnutí je pomocí tlačítka „Přepni na grafy“.

U konkrétní úlohy si můžeme vybrat, zda budeme měřit přechodovou nebo impulsní charakteristiku, nebo zda budeme regulovat technologický proces. Nutno upozornit, že v daný okamžik můžeme provádět pouze jeden proces (měření nebo regulaci). Pokud zvolíme některou z charakteristik, naměřená data se automaticky ukládají do souboru. Soubor se jmenuje podle stanice a podle charakteristiky (např. LAB2 impulsni.dat). V rámci regulace může ještě obsluha zvolit variantu s ukládáním nebo bez ukládání hodnot. Tento soubor bude vždy uložen na stanici, kde běží TPServer.exe, v našem případě na EAGLE (v pevně definovaném adresáři **e:\users\tompack**). Soubor se jmenuje podle stanice, ze které byl poslán (např. LAB2 regulace.dat). Hodnoty v těchto souborech jsou uloženy v textové podobě a lze s nimi bez problému dále pracovat např. v MS Excel.

Při každém novém přihlášení k stanici EAGLE dojde kromě spuštění TPServer.exe a TPView.exe také ke smazání všech dříve vytvořených souborů .dat.

U konkrétní spuštěné úlohy stále platí to, že pokud není přihlášen operátor, nelze práci ukončit (pouze vzdáleně od operátora přihlášeného na jiné stanici). Je možné přihlásit se i ve spuštěné úloze, čímž se nám zaktivní tlačítko pro opuštění úlohy.

7.2 Matematický popis použitých soustav

Pro realizaci úloh v paměti počítače jsou potřeba matematické popisy (přenosy soustav).

➤ úloha 1 $G_{S(p)} = \frac{80}{1,679p + 1}$

➤ úloha 2 $G_{S(p)} = \frac{50}{85p^2 + 31p + 1}$

➤ úloha 3 $G_{S(p)} = \frac{5}{p^3 + 1,812p^2 + 0,7256p + 0,078}$

Vzhledem k tomu, že TomPack neumí zacházet se spojitým systémem, musíme naše přenosy diskretizovat a dostáváme následující rovnice:

➤ úloha 1

$$y_k = 0,942180y_{k-1} + 4.625624u_{k-1}$$

➤ úloha 2

$$y_k = 1,964071y_{k-1} - 0,964186y_{k-2} + 0,002906u_{k-1} + 0,002871u_{k-2}$$

➤ úloha 3

$$y_k = 2,827599y_{k-1} - 2,661939y_{k-2} + 0,834268y_{k-3} + 0,0007976u_{k-1} + 0,003046u_{k-2} + 0,000728u_{k-3}$$

Pro realizaci regulačního obvodu je ještě zapotřebí popis regulátoru v diskrétním tvaru.

$$u_k = P \left\{ e_k - e_{k-1} + \frac{T}{T_I} e_k + \frac{T_D}{T} \{ e_k - 2e_{k-1} + e_{k-2} \} \right\} + u_{k-1},$$

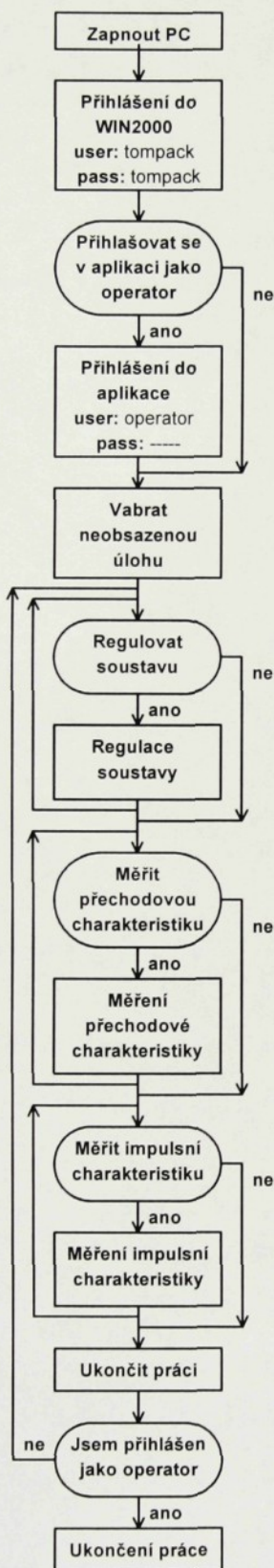
kde P , T_I a T_D jsou stavitelné parametry regulátoru. Tento popis není v aplikaci využit, protože součástí TomPacku je algoritmus pro regulátor typu PID a jeho modifikací.

