

TECHNICKÁ UNIVERZITA V LIBERCI
FAKULTA STROJNÍ



DIPLOMOVÁ PRÁCE

DUŠAN IKER

**Regulace výšky hladiny v nádobě s výtokem s využitím
programovatelného logického automatu**

**Regulation of a water level in tank with outflow by using
programmable logical automat**

LIBEREC 2002

TECHNICKÁ UNIVERZITA V LIBERCI
FAKULTA STROJNÍ
KATEDRA APLIKOVANÉ KYBERNETIKY

Obor 23 - 40 - 8

Automatizované systémy řízení ve strojírenství

Zaměření

Automatizace inženýrských prací

**Regulace výšky hladiny v nádobě s výtokem s využitím
programovatelného logického automatu**

Dušan Iker

Vedoucí diplomové práce: Ing. Slavomír Němeček

Konzultant diplomové práce: Ing. Michal Moučka

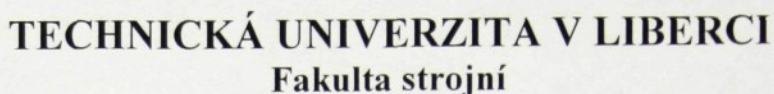
Rozsah práce a přílohy:

Počet stran textu: 59

Počet obrázků: 44

Počet příloh: 4

V Liberci 21. 05. 2002



Studijní rok: 2001/02

+ CT
V17/02 S

Forma zpracování diplomové práce:

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

Seznam literatury (uved'te doporučenou odbornou literaturu):

- [1] HANUŠ, B. : Teorie automatického řízení.
- [2] OLEHLA, M. : Základy aplikované kybernetiky.
- [3] BALÁTĚ, J. : Vybrané statě z automatického řízení.
- [4] MARTINÁSKOVÁ, M., ŠMEJKAL, L. : Řízení programovatelnými automaty
- [5] DOX 5. User's manual. Version 4.
- [6] Popis a použití vizualizačního systému TOMPACK.

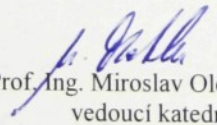
Vedoucí diplomové práce:

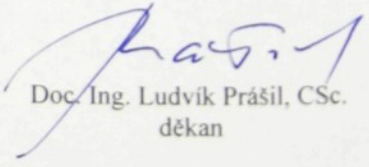
Ing. Slavomír Němeček

Konzultant diplomové práce:

Ing. Michal Moučka




Prof. Ing. Miroslav Olehla, CSc.
vedoucí katedry


Doc. Ing. Ludvík Prášil, CSc.
děkan

V Liberci dne 15.12. 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 konzultantu.

V Liberci dne 21. 05. 2002



Dušan Iker

ANOTACE

Tato diplomová práce se zabývá návrhem řídicího subsystému pro model laboratorního procesu, který představuje regulaci výšky hladiny v nádobě s přítokem a výtokem. Nejprve se seznámíme s možnostmi používání PLC v současném světě, možnostmi regulace a použití vizualizačního softwaru TomPack. Ve druhé části se zabýváme technickým a programovým řešením pro danou úlohu. Další část dává konkrétní pohled na vyvinutý software. V závěrečné části je uveden způsob využití navrženého SW při výuce předmětu Základy aplikované kybernetiky, vyučované na strojní fakultě Technické univerzity.

ANNOTATION

This diploma work deals with, how to the control system for laboratory process model, which represents control of a water level in a tank with inflow and outflow. First of all, we acquaint with the possibilities of using PLC in the current world, the control possibilities and visual application of the TomPack software. In a second part, we are interested in technical and program solutions for given problem. Next part gives us a concrete view on the developed software. In the final part there is introduced the method of using this software during a lesson of the Elementary applied cybernetics, which is taught in a faculty of Liberec.

Poděkování:

Tímto bych chtěl poděkovat vedoucímu diplomové práce ing. Slavomíru Němečkovi a konzultantovi ing. Michalu Moučkovi za odborné vedení, cenné rady a připomínky k mé práci.

OBSAH:

1. ÚVOD.....	8
2. LOGICKÉ ŘÍZENÍ.....	9
2.1 Technické vybavení PLC.....	9
2.2 Modulárnost PLC.....	10
2.3 Komunikační možnosti PLC.....	10
2.4 Vývojová prostředí a programování PLC.....	10
2.5 PLC v systémech řízení.....	11
2.6. Programovatelný automat versus řídicí systémy.....	12
2.7. Výhody PA.....	14
2.8. Nevýhody PA.....	14
3. REGULÁTORY.....	16
3.1 Rozdělení regulátorů.....	16
3.2 Dynamické vlastnosti spojitých regulátorů.....	18
4. TOMPACK.....	23
4.1 Vizualizační software TomPack.....	23
4.2 Jednotlivé komponenty systému.....	23
4.2.1. DDE server.....	24
4.2.2 TPServer.....	27
4.2.3 TPView.....	27
4.2.4 TPConfig.....	28
4.3 Možné konfigurace pro realizaci vizualizace.....	28
4.4 Typy proměnných v TomPacku.....	30
4.5 Složky proměnných.....	31
5. ÚVOD DO PROJEKTU.....	33
5.1 Laboratoř aplikované kybernetiky.....	33
5.2 PLC Siemens Simatic S7 - 300.....	33
5.3 Procesorová jednotka Siemens CPU 314 IFM.....	37
5.4 Zapojení PLC v laboratoři.....	38
5.5 Modelová technologie vodní nádrž.....	38
5.6 Popis soustavy.....	38
6. MATEMATICKÝ MODEL SOUSTAVY.....	41

7. TVORBA APLIKACE V PROSTŘEDÍ OMPACK.....	42
7.1 Vizualizační SW vyvinutý pro zadanou úlohu.....	42
7.2 Návrh algoritmů pro realizaci dvupolohové regulace.....	44
7.3 Návrh algoritmů pro realizaci třípolohové regulace.....	47
7.4 Charakteristiky regulátorů.....	49
7.5 Měření přechodové a impulsní charakteristiky.....	50
8. MODEL NÁDRŽE.....	52
8.1. Sestavení modelu.....	52
8.2. Propojení TomPacku a Matlabu.....	53
9. NÁVRH VÝUKY S VYUŽITÍM VYVINUTÉ APLIKACE.....	56
10. ZÁVĚR.....	57
LITERATURA.....	58
SEZNAM PŘÍLOH.....	59

1. ÚVOD

Dnes není automatizace něčím unikátním, co je výsadou drahého komfortu rozsáhlých výrobních linek a náročných technologických procesů. Kvalitní a inteligentní řízení je dostupné i pro obyčejné stroje, pomocné mechanismy a technologická zařízení ve všech oborech.

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 se zavedení automatizace do provozu dosáhne stanovených požadavků.

Úkolem této diplomové práce je návrh řídicího subsystému pro model laboratorního procesu, který představuje regulaci výšky hladiny v nádobě s přítokem a výtokem. Navrhnout vhodné technické a programové prostředky subsystému a jeho začlenění do projektu modernizace laboratoře aplikované kybernetiky, která je určena pro praktickou výuku předmětů studijního programu Automatizované systémy řízení ve strojírenství na strojínské fakultě TU v Liberci. Tato laboratoř je projektována tak, aby umožnila využití moderních technických a programových prostředků automatického řízení pro různé druhy automatické regulace a logického řízení. Při navrhování jednotlivých subsystémů modelu laboratorního procesu je proto kladen důraz na zachování modulární koncepce laboratoře, na její snadnou rozšiřitelnost a na možnost ovládání jednotlivých subsystémů modelu z jednoho místa. To vyžaduje nejen návrh a realizaci modelu subsystému technologického procesu, ale i vytvoření počítačové sítě, propojení PLC automatů s laboratorním procesem a jeho vizualizací. Výsledkem diplomové práce je proto návrh SW s možností vizualizace a ovládání modelu regulace výšky hladiny pomocí vizualizačního softwaru TomPack.

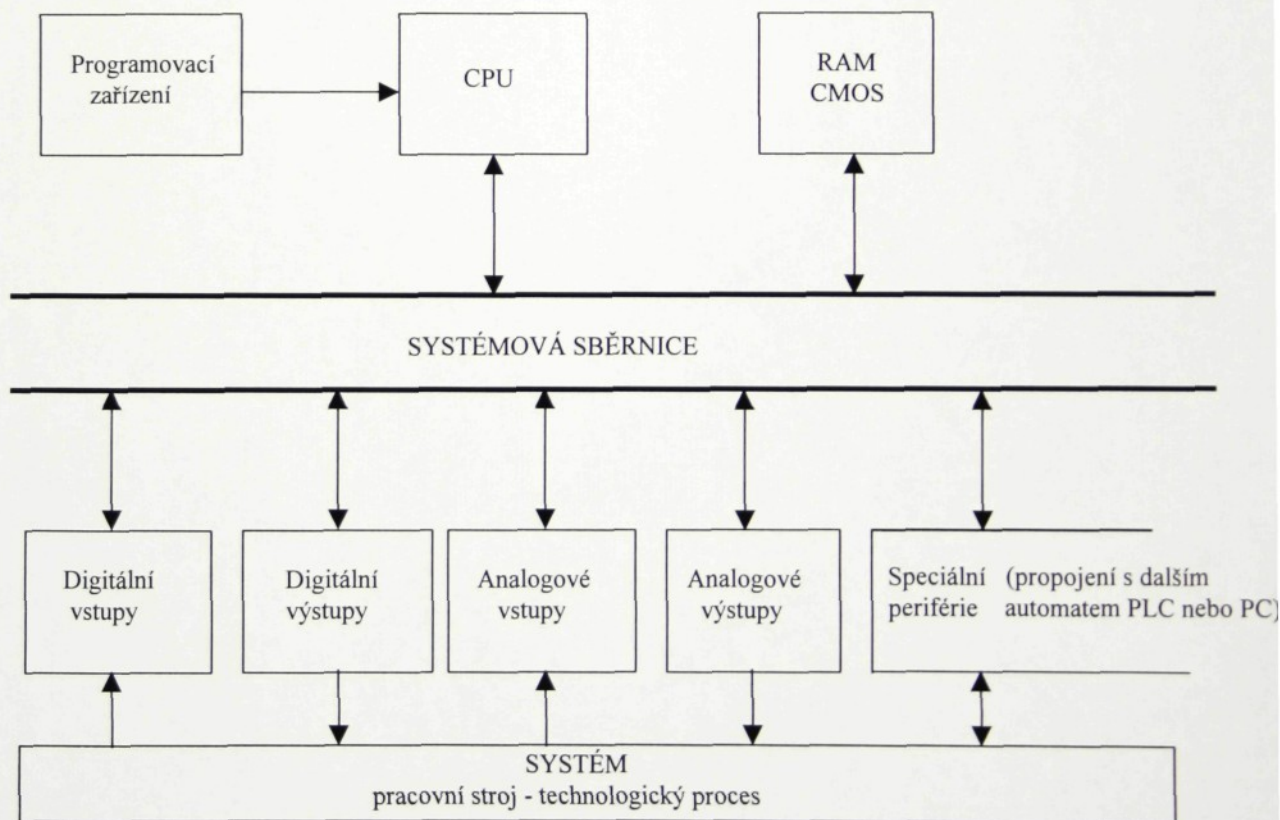
2. LOGICKÉ ŘÍZENÍ

Řídící systémy, které pracují s dvouhodnotovými signály nazýváme *logické automaty*. Podle daného algoritmu zpracovávají vstupní binární signály a generují akční signály, jimiž řídí technologický proces. Bývají označovány jako **PLC** (Programmable Logic Controller). Česká značka, která se teprve začíná používat, je **PA** (Programovatelný automat).

Původně byly programovatelné automaty navrženy k řešení úloh logického řízení, často jako přímá náhrada pevné reléové logiky. V současných aplikacích se však zvyšuje podíl úloh regulačního typu, úloh monitorování řízeného procesu i úloh analogových měření.

2.1 Technické vybavení PLC

Každý programovatelný automat se skládá z *centrální procesorové jednotky, systémové paměti, uživatelské paměti, souboru vstupních a výstupních jednotek* pro



RAMsystémová paměť CMOS ...uživatelská paměť CPUcentrální procesorová jednotka

Obr. 1.1 Principiální schéma PLC

připojení řízeného systému a souboru **komunikačních jednotek** pro komunikaci se souřadnými a nadřazenými řídicími systémy. Principiální schéma je na obr. 1.1.

Řídicí algoritmy jsou realizovány uživatelským programem, který může být zapsán v různých programovacích jazycích a po přeložení je uložen v uživatelské paměti programovatelného automatu. Program obsahuje posloupnost instrukcí, kterou procesor vykonává cyklicky. Chování PA je v podstatě dáno **zaměnitelným programem**.

2.2 Modulárnost PLC

Většina PLC automatů používá pro řízení procesů tzv. modulárnost PLC. Jde o to, že každý PLC systém lze libovolně sestavit z jednotlivých modulů podle náročnosti konkrétní řízené úlohy. Podle technických požadavků na řízení technologického procesu můžeme snadno z firemního katalogu vybrat právě ty jednotky, které nejlépe vyhovují požadavkům dané úlohy. Toto je rozhodující právě pro cenu celého systému. Znamená to tedy, že některé PLC mohou mít například jenom logické vstupy a výstupy nebo na druhou stranu jenom analogové atd. Dá se říci, že pouze ty nejmenší a nejlevnější PLC systémy, někdy označovány jako SLC (Small Logical Control) nabízejí uživateli konkrétní pevnou sestavu. Pochopitelně tomu odpovídá i nízká cena.

2.3 Komunikační možnosti PLC

Právě toto je velmi silnou stránkou PLC automatů. Důležitou vlastností současných PLC systémů je schopnost komunikovat se vzdálenými moduly vstupů a výstupů, s podsystémy, souřadnými i nadřazenými systémy, s operátorskými panely, s jinými inteligentními přístroji, s počítači a jejich sítěmi. U některých systémů jsou k dispozici i jednotky dálkových přenosů umožňující přenosy dat přes modem, JTS (jednotná telefonní síť) nebo přes radiomodem.

Pro sériovou komunikaci mezi zařízeními se dnes nejčastěji používají sériová rozhraní jako jsou proudová smyčka RS 232 nebo RS 485.

2.4 Vývojová prostředí a programování PLC

Pomocí programovacího přístroje lze zapsat a odladit uživatelský program určený pro PLC. Dnes jsou to především přenosné počítače typu notebook či laptop, které nahrazují dříve velmi často používané speciální programovací přístroje. Současná prostředí nabízejí škálu funkcí mimo ladění a zapisování, jako je monitorování

proměnných programu, zpracování dokumentace programu, ladění v simulačním režimu a jiné.

Při vybírání vhodného vývojového prostředí nesmíme zapomenout na to, že právě vývoj programů, jejich ladění, úpravy a dokumentování jsou nedílnou součástí výsledného projektu a tedy i jeho ceny. Prostor lze snadno spouštět ze standardních PC.

Vývojová prostředí nám nabízejí širokou škálu programovacích jazyků pro vytváření aplikací. Některé z nich umožňují i přepínání mezi jednotlivými typy a programování částí v jednom a částí v druhém typu programovacího jazyka.

Ve světě se ustálily tyto prog. jazyky:

jazyk mnemokodů - jedná se o jazyk blízký assembleru

jazyk funkčních bloků - popisuje logické operace obdobnými značkami, jako při kreslení logických schémat pro pevnou logiku. Velmi často se používá při regulaci a měření.

jazyk reléových schémat - používá se při kreslení kontaktních a reléových schémat.

grafický jazyk pro sekvenční programování - tento jazyk je velmi názorný a podporuje programátory k vyvíjení uspořádaných algoritmů.

jazyk vyšší úrovně - vývojové prostředí, které umožňuje programování úloh pomocí vyšších jazyků například podobných PASCALU. Je to třeba programovací jazyk AWL od firmy FESTO.

2.5 PLC v systémech řízení

a) dopředné řízení

Při dopředném řízení působí PA na řízený objekt jednosměrně, jen je ovládá a nekontroluje dosažený stav. Mezi řídicím systémem a řízeným objektem jsou zařazeny jen akční členy. Povel pro programovatelný automat může zadávat při tzv. **ručním řízení** člověk-operátor. Programovatelný automat zde může figurovat jen jako prostředník mezi povelů operátora a posloupnosti dílčích akcí, zajistit koordinaci povelů pro pohony s jinými akčními zásahy, kontrolu zadaných akčních zásahů apod.

PA může plnit úlohy dopředného řízení i v **automatizovaném** či **plně automatickém** systému řízení, kdy povel pro automat poskytuje nadřazený řídicí systém ve spolupráci s člověkem nebo jen nadřazený řídicí systém.

b) zpětnovazební řízení

Při zpětnovazebním řízení získává řídicí systém zpětnou informaci o stavu řízeného objektu (realizuje zpětnou vazbu, uzavírá zpětnovazební smyčku). Porovnává požadovaný stav se skutečným a podle zjištěné odchylky upravuje své akční zásahy tak, aby dosáhl požadovaného stavu nebo se mu alespoň co nejvíce přiblížil. Zpětnovazební řízení je typické pro regulační úlohy. Při použití PLC to znamená, že zadání žádané hodnoty je provedeno v číslicové formě. S číselnou informací systém pracuje při zpracování skutečné hodnoty a odchylky, ale i při výpočtech pomocných veličin potřebných k realizaci regulačního algoritmu. Řízený objekt je proto třeba doplnit o snímače, potřebné pro měření stavu sledovaných veličin (např. teploty, výšky hladiny a jiné).

Za zpětnovazební řízení můžeme považovat i logické řízení, při kterém na objekt působíme jen dvouhodnotovými povely typu „vypni - zapni“ a zpracováváme i zpětnovazební informace dvouhodnotového charakteru ve významu hlášení o vykonání povelu nebo překročení povolených hodnot (např. informace typu: „nádrž plná“, „nádrž prázdná“ apod.)

Zpětnovazební řízení může být *ruční*, *automatizované* nebo *plně automatické*.

2. 6. Programovatelný automat versus řídicí systémy

a) relé a kontakty versus programovatelné automaty

Relé, stykače a tlačítka jsou v některých případech nenahraditelné a nemělo by žádnou výhodu, bránit se jejich použití i v případech, kdy je použit programovatelný automat. Z bezpečnostních důvodů se takto realizují záložní okruhy. Nemá smysl bránit se příležitostnému vytváření logických funkcí prostým propojením kontaktů, obzvláště, pokud tím ušetříme počty vstupů a výstupů programovatelného automatu a snížíme tak jeho cenu. Rozsáhlejší funkce se kontaktní reléovou technologií již nerealizují a svěříme je důsledně programu PA.

b) regulátory versus PA

Rozdíl mezi těmito dvěma tradičními kategoriemi výrobků (regulátory a PLC) se v dnešní době postupně stírá. Obě skupiny těchto výrobků pracují číslicově. Hranice

mezi nimi je tak neostrá, že je mnohdy obtížné rozhodnout, zda je produkt ještě regulátorem nebo již programovatelným automatem. Starší typy kompaktních regulátorů (cca > 10 let) pracují analogově, nejsou schopny jakékoli komunikace s okolím (zejména s nadřazeným řídicím systémem) a jsou proto nahrazovány programovatelnými automaty.

c) personální počítače versus PA

Někdy se setkáváme s přímým řízením technologických procesů standardním personálním počítačem (PC), mnohdy umístěným přímo v technologii. Toto řešení je přinejmenším riskantní a diskutabilní. Běžný počítač kategorie PC je produkt spotřební elektroniky a je konstruován pro provoz v prostředí domácností, laboratoří a kanceláří. V drsných průmyslových podmínkách mnohdy selhává, bývá málo spolehlivý, je citlivý na rušení a přepětí a nemá potřebnou životnost. Problémy vznikají už s pouhým připojením většího počtu vstupních a výstupních vodičů a jejich odrušením. Přímé řízení počítačem je dnes účelné jen v laboratorních podmínkách, pro potřeby výuky a řízení laboratorních a modelových úloh. Pro náš konkrétní případ, je možné použít pro řízení osobní počítač vybavený měřicí kartou, avšak laboratoř má sloužit k názornému seznámení s moderními prostředky automatického řízení používaných běžně v průmyslu, kam využití PC a měřících karet nepatří.

Průmyslové počítače se někdy používají pro přímé řízení strojů a technologií, někdy jen v roli inteligentního operátorského panelu nebo komunikačního adaptéru. Problém při jejich nasazování je vysoká cena. Jsou účelné jen tam, kde je zdůvodněna. Zejména při archivaci a zpracování velkých objemů dat, při využití standardních programových produktů, při využívání výkonných komunikací, při řešení geometrických a jiných výpočetně náročných úloh. Je-li v průmyslových podmínkách použití PC nezbytné, pak je striktně nutno použít průmyslový typ. Standardním řešením je použití distribuovaného systému, kdy je osobní počítač použit na dispečerském pracovišti a do drsného prostředí jsou přesunuty PA.

Může sloužit digitální PC versus PA.
Když má být specifické slovní a
slovní použití !!!

2. 7. *Výhody PA*

a) rychlá realizace

Hlavní předností PA je možnost rychlé realizace systému. Technické vybavení nemusí uživatel vyvíjet. Stačí navrhnout a včas objednat vhodnou sestavu modulů PA pro danou aplikaci, vytvořit projekt, napsat a odladit uživatelský program, realizovat a uvést do chodu.

b) spolehlivost, odolnost, diagnostika

Technické vybavení PA je navrženo tak, že jsou extrémě spolehlivé i v drsných průmyslových podmínkách, jsou odolné proti rušení i poruchám, vyznačují se robustností a spolehlivostí. PA bývají vybaveny i vnitřními diagnostickými funkcemi, které průběžně kontrolují činnost systému a včas zjistí případnou závadu, lokalizují ji, ošetří a usnadní její odstranění.

to mají i PPC !!!

c) snadná přizpůsobitelnost řešení

Málokdy se stane, že první varianta řešení je i tou poslední a konečnou. Představy zadavatele a koncového uživatele se postupem vývoje mění a rozšiřují. Mnohé nedostatky zadání a zvoleného postupu řešení se projeví většinou ve fázi finalizace zakázky. Dodatečné požadavky a zadání nových funkcí vznikají i po mnoha měsících či letech rutinního provozu.

d) schopnost komunikace

K velkým výhodám PA patří jejich schopnost komunikovat s nejrůznějšími systémy a zařízeními jak v podřízené úrovni, což je oblast senzorů, měřících zařízení a akčních členů, tak i v souřadné úrovni s ostatními programovatelnými automaty nebo jinými řídicími systémy.

2. 8. Nevýhody PA

a) prodloužení odezvy

Řídicí systémy s pevnou logikou se od systémů s PLC se liší v době odezvy, t.j. v době, za kterou zareagují výstupy na změnu na vstupech systému. V pevné logice jsou

členy trvale aktivní, algoritmus systému se realizuje paralelně a ve spojitém čase. Odezva na změnu vstupů je dána jen celkovým zpožděním logických členů v nejdelší větvi. U integrovaných obvodů to bývají řádově nanosekundy až mikrosekundy, u reléových systémů jednotky, desítky, někdy i stovky milisekund.

Odezva PLC bývá delší a je dána dobou průchodu programu. Závisí na rychlosti procesoru a na délce aktivní větve programu. Nabývá hodnot v řádu jednotek až desítek milisekund, což pro běžné aplikace postačuje. Je však třeba s touto skutečností počítat, aby nebyla příčinou nečekaných chyb.

b) nespojitost v čase

Dalším znakem programovatelných systémů je časová nespojitost zpracování. Algoritmus je vykonáván cyklicky, vždy jen v určitých okamžicích. Uvnitř intervalu, mezi okamžiky aktivace, systém nereaguje na změny vstupních hodnot. Tuto skutečnost je třeba respektovat při návrhu a programování systému, jinak může být příčinou chyb, ztráty krátkého vstupního impulsu apod.

c) posloupnost zpracování

Program PLC je vykonáván v pořadí, v jakém je zapsán, nikoli v pořadí „toku signálu“ v odpovídajícím logickém schématu. Je-li možné zapsat PLC program sousledně s tokem signálů (aby pořadí sledovalo tok signálů směrem od vstupů k výstupům), nebývají problémy. U složitých a nepřehledných logických funkcí se to vždy nepodaří. V lepším případě je následkem prodloužení doby odezvy systému, v horším případě může být následkem i chybná funkce programu nebo jeho zdánlivě nahodilé chyby.

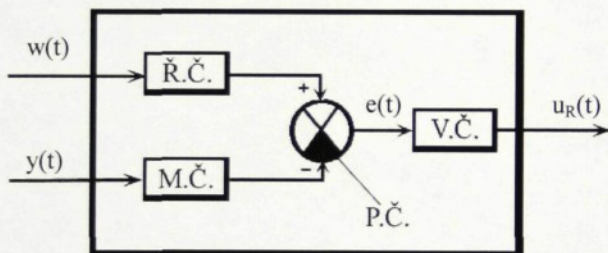
*Kapitoly 2.6, 2.7 a částí 2.8
jsou velmi zavádějící. Může
soust PC proti PLC, když každý
z nich je určen pro něco jiného*

3. REGULÁTORY

3.1 Rozdělení regulátorů

Regulátorem je nazýváno zařízení v regulačním obvodu, kterým se uskutečňuje proces regulace. Do regulátoru zahrnujeme obvykle, kromě regulované soustavy, všechny členy regulačního obvodu. Podstata činnosti regulátoru spočívá ve vyhodnocení regulační odchylky $e(t) = w(t) - y(t)$ jako vstupního signálu. Snaha je docílit, aby odchylka $e(t)$ byla eliminována, či byla co nejmenší. Podle přívodu energie dělíme regulátory na **přímé** a **nepřímé**.

Přímé regulátory (obr. 3. 1.) nepotřebují ke své činnosti vnější přívod energie. Energie snímače stačí k ovládání akčního členu. Patří mezi ně např. Wattův odstředivý regulátor.



Ř. Č. - řídicí člen

M. Č. - měřicí člen

P. Č. - porovnávací člen

V. Č. - výstupní člen

vstupy: $w(t)$ - řídicí veličina

$y(t)$ - regulovaná veličina

výstup: $u_R(t)$ - akční veličina

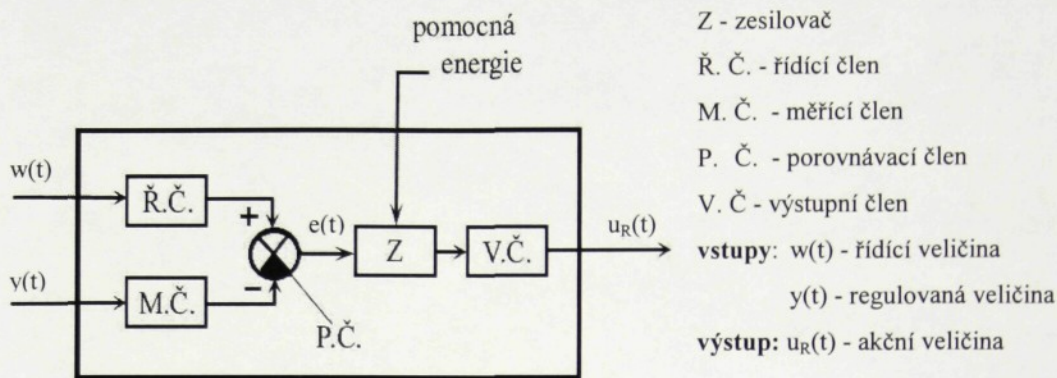
*co data ai
jaka je funkce
řídícího členu*

Obr. 3. 1. Blokové schéma přímého regulátoru

Nepřímé regulátory (obr. 3. 2.) se používají v těch případech, kdy signál od snímače není dostatečně silný k přestavění regulačního orgánu. Regulátor potřebuje pro svoji činnost pomocnou energii.

Podle druhu použité pomocné energie lze nepřímé regulátory rozdělit na:

- a) pneumatické
- b) hydraulické
- c) elektrické



Obr. 3. 2. Blokové schéma nepřímého regulátoru

Podle průběhu výstupní veličiny se regulátory dělí na **spojité** a **nespojité**. U spojitých regulátorů jsou všechny veličiny po částech spojitě v čase. U nespojitých regulátorů některý člen pracuje nespojitě.

Dále lze regulátory rozdělit podle použití: a) regulátory teploty

b) regulátory otáček

c) regulátory tlaku

d) regulátory množství atd.

je to? ženo?

to není správně vyjádřeno!

Regulátory jsou složeny z hlavních prvků:

- **měřicí člen** - snímač (čidlo)

- **porovnávací člen**

zjišťuje regulační odchylku $e(t)$, tj. snímačem se měří skutečná hodnota regulované veličiny a v porovnávacím členu se počítá její rozdíl od žádané hodnoty.

to patří do měřicího členu!!

- **řídicí člen**

- **ústřední člen** - převodník

- zesilovač

- zpětná vazba

kde je to na obrázku?

zpracovává regulační odchylku podle matematického zákona, který je určen např. P, PD, PID.

- **akční člen** - pohon

- regulační orgán

kde je to na obrázku?

odpovídajícím způsobem působí na řízenou soustavu.

3.2. Dynamické vlastnosti spojitých regulátorů

Použijeme-li značení podle obr. 3. 3. můžeme dynamické chování regulátoru popsat lineární integrodiferenciální rovnicí (3. 2. 1.).

$$\dots + T_2^2 u''(t) + T_1 u'(t) + u(t) = r_0 e(t) + r_{-1} \int_0^t e(\tau) d\tau + r_1 \frac{de(t)}{dt} \quad (3. 2. 1.)$$

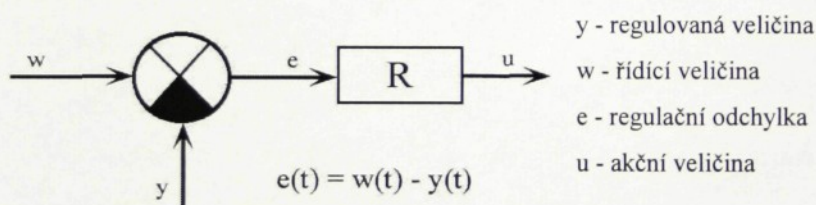
kde $r_0 e(t)$ je proporcionální složka regulátoru

$r_{-1} \int_0^t e(\tau) d\tau$ je integrační složka regulátoru

$r_1 \frac{de(t)}{dt}$ je derivační složka regulátoru

$T_2^2 u''(t), T_1 u'(t)$ jsou zpožďující členy

Jde o popis chování proporcionálně – integračně – derivačního regulátoru se zpožďujícími členy, neboli skutečného PID-regulátoru.



Obr. 3. 3. Zjednodušené blokové schéma regulátoru

Laplaceovou transformací rovnice (3. 2. 1.), za splnění nulových počátečních podmínek, získáme přenos skutečného PID regulátoru. (3. 2. 2.).

$$F_R(p) = \frac{U(p)}{E(p)} = \frac{r_0 + \frac{r_{-1}}{p} + r_1 p}{1 + T_1 p + T_2^2 p^2 + \dots} = \frac{r_0 \left(1 + \frac{1}{T_I p} + T_D p \right)}{1 + T_1 p + T_2^2 p^2 + \dots} \quad (3. 2. 2.)$$

kde r_0 je proporcionální konstanta regulátoru

r_{-1} je integrační konstanta regulátoru

r_1 je derivační konstanta regulátoru

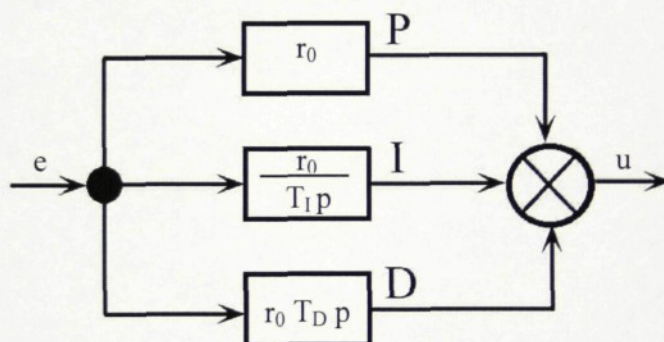
$T_I = \frac{r_0}{r_{-1}}$ je integrační časová konstanta regulátoru

$T_D = \frac{r_1}{r_0}$ je derivační časová konstanta regulátoru

Pro nulové časové konstanty zpožďujících členů dostaneme přenos ideálního PID-regulátoru (3. 2. 3.).

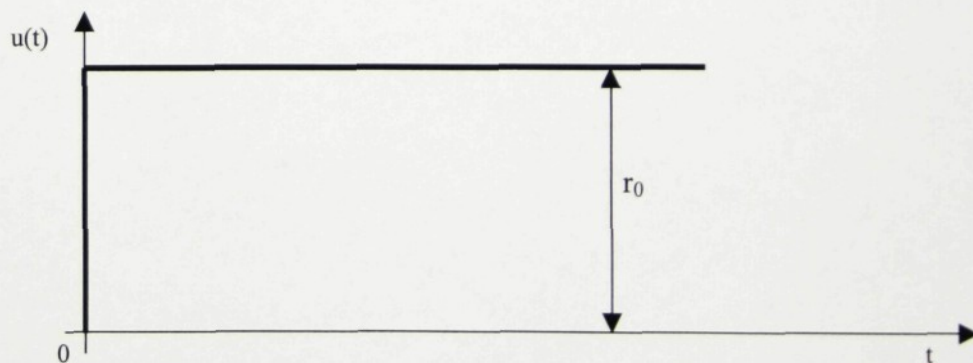
$$F_R(p) = r_0 \left(1 + \frac{1}{T_I p} + T_D p \right) \quad (3. 2. 3.)$$

Podle dynamických vlastností rozdělujeme regulátory na proporcionální(P), integrační(I), a derivační(D). Dále pak na jejich kombinace PI, PD a PID. Tyto regulátory můžeme uvažovat jako ideální nebo skutečné (se zpožďujícími členy).



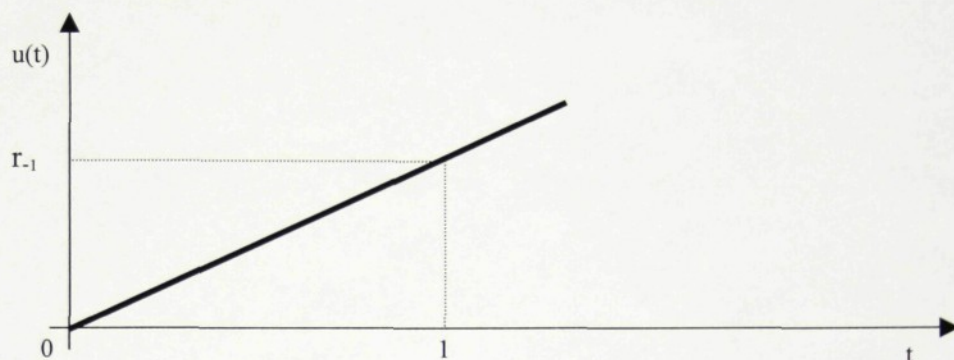
Obr. 3. 4. Blokové schéma ideálního PID regulátoru

1. **Proporcionální regulátor** – je velmi jednoduchý, levný a stabilní. Jeho nevýhodou je, že pracuje s trvalou regulační odchylkou.



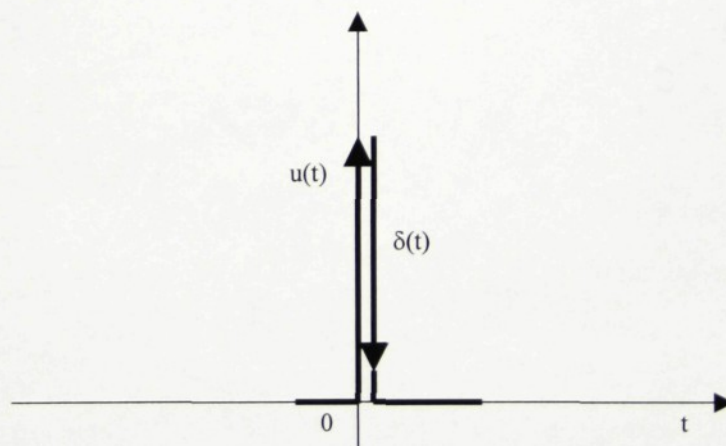
Obr. 3. 5. Přechodová charakteristika ideálního proporcionálního regulátoru

2. **Integrační regulátor** – v uzavřeném regulačním obvodu pracuje pouze s přechodovou regulační odchylkou. Regulační pochod se ustálí tehdy, když je regulační odchylka $e(t)$ rovna nule. Nevyhoví podmínkám stability regulačního obvodu, když se reguluje astatická soustava.