Základní nastavení parametrů regulátorů je určeno metodou znalosti přechodové charakteristiky.

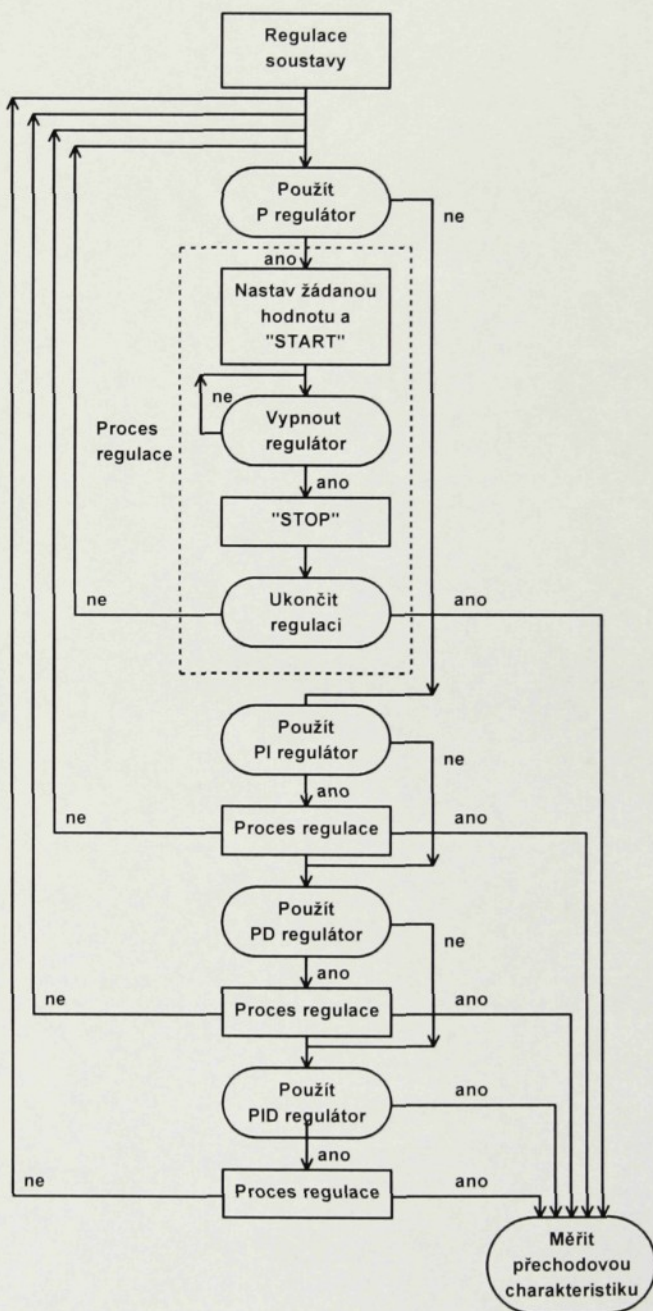
8. NÁVRH VÝUKY S VYUŽITÍM VYVINUTÉ APLIKACE

Na začátku výuky seznámíme studenty s možnostmi automatického řízení, popř. podrobněji popíšeme PLC, vysvětlíme jeho funkce, způsob programování a jeho komunikační možnosti. Dále studenty seznámíme s možností propojení s nadřazeným vizualizačním softwarem (v našem případě TomPackem). Na stanici EAGLE můžeme ještě předvést tvorbu jednoduchého projektu v TomPacku.

V rámci konkrétního ovládání vyvinuté aplikace seznámíme studenty s možnostmi regulace nebo měření přechodové a impulsní charakteristiky. Pro detailnější pohled poslouží následující vývojové diagramy (obr. 25 a 26).



Obr. 25: Vývojový diagram pro ovládání úlohy



Obr. 26: Vývojový diagram pro regulaci soustavy

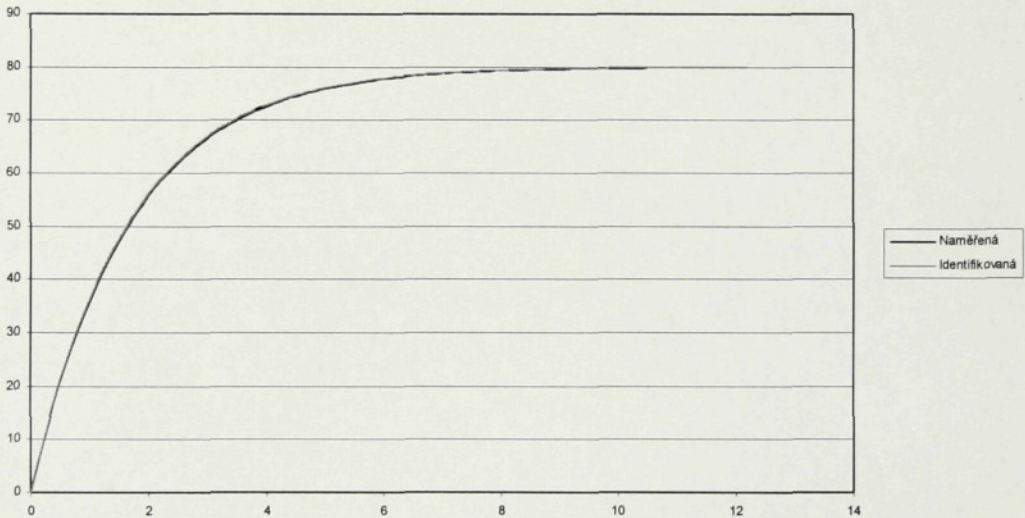
Měření přechodové a impulsní charakteristiky není zpracováno do vývojového diagramu. Měření spočívá ve volbě charakteristiky. Po stisknutí příslušného tlačítka se měření spustí, po ustálení hodnoty se měření automaticky ukončí. Výsledky jsou uloženy v textovém souboru. Následně můžeme provést identifikaci vybrané soustavy. Zpracování identifikace může být následující (pro identifikaci použity výsledky měření přechodové charakteristiky úlohy 1).

Identifikace je provedena z grafického vyjádření přechodové charakteristiky metodou Prof. Strejce. Matematický popis, ze kterého vycházelo naše měření je

$G_{S(p)} = \frac{80}{1,679p + 1}$. Z přechodové charakteristiky dostáváme popis

$G_{S(p)} = \frac{80}{1,655p + 1}$. Porovnání naměřené a identifikované přechodové

charakteristiky je na grafu 1.



Graf 1: Porovnání naměřené a identifikované přechodové charakteristiky

9. ZÁVĚR

Tato práce se zabývala problematikou automatizace technologických procesů s využitím vizualizačního SW vyvinutém v prostředí TomPack.

Navržené síťové prostředí v současné době vykazuje problémy s přístupovými právy. Za optimálního stavu by se uživatel **tompack** vytvořil na serveru. Tyto problémy jsou způsobeny nedostatečnou kompatibilitou mezi WIN NT 4.0 Server a Windows 2000 Professional. Problémy spojené s nekompatibilitou nemusí být pouze u síťového vizualizačního SW, ale mohou se vyskytnout i u jiných síťových SW, které se v budoucnu budou instalovat. Proto doporučuji v nejbližší době z finančních dotací zakoupit síťový operační systém Windows 2000 Server včetně stanice pro něj určené.

Díky výše popsaným problémům je nutné vytvořit uživatele pouze lokálně na stanicích a všude instalovat TomPack (v našem případě verzi TomPack 1.9E).

V rámci této diplomové práce navrhuji ještě jeden počítač, který by měl nahradit již ne zcela vyhovující EAGLE.