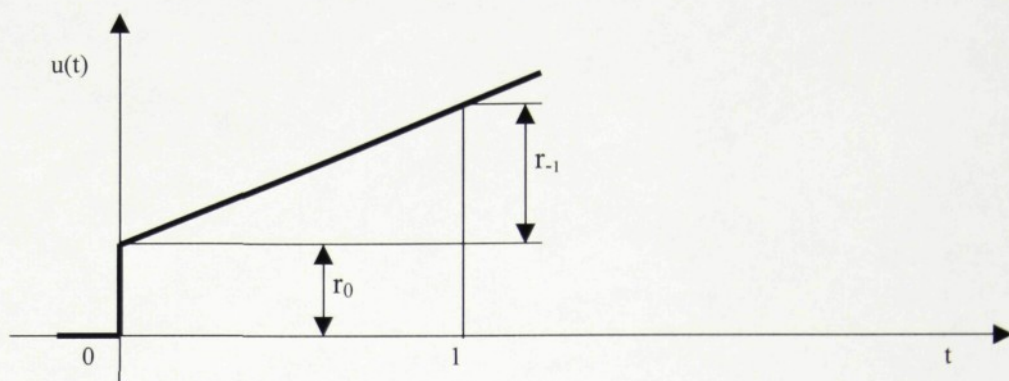
Obr. 3. 6. Přechodová charakteristika ideálního integračního regulátoru

3. **Derivační člen** – není schopen samostatné funkce jako regulátor připojený k soustavě, protože vstupním signálem je derivace regulační odchylky, nepracuje tedy s velikostí regulační odchylky $e(t)$. Tento člen je schopen pouze společné práce s regulátory P nebo PI.



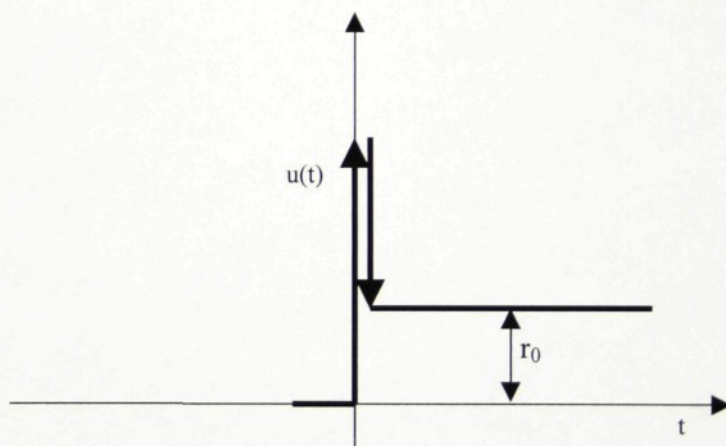
Obr. 3. 7. Přechodová charakteristika ideálního derivačního členu

4. **PI-regulátor** – v uzavřeném regulačním obvodu odstraňuje trvalou regulační odchylku. Zlepšuje stabilitu regulačního pochodu oproti použití samostatného I-regulátoru. V počátku regulačního pochodu převládá vliv proporcionální složky, s narůstajícím časem začíná převládat vliv integrační složky.



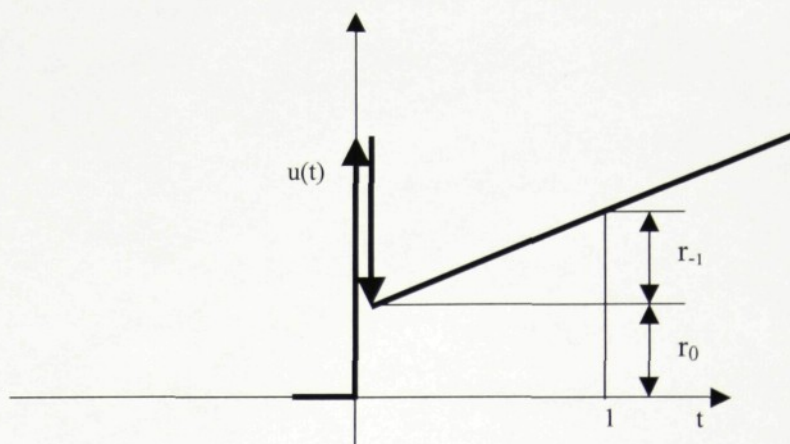
Obr. 3. 8. Přechodová charakteristika ideálního PI-regulátoru

5. **PD-regulátor** – zlepšuje stabilitu regulačního pochodu ve srovnání s použitím čistě P-regulátoru. Je tedy možné pracovat s vyšším zesílením regulátoru a tedy menší trvalou regulační odchylku, než je tomu při použití pouze P-regulátoru



Obr. 3. 9. Přechodová charakteristika ideálního PD-regulátoru

6. **PID-regulátor** – v uzavřeném regulačním obvodu odstraňuje vlivem integrační složky trvalou regulační odchylku a vlivem derivační složky zlepšuje stabilitu regulačního obvodu. V počátku přechodového děje převládá derivační složka regulátoru, s narůstajícím časem se stává převládající integrační složka regulátoru.



Obr. 3. 10. Přechodová charakteristika ideálního PID regulátoru

4. TOMPACK

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

4. 1. 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. Základní DDE server, UniServer, je též produktem firmy ProjektSoft.

Součástí TomPacku je i skriptovací jazyk, který je podmnožinou jazyka C. TomPack obsahuje řadu funkcí, mezi jinými i 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, aby obsluha bez znalostí hesla mohla vstoupit do operačního systému.

Datové soubory generované TomPackem jsou ve formátu Microsoft Acces 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 třeba.

4. 2. 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 běžet stále.
- Na základě požadavku klientů, připojených DDE komunikací, bere data PLC a posílá je klientovi.
- Při požadavku klienta o zápis do PLC, tento povel provede.

TPServer:

- Musí běžet stále.
- Na základě požadavků od TPView 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 předá změny do DDE serveru, potažmo PLC.

TPView:

- Běží pouze v případě, je-li to potřeba (např. 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 umožňuje jeho správu.

4. 2. 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.

Podrobněji o DDE

DDE server (Dynamic Data Exchange) slouží pro komunikaci mezi programy v prostředí Windows. Každá předávaná hodnota má svou adresu a formát. Nejobecnějším je textový formát.

Komunikace se odehrává mezi DDE serverem a DDE klientem, přičemž jeden server může obsluhovat více klientů, které 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“, „Topic name“, „Item name“.

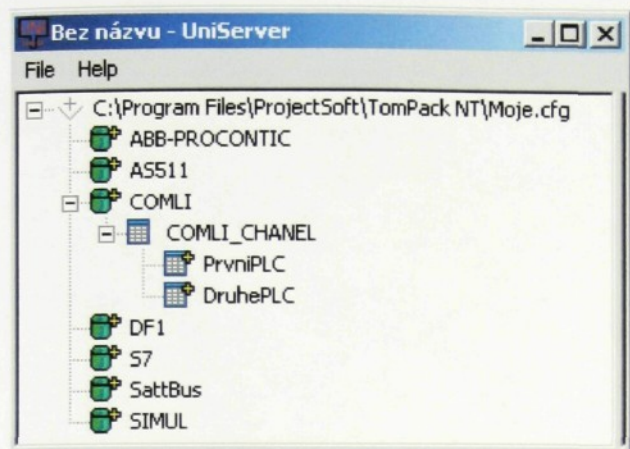
- **Service name** - Někdy též označující jako „Application name“
- většinou je to jméno programu bez přípony .exe
- **Topic name** - jméno souboru hodnot (např. TAB1 - jméno tabulky
v EXCELU) *prac v Excelu*
- **Item name** - jméno prvku (např. R1C2 - první sloupec a druhý řádek TAB1).

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 zvyklosti DDE serveru.

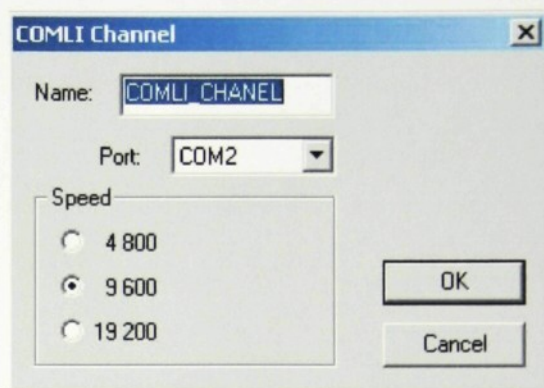
Při navazování komunikace nejprve klient pošle požadavek všem aplikacím s adresou Service/Topic (INITITE zpráva) a server, který má tuto adresu, klientovi odpoví (odpovědní INITIATE zpráva). Potom už klient sděluje přímo svému serveru, který Item chce posílat (ADVISE zpráva) a server mu posílá hodnotu 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 dokončení celé komunikace si klient a server ještě vymění TERMINATE zprávu o ukončení.

Příklad DDE adresace

Jako DDE server využijeme součást TomPacku - UniServer (obr. 4. 1.), ke kterému jsou připojeny dva automaty s adresami 5 a 6, oba přes sériovou linku COM1. V serveru nejprve musíme vytvořit COM1_CHANNEL (obr. 4. 2.) pro komunikaci přes COM1 protokol.



Obr. 4. 1. Prostředí UniServer



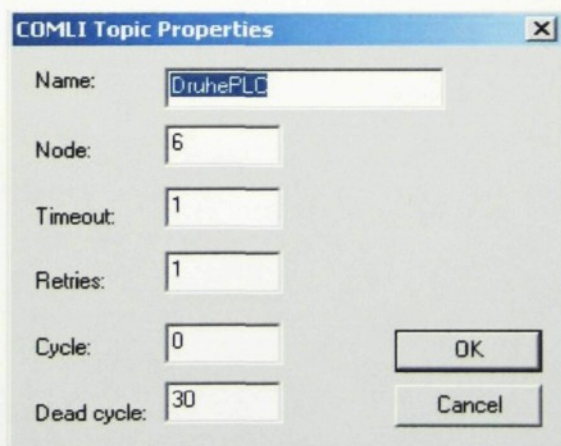
Obr. 4. 2. Prostředí UniServer - vytvoření COM1_CHANNEL pro COMLI protokolu

V nově vytvořeném kanálu potom vytvoříme dvě PLC - jedno s hodnotou Node 5 (obr. 4. 3.) 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*



Obr. 4.3. Prostředí UniServer - vytvoření spojení pro první PLC v COMLI protokolu

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

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 uchovány v souboru.

4. 2. 2. TPSTServer

TPSTServer 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 serverem, 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 TPSTServeru 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 TPSTServer otevřen žádný projekt, nelze se k němu přihlásit. Je však samozřejmě možné poslat projekt z TPConfig.

Do pracovního adresáře si TPSTServer 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), soubory historických hodnot (s příponou. jou) a soubor .ldb vytvořené databázovými funkcemi.

4. 2. 3. TPView

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

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

Byl jsem poděkován

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

- Dostatečný HW, který musí umožnit spuštění síťového operačního systému Windows NT 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 `~TPS~.tp`) logy (`ViewSys.log`, `ViewSys.lo1`) a soubor `.ldb` vytvořený databázovými funkcemi.

4. 2. 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).

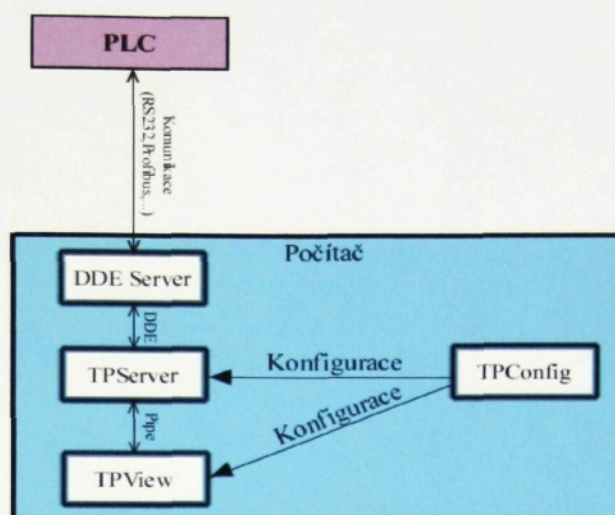
TPConfig rozešle hotový projekt do TPServer a do TPView. Ti si z projektu vytvoří plnohodnotnou kopii (`~TPS~.tp` a `~TPV~.tp`), se kterou pracují. Při posílání projektu je nutné určit počítač, na kterém je umístěn příslušný TPServer, resp. TPView. Jinak se předpokládá, že běží na stejném počítači jako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

4. 3. *Možné konfigurace pro realizaci vizualizace*

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

a) triviální konfigurace

Celý projekt může být složen například z jednoho DDE serveru, TPServeru a jednoho TPView běžících na jednom počítači. TPConfig se spouští pouze v případě konfigurace TPServeru a TPView. Zobrazení takové struktury je na obr. 4. 4.

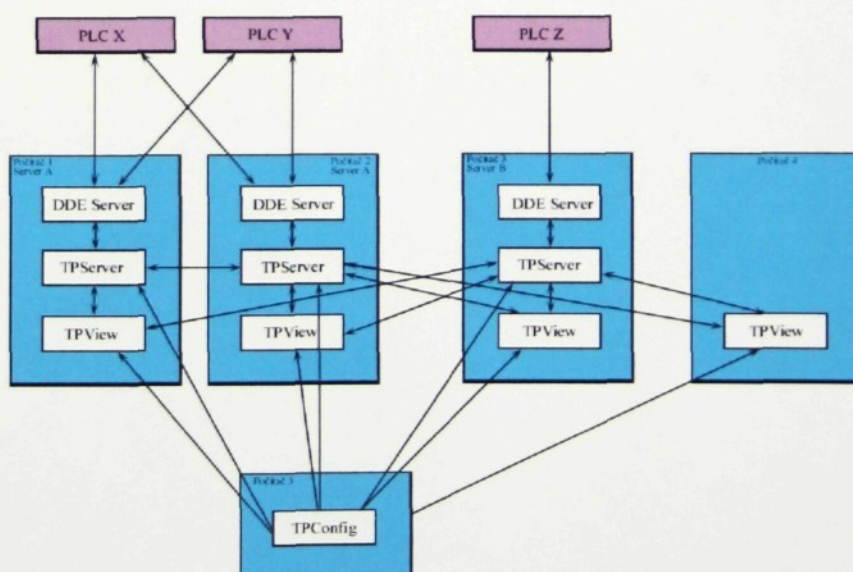


Obr. 4. 4. Zobrazení triviální konfigurace systému TomPack

b) složitější konfigurace

Konfigurace systému může být podstatně složitější, jako třeba na obr. 4. 5. V tomto případě je aplikace složena ze dvou serverů. Server A běží na počítačích 1 a 2, ty jsou 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 se používá pouze ke konfiguraci celého systému.



Obr. 4.4. Zobrazení složitější konfigurace systému TomPack

4. 4. Typy proměnných v TomPacku

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

- **Process** - Proměnná je v automatu a TPServer uchová 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). To je proto, aby se na obrazovce např. změněný 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 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 uchována 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.

4. 5. Složky proměnných

Každá proměnná (kromě typu View) je 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 uvádí následující tabulka.

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
Proměnná String				
Složka	Typ	Atribut	Druh	Popis
.Valid	bool	R	virtuální	zda je proměnná platná
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. 1. Složky proměnných

5. ÚVOD DO PROJEKTU

5. 1. Laboratoř aplikované kybernetiky

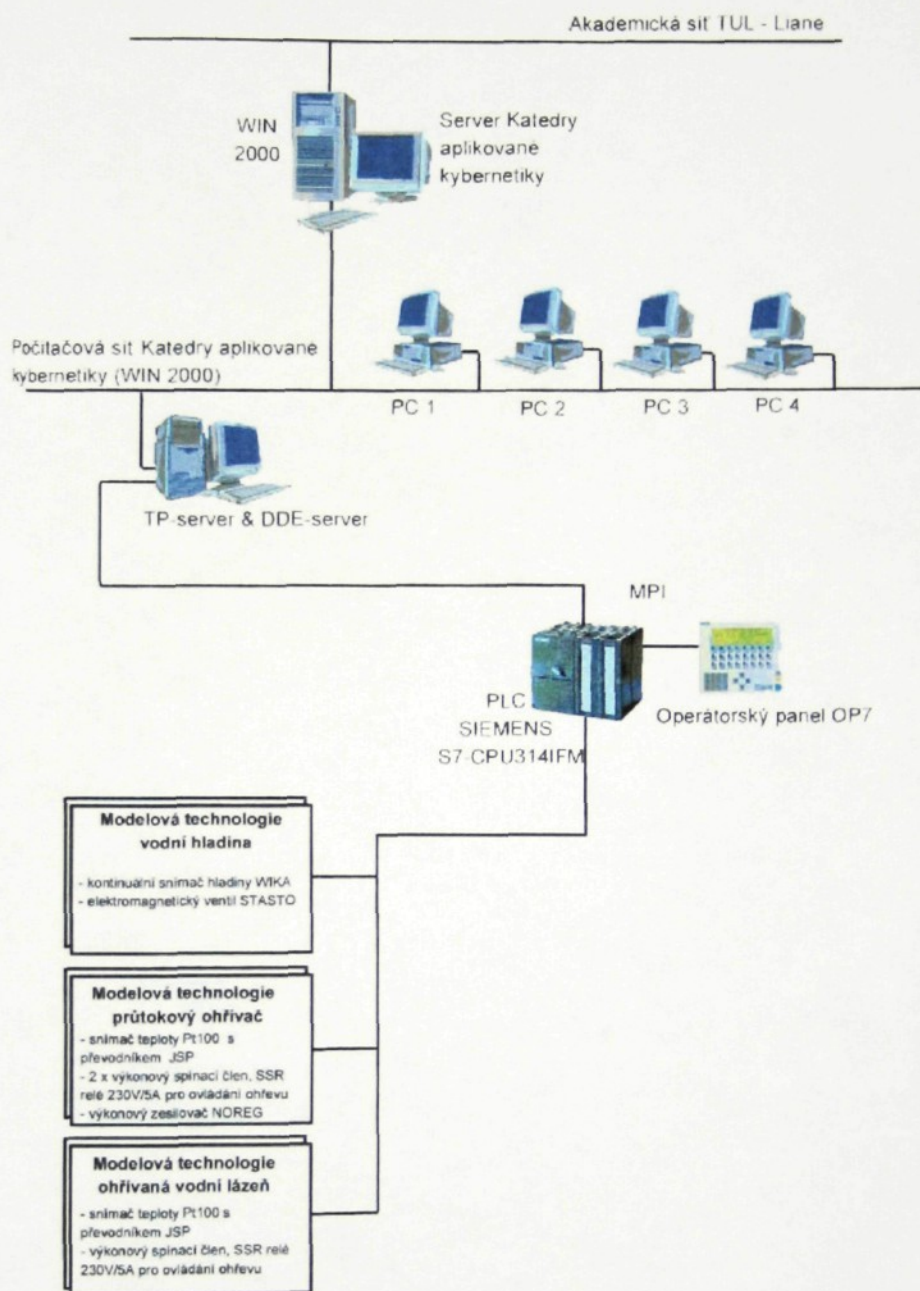
Nově koncipovaný systém laboratorních úloh musí splňovat požadavky na modularitu, snadnou rozšiřitelnost a přizpůsobitelnost nejrozličnějším nárokům, které mohou vzniknout v budoucnu. Dále je kladen velký důraz na komunikaci mezi úrovní řízení laboratorního procesu, řízení procesů a vizualizací. V budoucnu je zamýšlena výměna dat na úrovni akademické počítačové sítě LIANE.