Jak již bylo výše napsáno, TomPack je velmi elegantní nástroj pro tvorbu vizualizačního softwaru. Tvorba vlastního projektu je jednoduchá a pro zkušeného programátora i rychlá. Možná jediná negativní věc, kterou jsem na TomPacku zpozoroval je skutečnost, že se jedná o software pro vizualizaci technologických soustav s většími časovými konstantami. Může se tedy stát, že při několika měřeních nám TomPack zobrazí vždy jinou křivku (týká se hlavně imlusní charakteristiky u úlohy 1). Hodnoty se však ukládají do souboru s periodou 0,1s a při několika měřeních budou vždy stejné.

Vzhledem k výše popsaným problémům s PLC, bych po jeho opravení doporučoval na tuto diplomovou práci navázat a zrealizovat fyzické propojení vizualizačního SW a reálných úloh.

Součástí této práce je CD s vytvořeným projektem a SW TomPack 1.9E.

LITERATURA:

- [1] Hanuš, B.: Základy technické kybernetiky, VŠST Liberec, 1989, skriptum
- [2] Olehla, M.: Identifikace technologických soustav, TU Liberec, 1997, skriptum
- [3] Bárta, J.: Úvod do počítačových sítí, KOPP České Budějovice, 1997
- [4] Janeček, J.: Lokální sítě, ČVUT Praha, 2000, skriptum
- [5] Hejna, L.: Lokální počítačové sítě, Grada Praha, 1994
- [6] Tlamicha, P.: Využití vizualizačního systému Control Panel pro monitorování technologického děje řízeného dvěma řídicími systémy, TU Liberec, 1997, diplomová práce
- [7] Podniková dokumentace firmy ProjectSoft HK, Hradec Králové, 2000

SEZNAM OBRÁZKŮ :

- Obr. 1: Řízení v otevřeném obvodu
- Obr. 2: Řízení v otevřeném obvodu s měřením poruchy
- Obr. 3: Řízení v uzavřeném obvodu
- Obr. 4: Návrh struktury laboratoře na KKY
- Obr. 5: Koaxiální kabel
- Obr. 6: Připojení stanice s použitím BNC a tenkého koaxiálního kabelu
- Obr. 7: Kroucená dvoulinka
- Obr. 8: Zásuvka typu RJ-45
- Obr. 9: Zástrčka typu RJ-45
- Obr. 10: Optický kabel
- Obr. 11: ST konektor
- Obr. 12: Architektura vrstev ISO OSI
- Obr. 13: Návrh propojení stanic v laboratoři
- Obr. 14: Připojení laboratoře na akademickou síť LIANE
- Obr. 15: Prostředí UniServer
- Obr. 16: Prostředí UniServer – vytvoření COMLI_CHANNEL pro COMLI protokol
- Obr. 17: Prostředí UniServer – vytvoření spojení pro první PLC v COMLI protokolu
- Obr. 18: Zobrazení triviální konfigurace
- Obr. 19: Zobrazení složitější konfigurace systému
- Obr. 20: Výběr úlohy bez přihlášeného uživatele
- Obr. 21: Výběr úlohy s přihlášeným uživatele
- Obr. 22: Omezený výběr úloh
- Obr. 23: Monitorování
- Obr. 24: Ukončení práce
- Obr. 25: Vývojový diagram pro ovládání úlohy
- Obr. 26: Vývojový diagram pro regulaci soustavy

SEZNAM TABULEK :

- Tab. 1: Architektura TCP/IP, IPX/SPX, NetBIOS, VINES a AppleTalk
- Tab. 2: Složky proměnné Boolean
- Tab. 3: Složky proměnné Real
- Tab. 4: Složky proměnné String

SEZNAM GRAFŮ :

Graf 1: Porovnání naměřené a identifikované přechodové charakteristiky

SEZNAM PŘÍLOH :

Příloha 1: Cenová nabídka firmy OZO Czech Republic, s.r.o.

Příloha 2: Úvodní nabídka bez přihlášeného uživatele

Příloha 3: Úvodní nabídka s přihlášeným uživatelem

Příloha 4: Úvodní nabídka s omezeným výběrem úloh

Příloha 5: Pohled na úlohu 1

PŘÍLOHA 1: Cenová nabídka firmy OZO Czech Republic, s.r.o.



OZO Czech Republic, s.r.o.
Barvířská 17, 460 01 Liberec

Nabídka počítačových sestav pro vybavení laboratoře

Stanice 1

Název komponentu

ZD AOPEN MX 33 skt.370 + ZVUK
CPU INTEL PIII 600EB
Chladič pro PIII
HD ST 20 GB
DIMM 128MB/133
CD ROM AOPEN
VGA S3 SAVAGE 16MB
FDD ALPS 3,5"
SK Planet ENW 9504
MIDITOWER ATX 606
Klávesnice + myš + podložka
Monitor 17" SAMPO

Celková cena bez DPH

26 523 Kč

Server W2000

Název komponentu

ZD ABIT BX 133 skt.370
CPU INTEL PIII 1000EB 133
Chladič pro PIII
HD ST 40 GB x 2 zrcadlení
DIMM 256MB/133 x 2
CD ROM AOPEN
VGA S3 4MB
FDD ALPS 3,5"
SK Planet ENW 9504
MIDITOWER ATX 606
Klávesnice + myš + podložka
Monitor 15" SAMPO

Celková cena bez DPH

45 662 Kč

Software

Windows 2000 Profesional OEM	5 431 Kč
WINdows 2000 Server 5 + 10 uživatelů	39 456 Kč
Instalace Windows 2000 server	10 000 Kč

Celková cena počítačových sestav bez DPH 22 % 72 185 Kč

Celková cena počítačových sestav s DPH 22 % 88 066 Kč

Celková cena software bez DPH 5% 54 887 Kč

Celková cena software s DPH 5% 57 631 Kč

Celková cena s DPH 145 697 Kč

Ceny jsou platné po dobu jednoho týdne. Poté si Firma OZO Czech Republic, s.r.o. vyhrazuje právo na změnu cen.

V Liberci 11.5.2001

Za OZO Czech Republic, s.r.o. Marek Hoffman

4.



OZO CZECH REPUBLIC
s.pol. s r.o.