Modelové úlohy budou řízeny programovatelným logickým automatem, což je vzhledem k rozsáhlosti navrhované laboratoře a požadované modularitě nejoptimálnější způsob řízení s ohledem na kapitolu (2. 6.). Schéma uspořádání laboratoře, modelových technologií a jejich řízení je na obr. 5. 1. Na nejnižší úrovni jsou zobrazeny senzory a akční členy jednotlivých modelových technologií, které jsou připojeny na vstupy a výstupy programovatelného logického automatu. Pro úlohy je použito PLC firmy SIEMENS, Siemens S7 CPU314 IFM s operátorským panelem OP7. Logický automat je modulární PLC, který umožňuje připojit velké množství funkčních technologických modulů, např. modul pro řízení servomotoru, modul pro řízení krokového motoru, atd. Logický automat je připojen přes rozhraní MPI (SIEMENS) k osobnímu počítači (PC), na kterém běží software (DDE - server), zajišťující komunikaci mezi PLC a vizualizačním softwarem (TOMPACK). Tento počítač je součástí katedrové počítačové sítě a pracuje s operačním systémem WINDOWS2000. Ve stejné síti jsou zapojeny další čtyři osobní počítače, rovněž s operačním systémem WINDOWS2000, kterých je využito pro ovládání dané technologie a vizualizace technologického procesu. Počítačová síť katedry aplikované kybernetiky je součástí akademické sítě Liane Technické univerzity v Liberci.

5. 2. PLC Siemens Simatic S7 - 300

Programovatelných automatů Simatic pracuje na světě úspěšně více než 1, 5 milionu. Jsou oblíbeny především pro svou robustnost, spolehlivost a kvalitu. V Evropě patří k nejrozšířenějším ze všech PA.

Je možno vybírat z komponent a systémů dolního výkonnostního rozsahu,



Obr. 5. 1. Schéma konfigurace řídicího systému laboratoře aplikované kybernetiky

systemů středního i nejvyššího rozsahu. Rodina automatů Simatic S7 je členěna do tří tříd podle potřebné výkonnosti, a to na následující třídy:

- SIMATIC S7-200 - mikro PLC
- SIMATIC S7-300 - dolní rozsah výkonnosti
- SIMATIC S7-400 - střední a nejvyšší rozsah výkonnosti

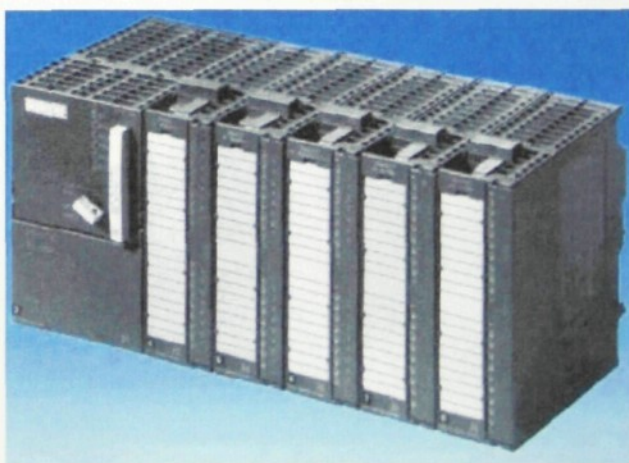
V rodině automatů Simatic S7 je patrná redukce počtu různých typů automatů a naopak zvýšení počtu volitelných centrálních jednotek, vstupně/výstupních modulů, funkčních technologických modulů a komunikačních procesorů

SIMATIC S7-300 je programovatelný automat třídy „mini“, jak s ohledem na rozměry, tak i na umístění ve výkonostním spektru programovatelných automatů. Z hlediska implementovaného souboru instrukcí se však řadí mezi automaty vyšší třídy. Má modulární výstavbu, která nevyžaduje nucené chlazení, kompaktní rozměry, možnost rozšiřování ve více řadách, může být uspořádán centrálně i decentrálně, může být vybudován až z pěti výkonostně odstupňovaných centrálních procesorových jednotek a má integrované komunikační schopnosti.

Je výhodný pro aplikace v řízení nejrozumnějších strojů, výrobních linek, balících a manipulačních zařízení, zemědělských zařízení nebo jednodušších chemických procesů.

Simatic S7-300 má velké konfigurační možnosti. Moduly systému jsou upevněny na speciální profilované liště a jsou jistěny proti záměně. Maximální sestavu systému tvoří 32 modulů ve čtyřech řadách po maximálně osmi kartách v řadě. K rozdělení systému do více řad slouží příslušný spojovací modul. Automat je napájen stejnosměrným napětím 24 V.

Pokud se týká procesorových jednotek, je možno vybírat z pěti typů, od CPU 312 IFM s vestavěnými binárními i analogovými vstupy, přes CPU 313, CPU 314 IFM, CPU 315 až po nejvýkonnější CPU 318-2 DP s integrovanou sběrnici Profibus-DP. Moduly binárních vstupů/výstupů mají hustotu až 32 V/V na jeden modul, čímž značně



Obr. 5. 2. PLC Siemens Simatic S7-300

šetří zastavěný prostor. Jsou k dispozici v provedení stejnosměrném i střídavém, výstupy pak v provedení tranzistorovém i reléovém. Moduly analogových vstupů/výstupů mají rozlišení 9 - 14 bitů a znaménko. Typ měřené veličiny může být proud, napětí nebo odpor a rozsah měřené veličiny se volí ve dvou krocích - nahrubo, pomocí hardwarových prostředků a jemně pomocí softwarových prostředků v rámci

programovacího prostředí STEP 7. Lze zvolit modul, na němž jsou umístěny analogové vstupy i výstupy. Konkrétní vodiče od prvků pro tvorbu vstupů (spínací prvky, senzory) je možno připojit přímo na svorkovnici na čelním konektoru modulu nebo přes vzdálenou svorkovnici, která je s modulem vstupů spojena kabelem o délce až 30m.

Pro propojování S7-300 s inteligentními zařízeními slouží komunikační procesory:

- Pro sériovou komunikaci s možností volného ASCII protokolu pro propojení se zařízeními jiných firem nebo s možností vlastního protokolu SIEMENS MPI pro propojení v rámci zařízení Simatic.
- Modul SINEC L2-DP, který umožňuje komunikaci v pozici řídicího nebo řízeného prvku s decentrálními periferiemi podle normy Profibus-DP, komunikaci s programovacím přístrojem, operátorskými panely nebo vizualizací, a také komunikaci s jinými automaty.

Rozhraní MPI je navrženo jako speciální programovací rozhraní pro řadu systémů Simatic S7. Současně je možné toto rozhraní využít k výměně menšího objemu dat mezi několika zařízeními z řady S7. Kromě řady S7-200, jsou všechny CPU z řady Simatic vybaveny integrovaným rozhraním MPI. Pomocí tohoto rozhraní lze jednoduše propojovat operátorské panely, ruční programátory a ostatní PLC Simatic S7. Zařízení na bázi počítačů PC lze tímto rozhraním vybavit prostřednictvím karty MPI nebo komunikačním procesorem CP 5411. Karta MPI je určena pro ISA slot PC.

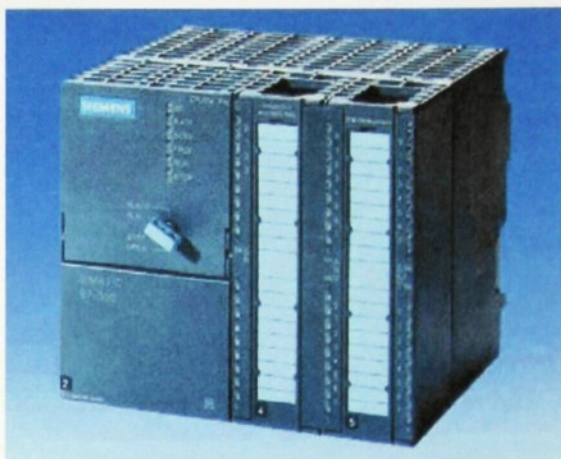
Funkční vlastnosti automatu Simatic S7-300

- Rychlé provádění instrukcí od 0,3 mikrosekundy/instrukcí
- Propracovaná diagnostika vnitřních chyb systému a chyb programového cyklu
- Možnost práce s čísly v pohyblivé řádové čárce
- Ochrana heslem orientovaná na tzv. know how protection nepřipustí v případě potřeby neautorizované kopírování nebo změny v programu PLC

5. 3. Procesorová jednotka Siemens CPU 314 IFM

Jedná se o kompaktní centrální procesorovou jednotku s integrovanými vstupy a výstupy umožňující přímé připojení k procesu a s pokročilými speciálními funkcemi.

Je vhodná pro zařízení s vysokými nároky na reakční časy. Umožňuje rychlé čítání dvěma čítači, měření frekvence a regulaci vestavěným regulátorem.



Obr. 5. 3. Siemens Simatic S7 CPU 314 IFM

CPU 314 IFM disponuje:

- Mikroprocesorem, který dosahuje rychlosti zpracování binární instrukce od 300 ns
- Dostatečně velikou operační paměti 32 kB, která umožní zapsat přibližně 10 000 instrukcí
- Vícefunkční sběrnici MPI
- Flexibilní rozšiřitelností až do 32 modulů
- Ochranou heslem
- Zásobníkem chyb, v němž je uchováno posledních 100 chyb
- Ochranou dat, která umožňuje i bez baterie uchovat data díky paměti typu NVRAM
- 20 integrovanými digitálními vstupy a 16 integrovanými digitálními výstupy
- Hodinami reálného času
- Integrovanými komunikačními schopnostmi

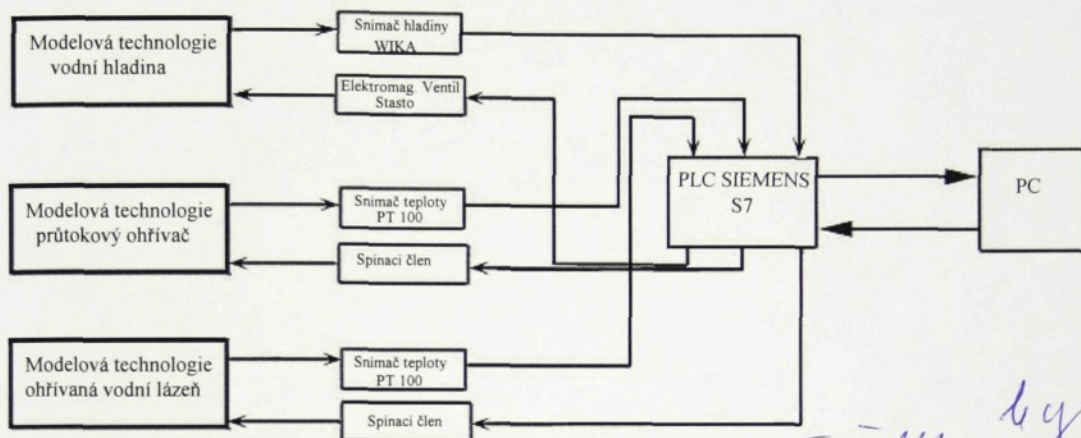
CPU 314 IFM má integrované funkce:

- Až dvě počítadla do 10 kHz, 31 bit + znaménko
- Jeden kanál pro měření frekvence do 10 kHz
- Jeden kanál pro řízení polohy
- Integrovaný PID regulátor

Ostatní technická data k CPU 314 IFM jsou uvedeny v příloze 4.

5. 4. Zapojení PLC v laboratoři

Jak již bylo dříve vidět z obr. 5. 1., pomocí PLC Siemens se budou řídit tři laboratorní procesy. K tomu nám postačí zapojit 3 analogové vstupy a 3 analogové výstupy. PLC bude řízeno pomocí nadřazeného systému. Na obr. 5. 4 je principiální schéma zapojení.



Obr. 5. 4 Principiální schéma zapojení PLC

*all process by PLC
jedna proces. proč?
proč po vidění
3 procesy každý
realizován byl
jednu jeden*

5. 5. Modelová technologie vodní nádrž

Naším úkolem je navrhnout účinné regulační algoritmy pro řízení výšky hladiny ve vodní nádrži s využitím automatu PLC Siemens Simatic S7 - 300 s operátorským panelem OP7. Zapojit PLC do obvodu a nastavit hodnoty parametrů v algoritmech tak, aby co nejlépe vyhovovaly požadavkům na regulovanou soustavu, k PLC připojit nadřazený počítač na němž běží vizualizace a archivace dat pomocí systému TomPack.

5. 6. Popis soustavy

Soustava je tvořena nádrží na vodu s trvalým odtokem přes ventil V2, který po dobu regulace zůstává na konstantní hodnotě. Regulačním ventilem RV měníme množství přitékající vody. Jedná se o nepřímé řízení elektromagnetický ventil STASTO typ 21W (obr. 5. 5.). Jde o ventil s membránovým uzávěrem s ovládacím napětím 24V= splňující četnost spínání 200 min⁻¹. Při odpojení od proudu je ventil v poloze uzavřeno. K nádrži je připojen závitovým šroubením G 1" (DN 25). Přesné

označení ventilu je STASTO 21W 4 K B 250 - D - 24/=. Ostatní technická data jsou uvedeny v příloze I.



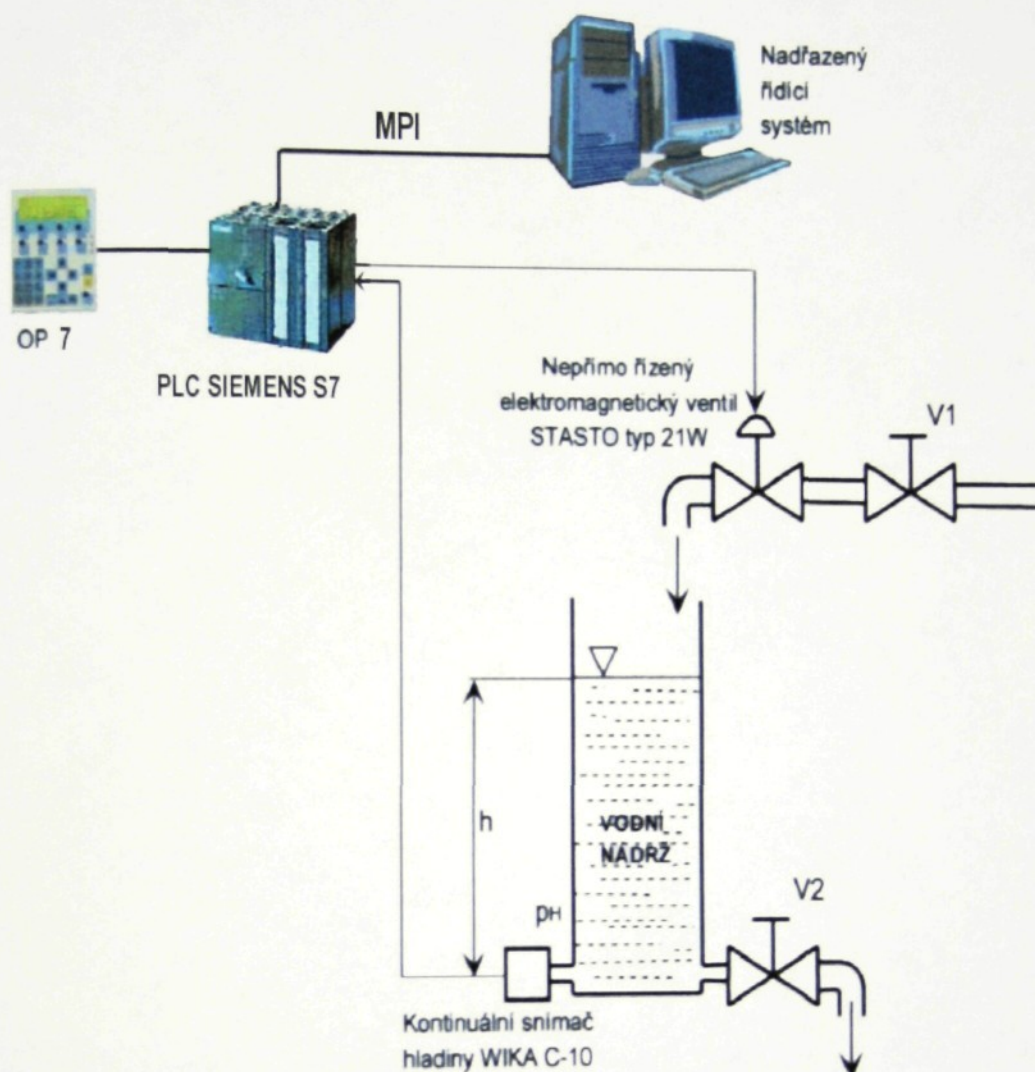
Obr. 5.5 Elektromagnetický ventil STASTO 21W



Obr. 5. 6. Kontinuální snímač hladiny WIKA C - 10

Dále je v úloze použito kontinuálního snímače hladiny WIKA C - 10 (obr. 5. 6.). Technická data ke snímači hladiny WIKA C - 10 jsou uvedeny v příloze II. Ze snímače dostáváme výstupní signál v podobě signálu proudového 4 - 20 mA.

Obr. 5. 7. Zobrazuje principiální popis soustavy. Elektromagnetický ventil i snímač hladiny jsou připojeny k PLC, které je přes rozhraní MPI (SIEMENS) připojeno k nadřazenému systému.



Obr. 5. 7. Model subsystému vodní nádrže

6. MATEMATICKÝ MODEL REGULOVANÉ SOUSTAVY

Pro realizaci úlohy v paměti počítače je potřeba vytvořit matematický popis (přenos soustavy).

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

Tato rovnice slouží pouze pro účely ukázky vyvinutého softwaru. Byla získána z předchozí diplomové práce T. Rajtra. — není uveden v literatuře !!!

Vzhledem k tomu, že TomPack neumí zacházet se spojitým systémem, musíme náš přenos diskretizovat a dostáváme následující rovnici:

- $$y_k = 0,942180y_{k-1} + 4,625624u_{k-1}$$

*píše y(k)
y(k-1)
u(k-1)
to nejsem index
all řádků dle tohoto
tam*

Pro realizaci regulačního obvodu je ještě zapotřebí popis regulátoru v diskretním tvaru,

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

kde P, T_I, T_D jsou nastavitelné parametry regulátoru. Tento popis není v aplikaci využit, protože součástí TomPacku je funkce pro regulátor typu PID.

*jak píše se
řady uvažují ??*

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, pak se rozesílá do TPServer a TPView (viz. kapitola 4. 2. 2).

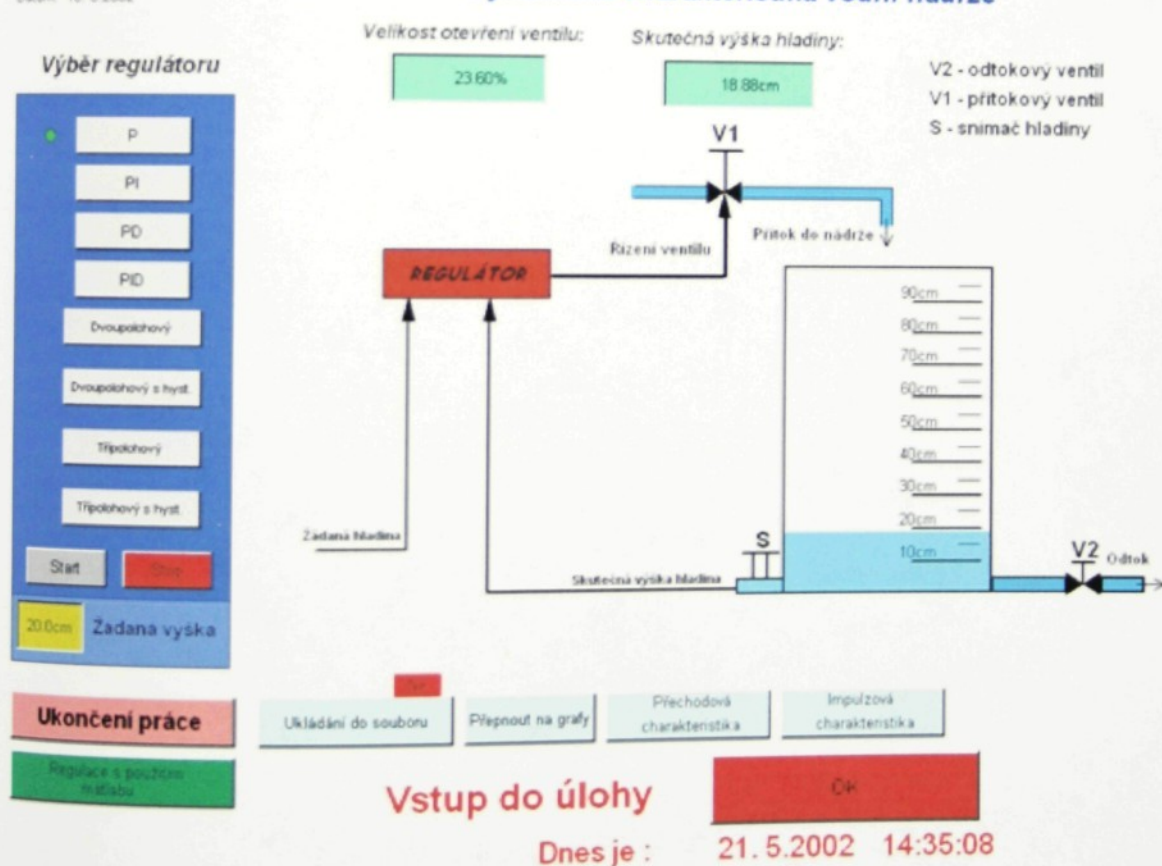
Vývojový prostředí TPConfig obsahuje kromě editoru proměnných také grafické prostředky pro tvorbu aplikace. Příkladem jsou editační okna proměnných, kreslení čar, elips, obdélníků a jiných geometrických tvarů, vkládání tlačítek, grafů atd. Kombinací těchto prvků, proměnných a logických porovnání dostáváme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

7. 1. Vizualizační SW vyvinutý pro zadanou úlohu

Na začátku vývoje tohoto SW byly základní podmínky a požadavky pro realizaci

Čas: 11:46:37
Datum: 10. 5. 2002

Dynamická charakteristika vodní nádrže



Obr. 7. 1. Vstupní okno

projektu. Podmínkou byla schopnost vyvíjeného SW komunikovat s úlohou, regulovat ji pomocí PLC přes MPI, návrh a ověření programů pro různé regulační algoritmy.

Vzhledem k tomu, že se v průběhu vyvíjení tohoto SW nepodařilo zajistit potřebné technické prostředky pro model subsystému regulace výšky hladiny (PLC, měřící a akční členy obvodu atd.), bylo zvoleno náhradní řešení.

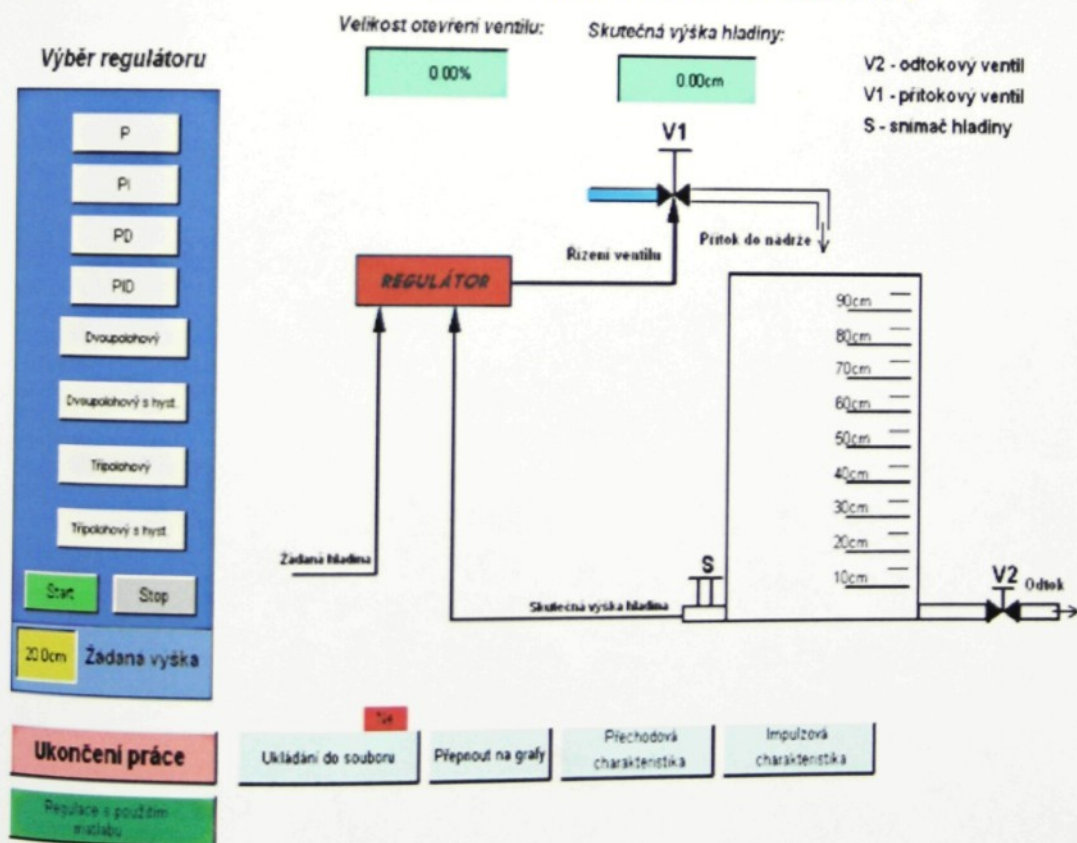
Náhradním řešením je vizualizace technologického procesu v paměti počítače s využitím regulátoru vestavěného v prostředí TomPack. V důsledku je tato situace s původním návrhem řízení pomocí PLC obdobná, protože není rozdíl, zda zavádím proměnnou do paměti počítače nebo do registru PLC.

Pro správný chod počítače je na PC vytvořen uživatel „**tompack**“ s přístupovým heslem „**tompack**“. Do spouštěcího scriptu PC je dán soubor TPView.exe a TPServer.exe. Po spuštění TPView se zobrazí vstupní okno obr. 7. 1.

Po zmáčknutí tlačítka OK, se uživateli zobrazí další okno, ve kterém již může ovládat danou úlohu obr. 7. 2.

Čas: 14.29.37
Datum: 21. 5. 2002

Dynamická charakteristika vodní nádrže



Obr. 7. 2. Hlavní okno pro regulaci

Obsluha si může vybrat libovolný proces regulace z dostupných regulátorů, včetně nastavení jeho parametrů. Pro snadné sledování regulované veličiny poslouží zobrazení technologie s grafickým znázorněním žádané veličiny. Detailnější pohled na změny skutečné veličiny a akčního zásahu slouží číselné zobrazení v horní části okna nebo se pomocí tlačítka „přepnout na grafy“ přepnout na zobrazení grafické. Uživatel si také může vybrat, zda chce měřit přechodovou. Pokud zvolí tuto variantu, naměřená data se ukládají do souboru prechodovka.txt. V rámci regulace je možnost ukládat měření do souboru s příponou txt, který si uživatel sám zvolí. Hodnoty v těchto souborech jsou uloženy v textové podobě a lze s nimi dále pracovat.

Tlačítko “Regulace s použitím Matlabu” je určeno pro komunikaci mezi TomPackem a Matlabem. Pokud není zapnuté probíhá realizace úlohy pomocí matematického modelu. Po zapnutí je realizace prováděna pomocí blokového schématu v programu Matlab. Aby proces mohl probíhat, je zapotřebí v Matlabu spustit soubor **komunikace.m**.

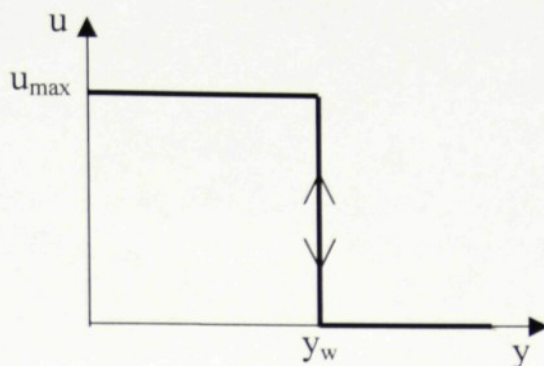
Pro regulaci pomocí P, PD, PI, PID regulátorů je použita vlastní funkce TomPacku. Pro dvoupolohovou, třípolohovou, dvoupolohovou s hysterezí a třípolohovou s hysterezí musely být napsány samostatné algoritmy, které jsou popsány níže.

Ukončení práce se SW je pomocí tlačítka “Ukončení práce“, kdy se uživateli objeví vstupní okno jako při spuštění obr. 7. 1.

7. 2. Návrh algoritmů pro realizaci dvoupolohové regulace

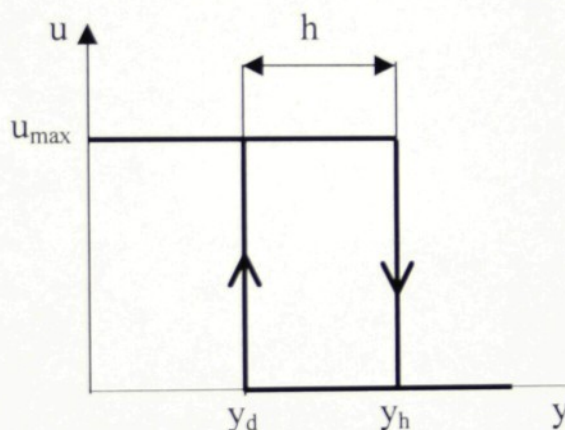
Dvoupolohové regulátory patří společně i s vícepolohovými regulátory do skupiny tzv. nespojitých regulátorů. Ve skutečnosti to tedy znamená, že akční veličina se nemění spojitě, ale může nabývat pouze omezený počet hodnot, v tomto případě dvou. Charakteristika dvoupolohového regulátoru s nulovou hysterezí je na obr. 7. 3.

Z charakteristiky je jasné, že je pokud regulovaná veličina menší než žádaná hodnota, potom akční zásah je maximální a v opačném případě je akční zásah nulový.



Obr. 7. 3. Dvupolohový regulátor s nulovou hysterezí

Charakteristika dvupolohového regulátoru s hysterezí je znázorněno na obr. 7. 4.



Obr. 7. 4. Dvupolohový regulátor s hysterezí

Použití tohoto typu bývá častější, neboť hysterezí omezíme počet sepnutí akčního členu, čímž většinou prodloužíme jeho životnost. Tato výhoda však přináší regulačnímu pochodu zmenšenou přesnost. Proto také velmi záleží na nastavení šířky necitlivosti. Pro úplnost se však musíme uvědomit, že velikost periody vzorkování je ve skutečnosti také jakási hystereze, protože pokud zvolíme větší periodu vzorkování, tak již máme jisté pásmo necitlivosti zajištěno a není potřeba používat hysterezi. Realizace algoritmu dvupolohového regulátoru je vyjádřena v **programu 1**. Realizace programu dvupolohového regulátoru s hysterezí je vyjádřena v **programu 2**. Hodnoty w_1 a h zadává obsluha v dialogovém okně a program je načítá z paměti počítače. Proměnná y_1 ,

*co h je mezi přechodem
regulačního pochodu*

při použití matematického modelu, se získává přímo z rovnice. Pokud proces probíhá pomocí blokového schématu, je $y1$ načítáno ze souboru $y1.txt$.

Program 1:

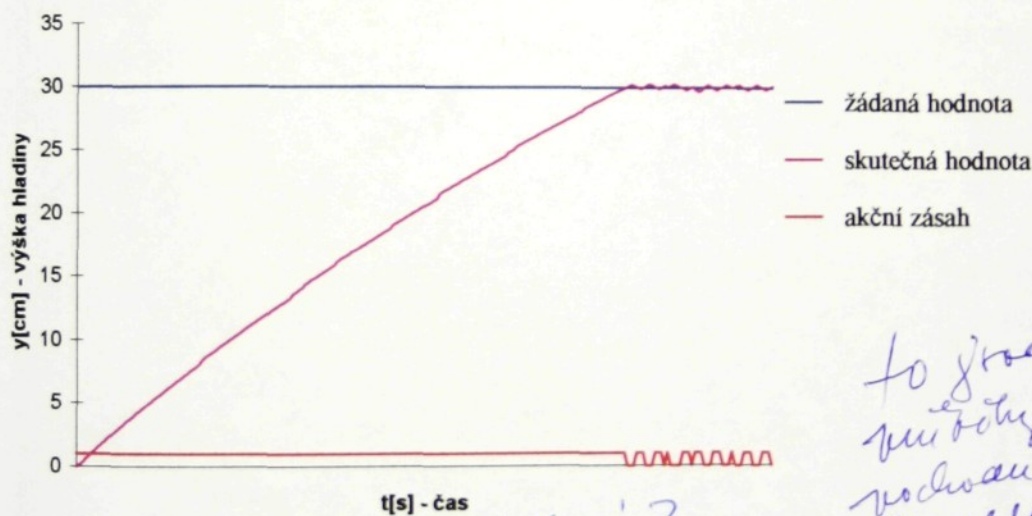
```
if( w1 > y1 )
    u1 = max;
else
    u1 = min;
```

Program 2:

```
if( w1 + h < y1 )
    u1 = min;
else if( w1 - h > y1 )
    u1 = max;
```

*To all nejste
programy, jemu
je to číslu*

Kde $w1$žádaná hodnota
 $y1$skutečná hodnota
 $u1$akční zásah
 hzvolená hystereze

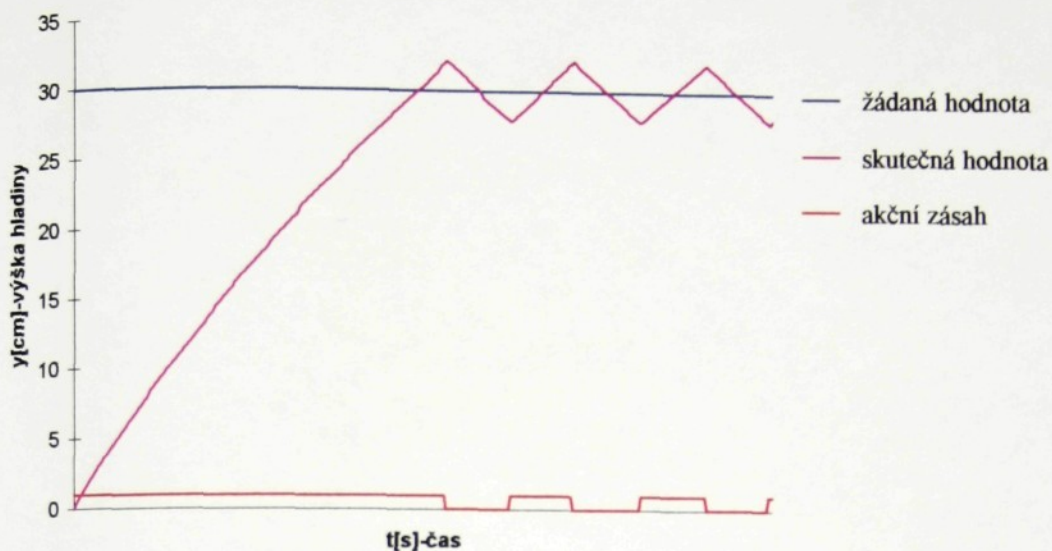


*To jsou čísla
pro tyto regulační
prochodu a se
charakteristika
regulátoru*

Obr. 7. 5. . Naměřená charakteristika dvoupolohového regulátoru s nulovou hysterezi

Na obr. 7. 5. je vidět naměřená charakteristika dvoupolohového regulátoru s nulovou hysterezi.

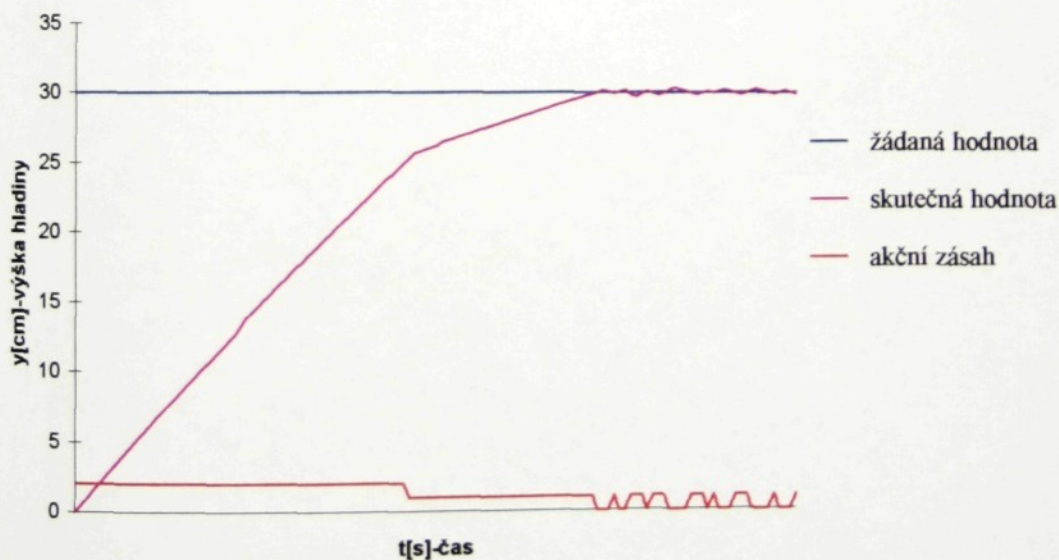
Na obr. 7. 6. je naměřená charakteristika dvoupolohového regulátoru s hysterezi. Velikost hystereze byla nastavena na 2cm.



Obr. 7. 6. . Naměřená charakteristika dvupolohového regulátoru hysterezí 2 cm

7. 3. Návrh algoritmů pro realizaci třípolohové regulace

Již z názvu je zřejmé, že se jedná o regulaci, ve které je možno nastavit regulátor do tří poloh.

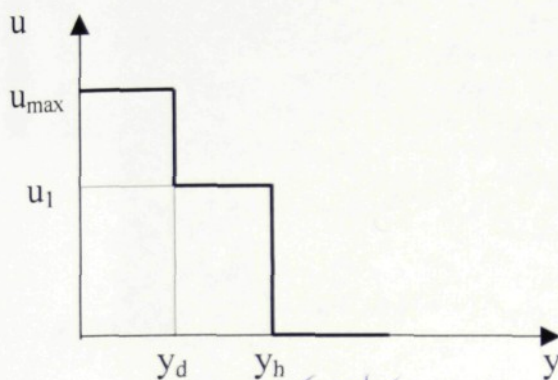


Obr. 7. 7. Naměřená charakteristika třípolohového regulátoru s nulovou hysterezí

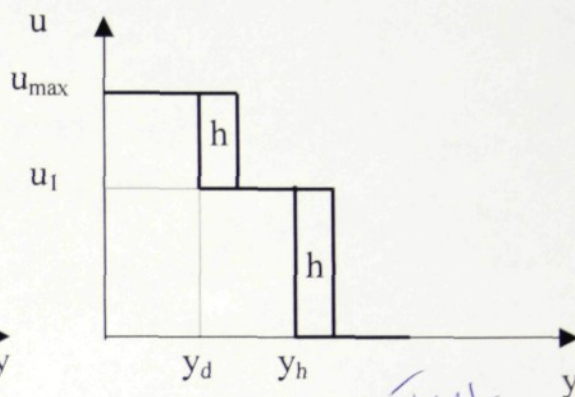
Právě tím, že můžeme regulátorem nastavit tři různě velké akční zásahy, dosahujeme, na rozdíl od dvupolohové regulace, kvalitnějšího regulačního pochodu. Třetí hodnota akční veličiny u_1 velmi ovlivňuje jakost pochodu, a proto také velmi záleží na jejím

nastavení. V našem případě, protože používáme dvoucestný ventil, realizujeme $u_{\max}$ jako bychom měli ještě jeden stejně velký přítok.

Charakteristiky třípolohového regulátoru s nulovou hysterezí a s hysterezí h jsou vyznačeny na obr. 7. 8. a 7. 9. Na obr. 7. 7. je naměřená charakteristika třípolohového regulátoru s nulovou hysterezí.



Obr. 7. 8.



Obr. 7. 9.

Realizace algoritmu třípolohového regulátoru je vyjádřena v programu 3. Realizace programu třípolohového regulátoru s hysterezí je vyjádřena v programu 4.

Program 3.

```
k2 = w1 - y1;
if( k2 > 10 ){
    u2 = 1;
    u1 = 1;
}
else if(w1 > y1){
    u2 = 0;
    u1 = 1;
}
else{
    u2 = 0;
    u1 = 0;
}
```

Program 4.

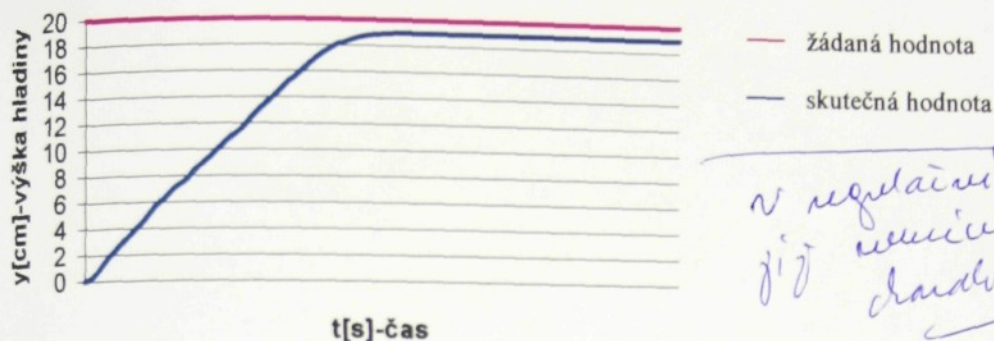
```
k2 = w1 - y1;
if( k2 > 10 ){
    u2 = 1;
    u1 = 1;
}
else if(w1 - h > y1){
    u2 = 0;
    u1 = 1;
}
else if(w1 + h < y1){
    u2 = 0;
    u1 = 0;
}
```

Kde $w1$žádaná hodnota $u1$akční zásah na hlavním ventilu hzvolená hystereze
 $y1$skutečná hodnota $u2$akční zásah na pomocném ventilu

7.4. Přechodové charakteristiky regulačního obvodu

a) P - regulátor

Proporcionální zesílení = 0.4



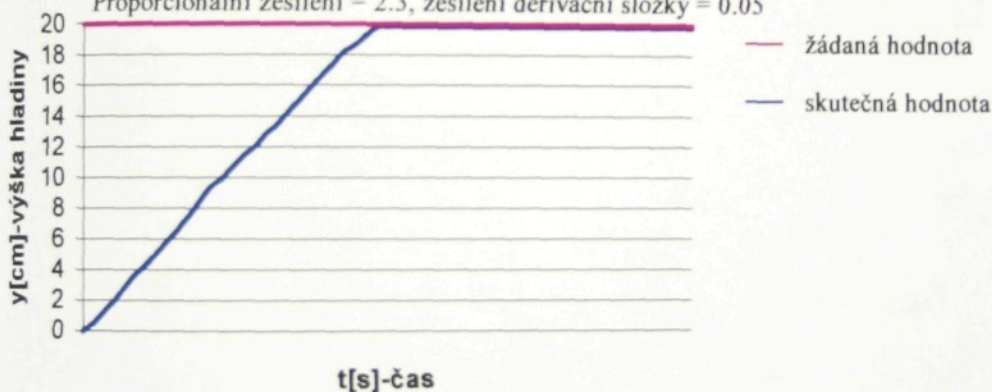
Přechodová charakteristika je odlišná na jednotlivé systémy

v regulačním obvodu již nemáme přelomový charakter

Obr. 7. 10 Naměřená charakteristika s P regulátorem

b) PD - regulátor

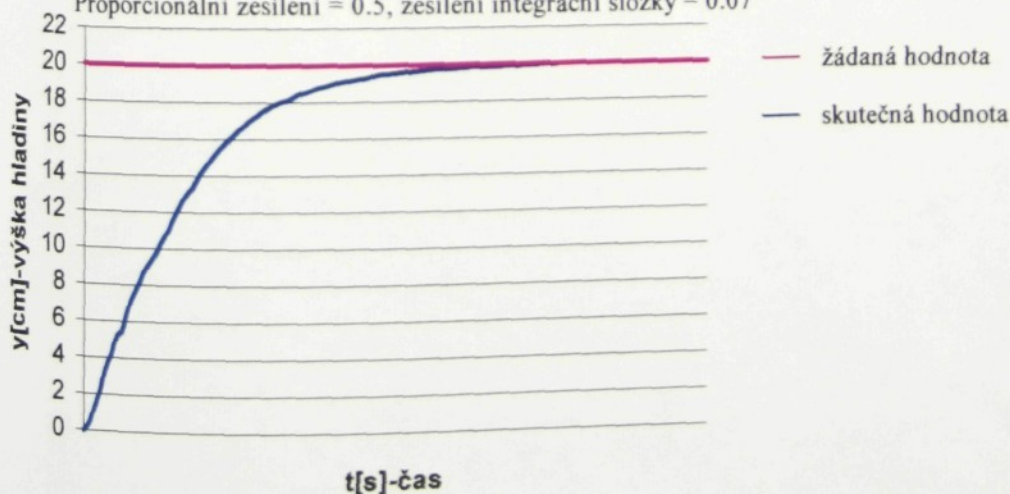
Proporcionální zesílení = 2.3, zesílení derivační složky = 0.05



Obr. 7. 11 Naměřená charakteristika s PD regulátorem

c) PI - regulátor

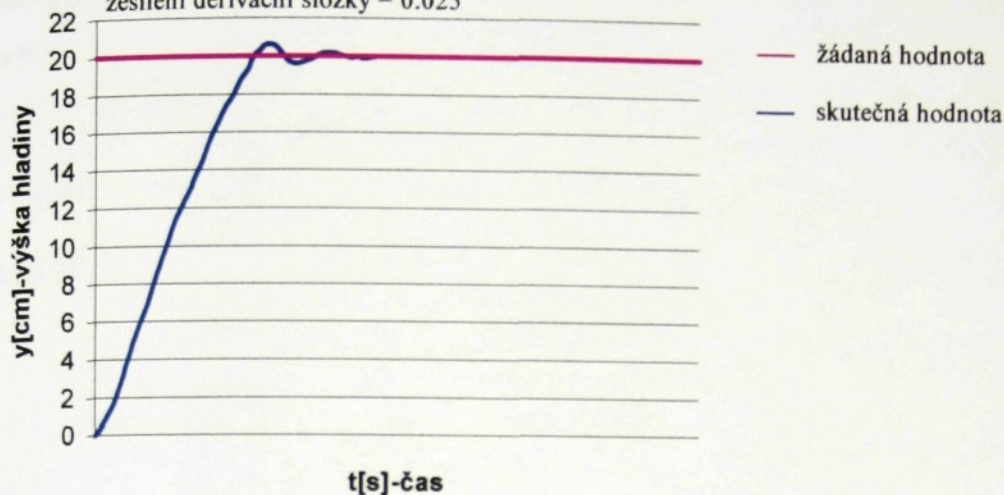
Proporcionální zesílení = 0.5, zesílení integrační složky = 0.07



Obr. 7. 12 Naměřená charakteristika s PI regulátorem

d) PID - regulátor

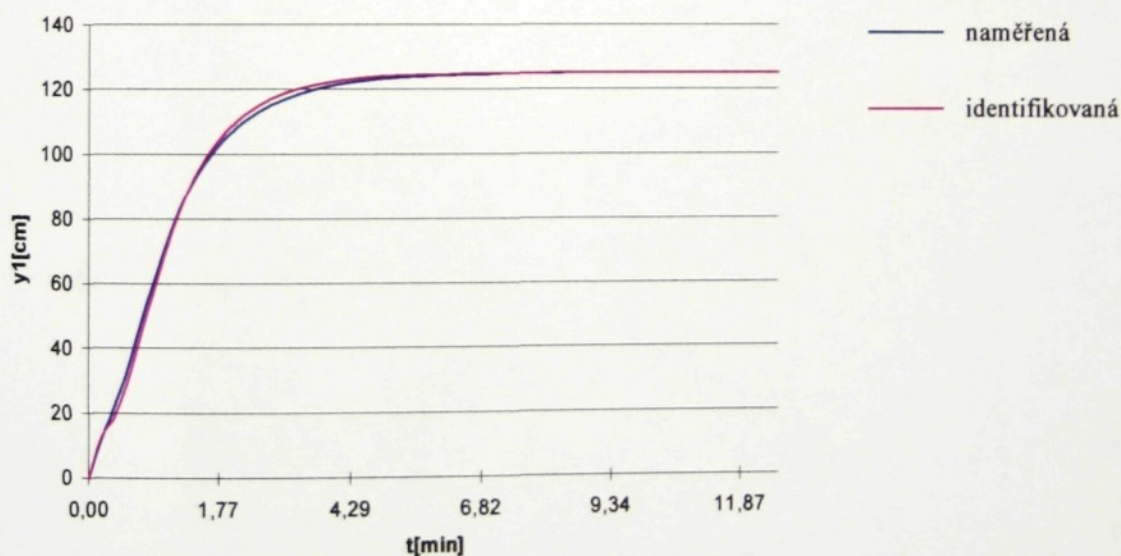
Proporcionální zesílení = 0.252, zesílení integrační složky = 0.630
zesílení derivační složky = 0.025



Obr. 7. 13. Naměřená charakteristika s PID regulátorem

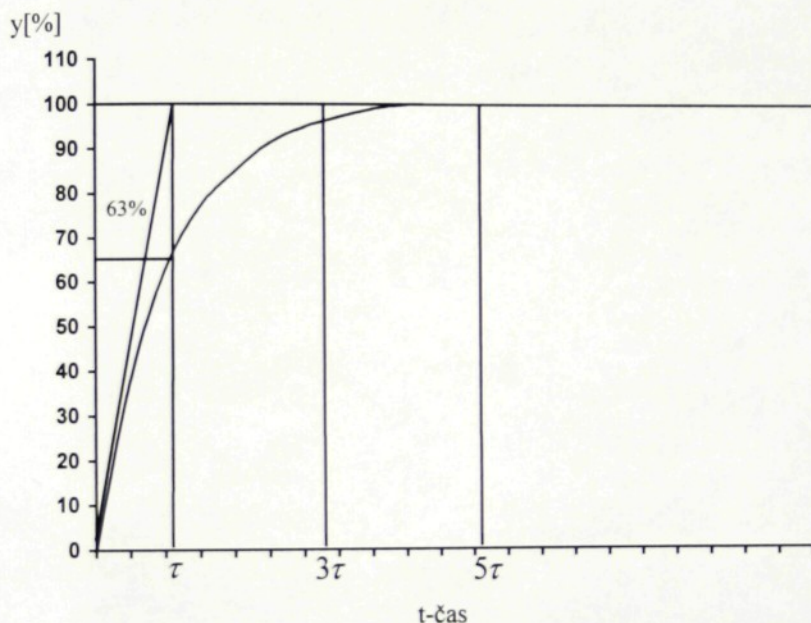
7. 5. Měření přechodové charakteristiky

Měření přechodové charakteristiky je provedeno automaticky, tzn. že po stisknutí příslušného tlačítka se měření spustí a po ustálení hodnoty se měření automaticky ukončí.



Obr. 7. 14 Porovnání naměřené a identifikované přechodové charakteristiky

Identifikace byla provedena metodou, která je ukázána na obr. 7. 15. Hodnota časové konstanty τ je měřena v místě, kde nám přechodovou charakteristiku protíná 63% hodnota regulované veličiny z ustáleného stavu. V hodnotě 3τ se počítá s tím, že veličina dosáhla 95% a v 5τ 99, 9% ustáleného stavu. Pomocí této aproximace dostáváme popis $G_{S(p)} = \frac{125}{1,417p+1}$. Na obr. 7. 14. je porovnání naměřené a identifikované přechodové charakteristiky.



Obr. 7. 15. Metoda aproximace přechodové charakteristiky soustavy 1. řádu

*To není aproximace,
ale vyhodnocení
parametrů předpokládané
funkce (T, A)*

8. MODEL NÁDRŽE

V této kapitole je popsán návrh modelu nádrže s použitím simulačního programu Matlab/Simulink, který simuluje reálný výtok z nádrže.

8. 1. Sestavení modelu

Musíme zjistit, jak se bude měnit v čase výška hladiny h a vytékající množství Q_0 . Z hmotové bilance systému $\rho Q = \rho Q_0 + \frac{d(\rho S h)}{dt}$, Torricelliho vztahu pro výtokovou rychlost (s korekcí na skutečnou rychlost pomocí hydrodynamického součinitele) $v_0 = \alpha \sqrt{2gh}$ a souvislosti mezi hmotovým průtokem a rychlostí $\rho Q_0 = \rho v_0 S_0$ sestavíme výslednou nelineární diferenciální rovnici prvního řádu pro závislost výšky hladiny h na přitékajícím množstvím Q .

$$Q = \alpha S_0 \sqrt{2gh} + S \frac{dh}{dt}$$

Pro potřebu převodu do schématu modelu tuto rovnici upravíme na ekvivalentní tvar implicitní integrální rovnice:

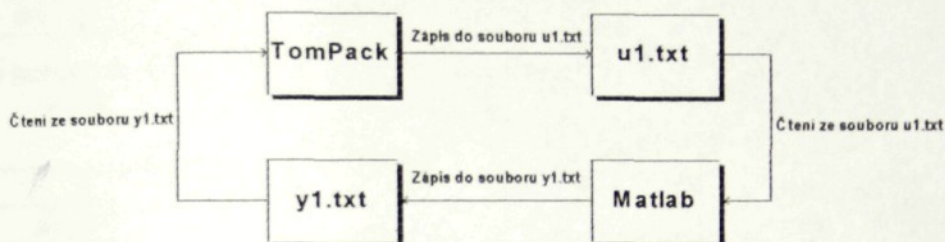
$$h = \int \frac{1}{S} \left(Q - \alpha S_0 \sqrt{2g\sqrt{h}} \right) dt$$

Kde hvýška hladiny
 S_0velikost výtokového otvoru
 Sprůřez nádoby
 gtíhové zrychlení
 Qpřitékající množství
 Q_0odtékající množství

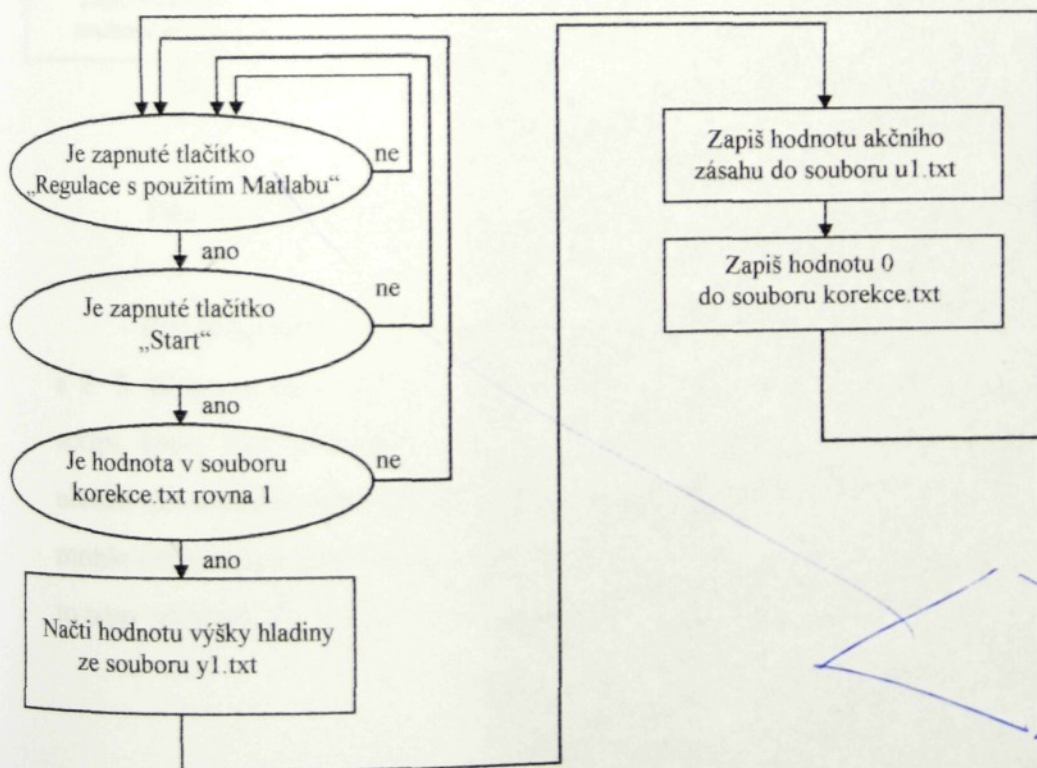
Tuto rovnici již můžeme přímo přepsat do schématu modelu SIMULINKu. Obr. 8. 4. ukazuje celou rovnici namodelovanou.

8. 2. Propojení TomPacku a Matlabu

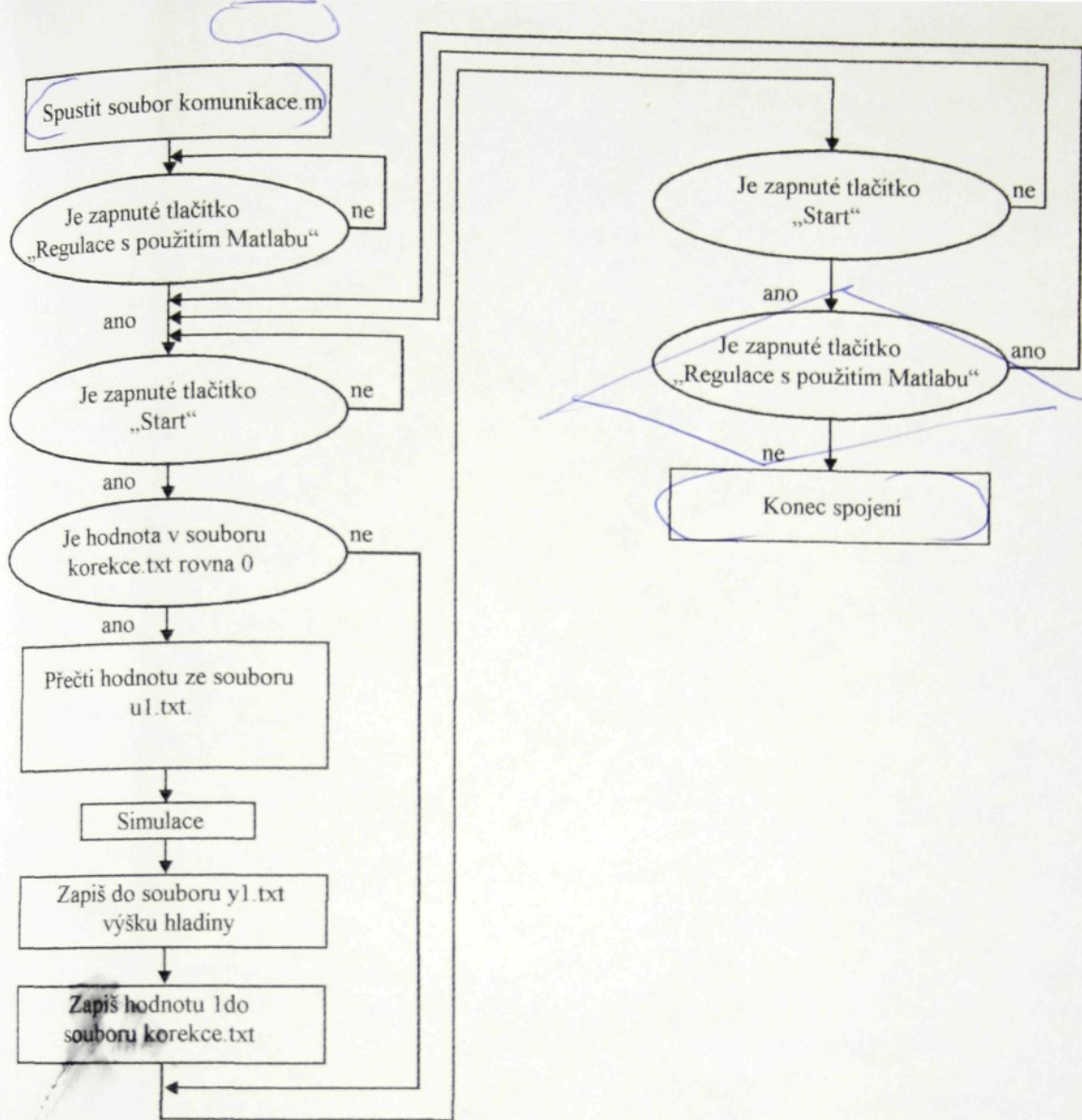
Principiální schéma na obr. 8. 1. nám ukazuje, jak spolu komunikují TomPack a Matlab. Jak je vidět, je celé spojení navrženo pomocí zápisů do souborů a čtení ze souborů. Aby oba programy věděly, kdy mohou číst hodnoty či zapisovat, byl vytvořen soubor **korekce.txt**, pomocí kterého se tak děje. Musí se dát také pozor na problematiku zápisu hodnoty do souboru. TomPack umí načíst pouze hodnotu, za kterou následuje binární hodnota konce řádku a odřádkování. To je v Matlabu řešeno tak, že se za ukládající proměnnou napíše „\r\n“. Bez tohoto zápisu TomPack nedokáže načíst potřebnou informaci.



Obr. 8. 1. Principiální schéma komunikace

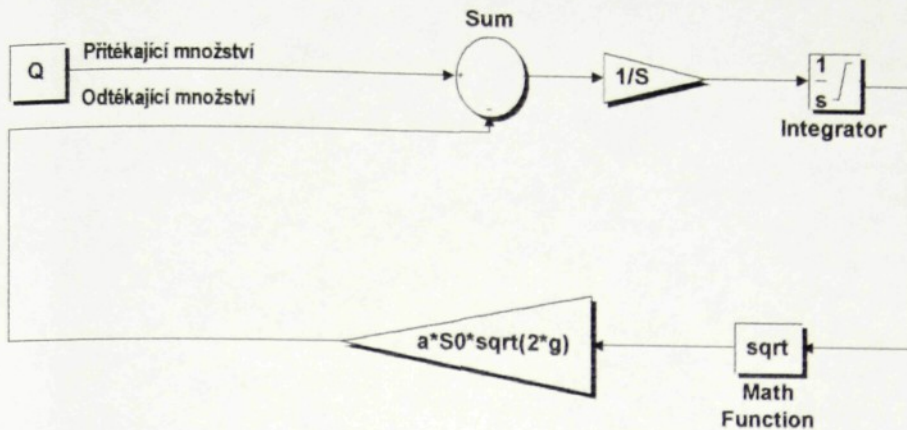


Obr. 8. 2. Principiální diagram pro komunikaci s Matlabem v TomPacku

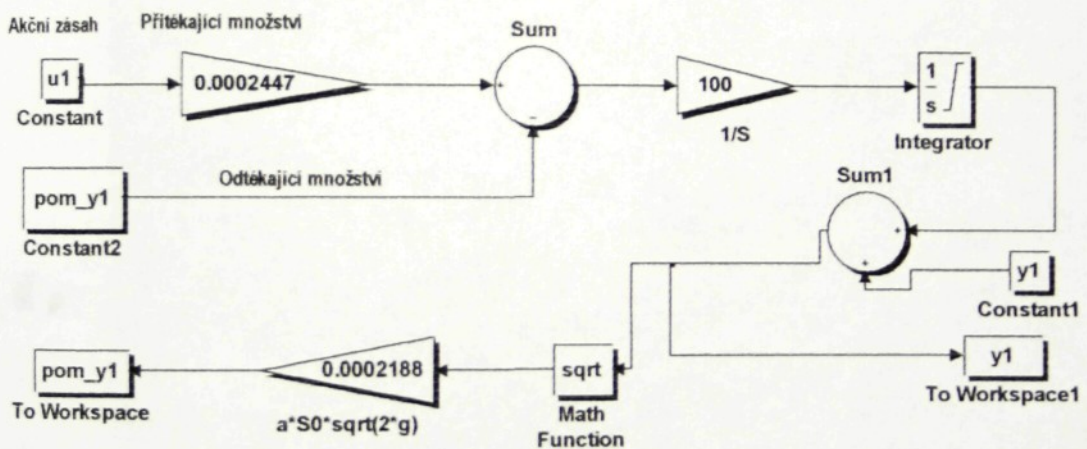


Obr. 8. 3. Principiální diagram pro komunikaci s TomPackem v Matlabu

Pro detailnější pohled celého spojení poslouží principiální diagramy na obr. 8. 2. a 8. 3. K tomu, aby TomPack mohl komunikovat s Matlabem, bylo třeba navrhnout script, který toto propojení umožní. Je napsán také v Matlabu, jako tzv M - file, a uložen do souboru **komunikace.m** . Jeho zdrojový text je v příloze 3. K tomu, aby mohla vizualizace správně probíhat, musí se simulační schéma z obr. 8. 12. upravit. Jak, to nám ukazuje obr 8. 13.



Obr. 8. 4. Blokové schéma nádrže

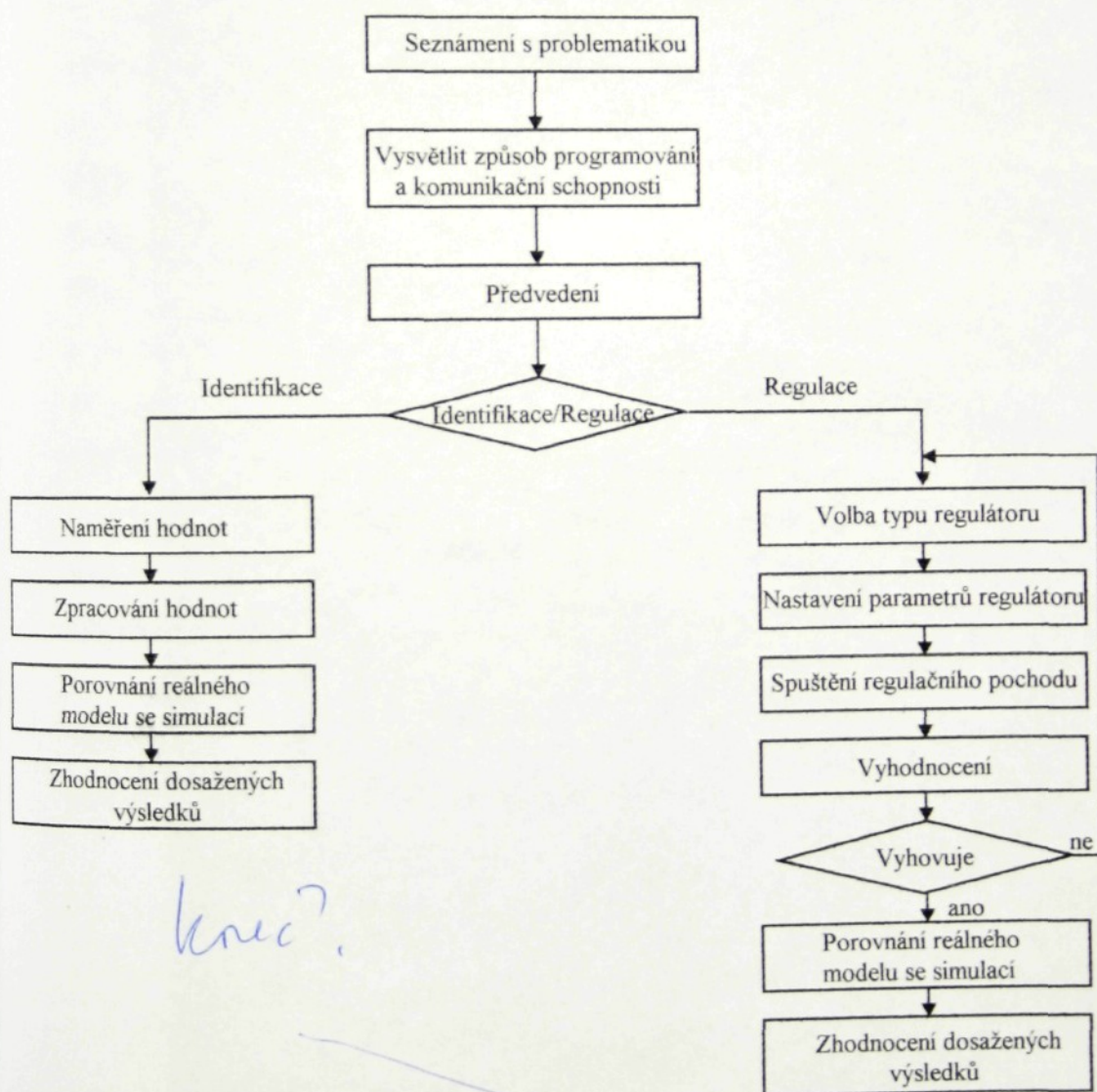


Obr. 8. 5. Blokové schéma nádrže pro komunikaci s TomPackem

9. 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). Můžeme i předvést malou ukázkou tvorby jednoduchého projektu.

V rámci konkrétního ovládání vyvinuté aplikace seznámíme studenty s možnostmi regulace nebo měření přechodové charakteristiky. Pro detailnější pohled poslouží následující vývojový diagram (obr. 9. 1.).



Obr. 9. 1. Vývojový diagram

10. ZÁVĚR

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

Přínos práce spočívá v tom, že byla prakticky ověřena spolupráce vizualizačního SW TomPack s modelem vytvořeným v prostředí Matlab/Simulink a tím rozšířena možnost srovnávat výsledky získané analýzou matematického modelu s chováním reálného regulačního obvodu. Konkrétní výsledky tohoto srovnání však bude možno publikovat až po realizaci modelu subsystému regulace výšky hladiny v nádrži s přítokem a odtokem kapaliny.

TomPack je velmi elegantní nástroj pro tvorbu vizualizačního softwaru. Výroba vlastního projektu je jednoduchá a pro zkušeného programátora i rychlá. Jednou z negativních věcí, kterou jsem si při vytváření SW zpozoroval je ta, 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. Hodnoty se však ukládají do souboru s periodou 0, 1s a při několika měřeních budou vždy stejné. Další problém, který vznikl při propojování TomPacku a Matlabu, je, že se musí dát pozor na problematiku zápisu hodnoty do souboru. Tento problém je popsán a vyřešen v kapitole 8. 2.

K již dříve popsaným problémům, nezajištění potřebných technických prostředků, bych navrhoval na tuto diplomovou práci navázat a zrealizovat fyzické propojení vizualizačního SW a úlohy.

Součástí této diplomové práce je CD s vytvořeným projektem, navrženým blokovým schématem v prostředí Matlab/Simulink a všechny potřebné soubory ke spojení TomPacku a Matlabu.

odmítl jsem

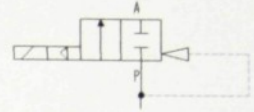
LITERATURA:

- [1] MARTINÁSKOVÁ, M., ŠMEJKAL, L.:
Řízení programovatelnými automaty, skriptum ČVUT Praha, 1998
- [2] HANUŠ, B.:
Teorie automatického řízení, VŠST Liberec, 1982, skriptum
- [3] OLEHLA, M.:
Základy aplikované kybernetiky, TU Liberec, 1997, skriptum
- [4] DUŠEK, F.:
Matlab a Simulink úvod do používání, Pardubice, 2000, skriptum
- [5] OLEHLA, M.:
Identifikace technologických soustav, TU Liberec, 1997, skriptum
- [6] NIXON, F.:
Principy soustav automatického řízení 37.
- [7] WWW.STASTO.CZ, on-line katalog firmy Stasto
- [8] WWW.WIKA.COM, on-line katalog firmy Wika
- [9] Podniková dokumentace firmy ProjectSoft HK, Hradec Králové, 2000

SEZNAM PŘÍLOH:

- Příloha 1. Elektromagnetický ventil STASTO 21 W
- Příloha 2. Kontinuální snímač hladiny WIKA C - 10
- Příloha 3. Zdrojový text souboru komunikace.m
- Příloha 4. Simatic S7 - CPU 314 IFM

2/2 – cestný magnetický ventil nepřímo řízený, R 3/8" - R 3/4" typ 21H



Technické údaje:

Konstrukce	2/2-cestný magnetický ventil s membránou, nepřímo řízený, bez proudu uzavřený
Závitové připojení	R 3/8" - R 3/4"
Teplota okolí	-20° C až + 50° C, při vyšší okolní teplotě nutno použít cívku DR
Teplota média	Závisí na materiálu těsnění
Viskozita	max. 21 mm ² /s (cSt) příp. 3° E
Materiál	Těleso mosaz, vnitřní díly mosaz a nerez ocel, materiál membrány NBR (Perbunan), EPDM nebo Viton
Způsob zástavby	Montáž do systému tuhého vedení
Montážní poloha	Libovolná
Rozsah dodávky	Včetně přístrojové zástrčky Pg 9 dle DIN 43650 – A / ISO 4400
Elektrické údaje:	
Druh napětí	Střídavé i stejnosměrné
Standardní napětí	230 V / 50 Hz, 24 V / 50 Hz, 24 V =
Zvláštní napětí	6 až 400 V / 50 Hz nebo 60 Hz, 6 až 220 V =
Připustné kolísání napětí	± 10 %
Příkon	Viz tabulku „Příkon magnetických cívek“
Doba zapnutí	100 % (trvalý provoz)
Druh ochrany	IP 65 dle DIN 40050 při montáži odpovídající přístrojové zástrčky (ochrana před prachem a vodním proudem)

Pneumatické a hydraulické údaje:

Protékající médium	Všechna kapalná a plynná média, která nenapadají použité materiály
Spinací doba	Závislá na pracovním tlaku a protékajícím médiu
Četnost spínání	max. 200 / min
Jiné provedení	Chemicky poniklováno
Výrobce:	O.D.E.

Objednací klíč :

	21H 7 K B 120 D - 230/50	
	Připojení	Druh proudu
	Materiál těsnění	Napětí
	Světlost (10 ⁻¹ mm)	Magnetická cívka
Připojení	7 = R 3/8", 8 = R 1/2", 9 = R 3/4"	
Materiál těsnění	B = Perbunan (NBR), V = Viton (FKM), E = EPDM	
Magnetická cívka	D = standardní cívka DA, DR = cívka pro vlhké prostředí	

Možnosti použití jednotlivých materiálů těsnění :

B (NBR, Perbunan)	-10 až +90°C	Neutrální plyny a kapaliny
E (EPDM)	-10 až +130°C	Horká voda, pára, kyseliny
V (Viton, FKM)	-10 až +130°C	Oleje, benzin

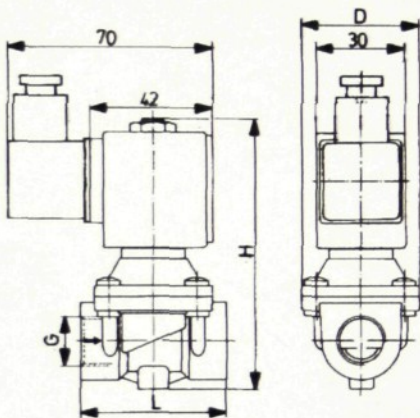
Přikony magnetických cívek :

	Cívka DA	Cívka DR
Přitažný příkon ve VA (střídavý proud)	28	28
Přidržovací příkon ve VA (střídavý proud)	13	13
Přidržovací příkon ve W (střídavý proud)	8	8
Příkon ve W (ss proud) při provozní teplotě	8	8

Magnetické cívky :

Typ	Druh ochrany	Použití
DA	IP65	Teplota média do max. 120°C
DR	IP65	Teplota média do max. 160°C, vysoká vlhkost vzduchu

Rozměry :



Typ	Cívka	Světlost DN[mm]	Připojení G	Tlakový rozsah [barů]		D	H	L	Hodnota KV [m³/h vody]	Hmotnost [kg]
				AC	DC					
21H7KB120	D	12	G 3/8"	0,1 - 20	0,1 - 20	40	94	50	2,2	0,42
21H8KB120	D	12	G 1/2"	0,1 - 20	0,1 - 20	40	94	50	2,2	0,42
21H9KB180	D	18	G 3/4"	0,1 - 12	0,1 - 6	52	96	65	3,3	0,68

Ventily jsou použitelné také pro podtlak.

Upozornění: Při objednávce uveďte napětí a druh proudu. Doporučujeme vždy předřadit filtr, aby při případném znečištění dopravovaného média nedošlo vlivem nečistot v médiu k funkčním poruchám.

STASTO Automation s.r.o.
K Náklí 512
CZ-257 41 Týnec nad Sázavou

Tel 0301/701 700, 701 834
Fax 0301/701 701
E-Mail stasto@czn.cz
www.stasto.cz



OEM Pressure Transmitters
Type C-10 - 100 INWC to 15,000 PSI

Tronic

(Previous model numbers 891.14.540, 891.24.540)

- Durable and cost effective for OEM applications
- 4-20 mA 2-wire output signal, voltage signals available
- Highly resistant to pressure spikes and vibration
- Stainless steel case and wetted parts
- Excellent vibration resistance
- Custom wiring and process connections available



WIKA C -10 pressure transmitters and transducers are precision engineered to meet the price and performance requirements of Original Equipment Manufacturers. Each unit undergoes extensive quality control testing and calibration to achieve an accuracy $\leq 0.50\%$ full scale. The printed circuit boards use state-of-the-art surface mount technology and are potted in silicone gel for protection against mechanical shock, vibration, and moisture. Each unit is temperature compensated to assure accuracy and long term stability when exposed to severe ambient temperature variations.

The C -10 OEM pressure transmitter provides a rugged design featuring excellent vibration resistance and long service life. OEM applications include hydraulics, pneumatics, press control, compressor control, pump protection and numerous other processing and control operations.



STANDARD RANGES

RANGE	MAXIMUM*	BURST**	RANGE	MAXIMUM*	BURST**
0-100 INWC	30 PSI	30 PSI	0-300 PSI	1100 PSI	1100 PSI
0-5 PSI	30 PSI	30 PSI	0-400 PSI	1100 PSI	3600 PSI
0-10 PSI	60 PSI	60 PSI	0-500 PSI	1100 PSI	5800 PSI
0-15 PSI(A)	70 PSI	70 PSI	0-600 PSI	1100 PSI	5800 PSI
0-25 PSI(A)	145 PSI	145 PSI	0-1000 PSI	1750 PSI	8000 PSI
0-30 PSI	145 PSI	145 PSI	0-1500 PSI	2900 PSI	11,600 PSI
0-50 PSI(A)	250 PSI	250 PSI	0-2000 PSI	4600 PSI	14,500 PSI
0-60 PSI	250 PSI	250 PSI	0-3000 PSI	4600 PSI	14,500 PSI
0-100 PSI(A)	500 PSI	500 PSI	0-5000 PSI	11,600 PSI	25,000 PSI
0-150 PSI	500 PSI	500 PSI	0-7500 PSI	17,400 PSI	35,000 PSI
0-200 PSI	500 PSI	500 PSI	0-10,000 PSI	17,400 PSI	35,000 PSI
0-250 PSI(A)	1100 PSI	1100 PSI	0-15,000 PSI	21,750 PSI	43,500 PSI

Notes:

* Maximum pressure, causing no permanent changes in specifications but may lead to zero and span shifts.

** Burst pressure, leading to destruction of the transmitter and possible loss of media.

(A) Standard ranges available with absolute pressure reference.

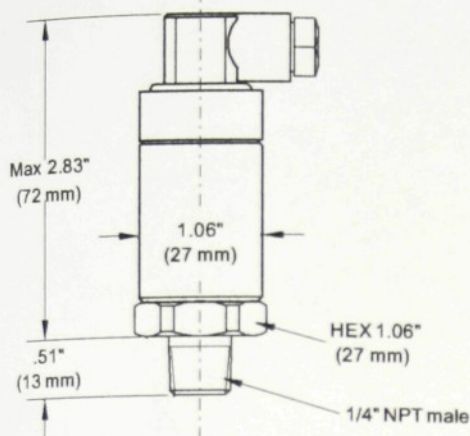
APE 81.06

Specifications	Units	Type C-10
Sensing principle Pressure ranges Pressure reference	PSI	piezoresistive up to 250 PSI, thin film ≥ 300 PSI standard ranges as listed {custom ranges available} relative pressure {absolute reference to 250 PSIA}
Pressure connection Material: -wetted parts -case -internal transmitting liquid		1/4" NPT male; (G1/4B) {SAE #4 (7/16-20 UNF) male O-ring boss for ranges ≥ 400 PSI} {other pressure connections available} 1.4571 and 1.4542 stainless steel (316 ss and PH17-4 ss) {for other materials see WKA chemical seals} 1.4301 stainless steel (304 ss) silicone oil for piezoresistive sensors to 300 PSI, {halocarbon oil for oxygen service}, no liquid fill used for thin film sensors ≥ 300 PSI
Supply voltage U_b Output and load limitations: Output signal and maximum load	DC Volts	10 - 30 (14 - 30 for 0 - 10 V output signal) 4-20 mA 2-wire system $R_A[\text{Ohm}] \leq (U_b[\text{V}] - 10\text{V}) / 0.02 \text{ A}$ {0-20 mA 3-wire system} $R_A[\text{Ohm}] \leq (U_b[\text{V}] - 10\text{V}) / 0.02 \text{ A}$ {0-5 V 3-wire system} $R_A > 5 \text{ kOhm (min)}$ {0-10 V 3-wire system} $R_A > 10 \text{ kOhm (min)}$ {other signal outputs available}
Upper cutoff frequency Response time (10...90%)	Hz milliseconds	150 ≤ 1
Accuracy (linearity, including hysteresis and repeatability) Repeatability Hysteresis 1 year stability	% of span % of span % of span	$\leq 0.50\%$ (B.F.S.L.) ≤ 0.05 ≤ 0.1 ≤ 0.2 (under reference conditions)
Temperature Media Ambient Storage Compensated range Temperature error (reference 70°F) on zero point on span	 % of span	-22°F to +212°F (-30°C to +100°C) { -40°F to +257°F (-40°C to +125°C)} -4°F to +176°F (-20°C to +80°C) -40°F to +212°F (-40°C to +100°C) +32°F to +176°F (0°C to +80°C) ≤ 0.3 per 18°F (10°C) change ≤ 0.2 per 18°F (10°C) change
CE conformity		Interference emission per EN 50 081-1 (March 1993) and EN50 081-2 (March 94), Interference immunity per EN 50 082-2 (March 1995)
Shock resistance Vibration resistance	g g	1000 per IEC 770 for mechanical shock 50 per IEC 770 for vibration under resonance conditions
Electrical connection Weight Dimensions Electrical protection Environmental protection	 lb	4-pin miniature L-plug per DIN 43 650 {5 foot vented flying lead, 4 or 6 pin MIL plug} {custom plug and cable assemblies} approximately 0.2 (0.1 Kg) see drawing protected against reverse polarity, short circuit, and overvoltage IP 65 (NEMA 5) with 4 pin L-plug, MIL plugs {IP 67 (NEMA 4) with 5 foot flying lead}

Notes: Items in curved brackets { } are available as special order options

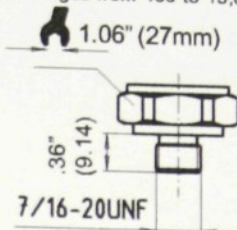
Dimensions

DIN 43 650 miniature L-plug

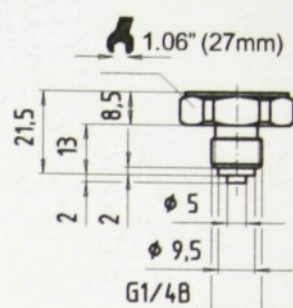


Optional process connections

SAE #4 male O-ring boss (for ranges from 400 to 15,000 PSI)

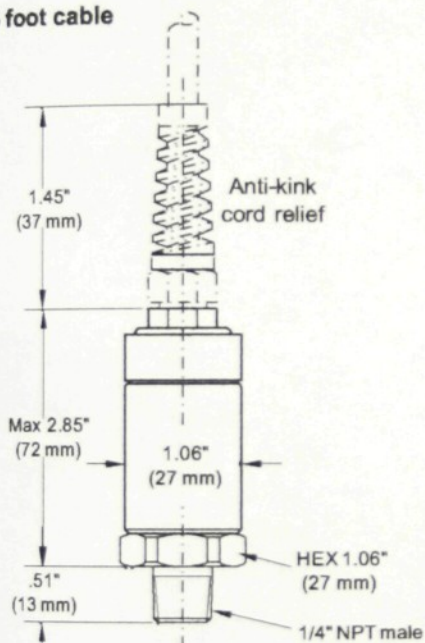


G1/4B BSP

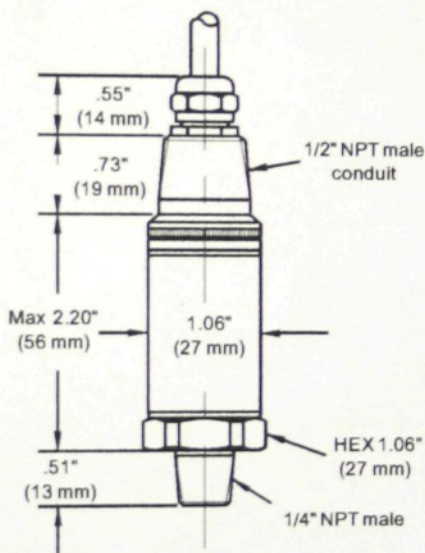


(metric connections
shown in mm)

5 foot cable



1/2" NPT male conduit with 5 foot cable



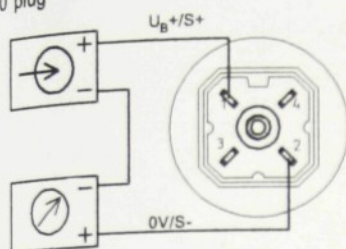
Electrical connections

	Standard			
Type	DIN 43 650 Mini -L plug	Din 43 650 with 6' cable	5' vented cable	1/2" NPT male conduit
Protection	NEMA 5 IP 65	NEMA 5 IP 65	NEMA 4 IP 67	NEMA 4 IP 67
Description and part numbers	Standard part # 9735882	Plug with heavy duty 6' molded cable part # 9740010	Vented cable with free ends. Longer lengths available. Custom order.	1/2" NPT male conduit and 5' cable with free ends

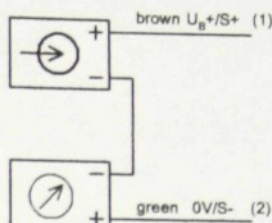
Wiring

4-wire system

W 43 650 plug

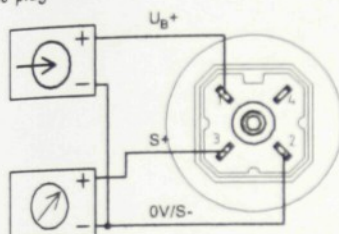


flying lead

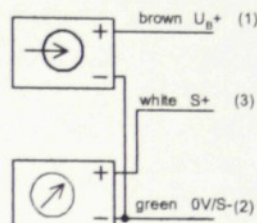


4-wire system

W 43 650 plug



flying lead



4-wire system

3-wire system

Wire	Coding	DIN Plug	Wire Color
Supply +	$U_B + / S +$	pin 1	brown
Signal -	$0V / S -$	pin 2	green

Wire	Coding	DIN Plug	Wire Color
Supply +	$U_B +$	pin 1	brown
Supply - Signal -	$0V / S -$	pin 2	green
Signal +	$S +$	pin 3	white

THE MEASURE OF Total Performance™

Ordering Information:

State computer part number (if available) / type number / size
Range / connection size and location / options required.

Specifications given in this price list represent the state of engineering at the time of printing.
Specifications may take place and the specified materials may change without prior notice.

000

WIKAL

WIKAL Instrument Corporation

1000 Wiegand Boulevard

Lawrenceville, Georgia 30043-5868

Tel: 770-513-8200 Fax: 770-338-5118

<http://www.wika.com> e-mail: info@wika.com

příloha 3: Zdrojový text souboru komunikace.m

```
clear all; close all; echo off; format long;
global konec pom2_y1 ;
konec_sim = 0;
spus_pr = 0;
model='nadrz';
clc
y1 = 0;
pom_y1 = 0;
i = 0;

while( spus_pr ~= 1 )

    soubor_spusteni = fopen( 'poc_spus.txt','r' );

    while( soubor_spusteni == -1 )
        pause( 0.001 );
        soubor_spusteni = fopen( 'poc_spus.txt','r' );
    end

    spus_pr = fscanf( soubor_spusteni, '%d' );
    fclose( soubor_spusteni );

end

konec_pr = spus_pr;

while( konec_pr ~= 0 ) %testování zda běží vizualizace pomocí matlabu

    while( konec_sim ~= 1 )

        soubor_spusteni = fopen( 'spusteni.txt','r' );

        while( soubor_spusteni == -1 )
            pause( 0.001 );
            soubor_spusteni = fopen( 'spusteni.txt','r' );
        end

        konec_sim = fscanf( soubor_spusteni, '%d' );
        fclose( soubor_spusteni );
        i = i + 1;

        if( i == 2 )
            konec_sim = 1;
        end

    end

    while ( konec_sim ~= 0 )
        pause(0.05);
        %zjištění hodnoty pro simulaci a zápis
        soubor_korekce = fopen( 'korekce.txt','r' );

        while( soubor_korekce == -1 )
            pause( 0.001 );
            soubor_korekce = fopen( 'korekce.txt','r' );
        end

        korekce = fscanf( soubor_korekce, '%d' );
        fclose( soubor_korekce );

        if( korekce == 0 )
            soubor_ul = fopen( 'ul.txt','r' ); %načtení akčního zásahu
```



```

while( soubor_ul == -1 )
    pause( 0.001 );
    soubor_ul = fopen( 'ul.txt','r' );
end

ul = fscanf( soubor_ul,'%f' );
fclose( soubor_ul );

y1 = y1 / 100;           %převod na m

sim(model);             %simulace nádrže

y1 = y1 * 100;          %převod na cm

%-----začátek zápisu y1 do souboru y1.txt-----
soubor_y1 = fopen( 'y1.txt','w' );

while( soubor_y1 == -1 )
    pause( 0.001 );
    soubor_y1 = fopen( 'y1.txt','w' );
end

fprintf( soubor_y1,'%f\r\n', y1 );
fclose( soubor_y1 );
%-----Konec-----

soubor_korekce = fopen( 'korekce.txt','w' );

while( soubor_korekce == -1 )
    pause(0.001);
    soubor_korekce = fopen( 'korekce.txt','w' );
end
    fprintf( soubor_korekce,'%d\r\n', 1 );
fclose( soubor_korekce );

end

soubor_spusteni = fopen( 'spusteni.txt','r' );

while( soubor_spusteni == -1 )
    pause( 0.001 );
    soubor_spusteni = fopen( 'spusteni.txt','r' );
end

konec_sim = fscanf( soubor_spusteni, '%d' );
fclose( soubor_spusteni );

end

soubor_y1 = fopen( 'y1.txt','w' );

while( soubor_y1 == -1 )
    pause( 0.001 );
    soubor_y1 = fopen( 'y1.txt','w' );
end

fprintf( soubor_y1,'%f\r\n', 0 );
fclose( soubor_y1 );

y1 = 0;
i = 0;

soubor_korekce = fopen( 'korekce.txt','w' );

```

```

while( soubor_korekce == -1 )
    pause( 0.001 );
    soubor_korekce = fopen( 'korekce.txt','w' );
end

fprintf( soubor_korekce,'%d\r\n', 1 );
fclose( soubor_korekce );

soubor_spusteni = fopen( 'poc_spus.txt','r' );

while( soubor_spusteni == -1 )
    pause( 0.001 );
    soubor_spusteni = fopen( 'poc_spus.txt','r' );
end

konec_pr = fscanf( soubor_spusteni, '%d' );
fclose( soubor_spusteni );
end
disp('-----Byla ukončeno spojení s TomPackem. Pro navázání
nového musíte spustit znova script.-----')

```


*kann je so
lesen?*

Technische Daten

Zentralbaugruppe	CPU 314 IFM
Arbeitsspeicher (1 Anweisung entspricht durchschnittlich 3 Byte)	32 KByte / 10 K Anweisungen RAM (integriert)
Ladespeicher	
• integriert	48 KByte RAM 48 KByte FEPR0M (nur 6ES7 314-5AE00-0AB0)
• steckbar als MC, max.	4 MByte (nur 6ES7 314-5AE10-0AB0)
Pufferung	
• ohne Batterie	144 Byte; Merker, Zähler, Zeiten und Daten
• mit Batterie	alle Bausteine
Echtzeituhr	ja
Programmiersprache	6ES7 314-5AE03-0AB0: STEP® 7 V5.0 6ES7 314-5AE10-0AB0: STEP® 7 V5.0 SP3 6ES7 314-5AE83-0AB0: STEP® 7 V5.0 SP1
Programmorganisation	linear, strukturiert
Bausteinarten	• Organisationsbausteine (OB))
	• Funktionsbausteine (FB)
	• Funktionen (FC)
	• Datenbausteine (DB)
	• Systemfunktionen (SFB, SFC)
Bausteinanzahl/-größe	
• OB	Siehe Operationsliste / max. 8 KRV16

*je so
physikalisch*

• FB	128 / max. 8 KByte
• FC	128 / max. 8 KByte
• DB	