Barvířská 17
460 01 Liberec 1
DIČ: 192-250204
tel.: (048) 510357

PŘÍLOHA 2: Úvodní nabídka bez přihlášeného uživatele

ÚLOHA 1: Dynamická charakteristika vodní nádrže

Dnes je: 13. 5. 2001

Uživatel: -----

19:04:24

ÚLOHA 2: Dynamická charakteristika průtokového ohříváče

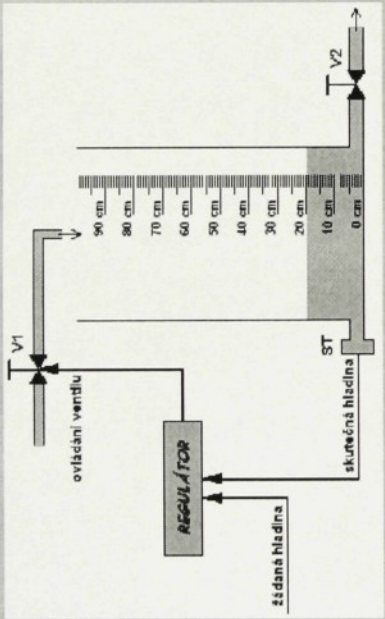
Přihlásit uživatele

ÚLOHA 4: Připravená úloha o regulaci polohy

ÚLOHA 3: Dynamická charakteristika ohřívání lázně

PŘÍLOHA 3: Úvodní nabídka s přihlášeným uživatelem

ÚLOHA 1: Dynamická charakteristika vodní nádrže

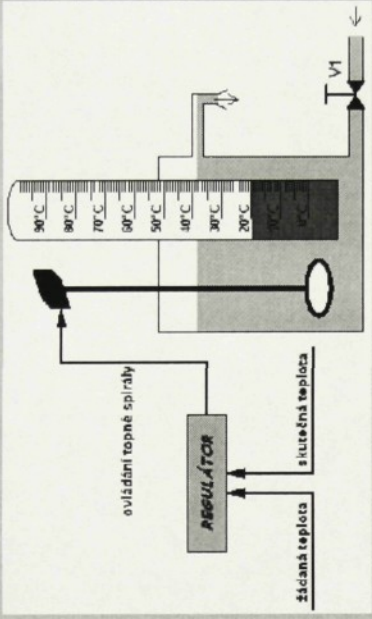


Dnes je: 13. 5. 2001

Uživatel: 19:05:12

operator

ÚLOHA 2: Dynamická charakteristika průtokového ohříváče

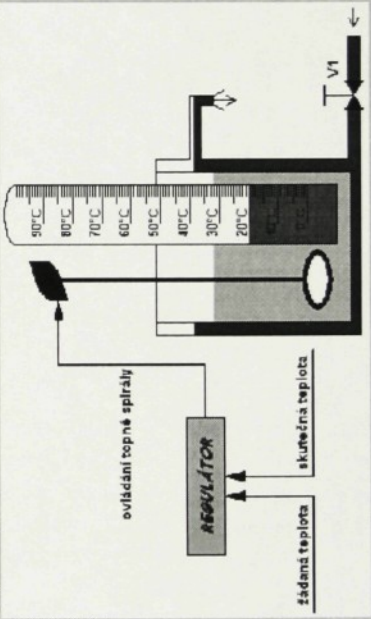


Odhlásit uživatele

Monitorování

Ukončit práci

ÚLOHA 3: Dynamická charakteristika ohřívání lázně



ÚLOHA 4: Přípravovaná úloha o regulaci polohy

PŘÍLOHA 4: Úvodní nabídka s omezeným výběrem úloh

[illegible]

ÚLOHA 1: Dynamická charakteristika vodní nádrže

● měření bez ukládání do souboru

● měření s ukládáním do souboru

16. 5.2001

13:36:40

operator

Odhlásit

Návrat do hlavní nabídky

MENU

VOLBA REGULÁTORU

☒ Dvoupolohový

☒ Dvoupoloh. s hystez.

P regulátor

PI regulátor

PD regulátor

☒ PID regulátor

NASTAV PID REGUL.

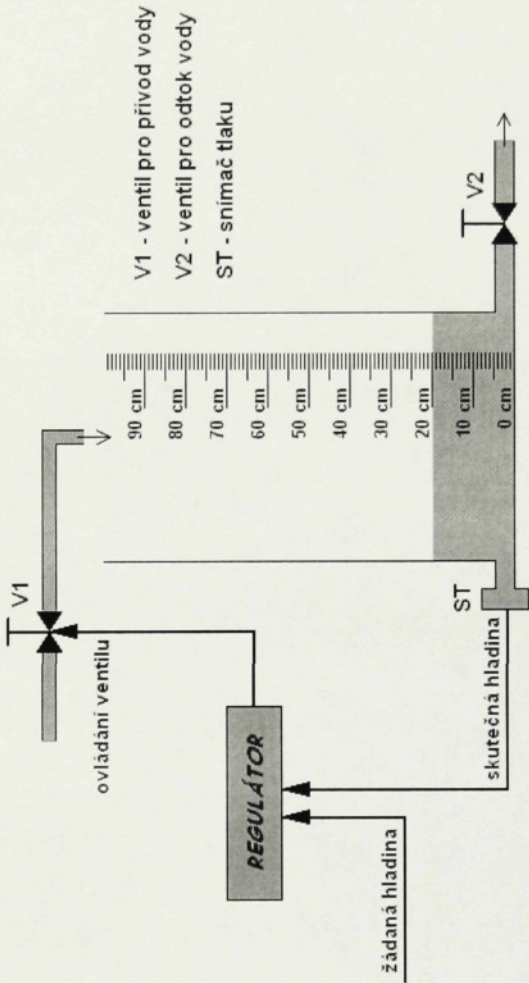
ŽÁDANÁ HLADINA

20 cm

◀ ▶

START

STOP



Přechodová charakteristika

Impulsní charakteristika

Prepni na grafy

PROHLÁŠENÍ

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 o právu autorském, zejména §60 (školní dílo) a § 35 (o nevýdělečném užití díla k vnitřní potřebě školy).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé práce a prohlašuji, že **souhlasím** s případným užitím mé práce (prodej, zapůjčení apod.).

Jsem si vědom toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mou jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Datum: 20.05.2001

Podpis:

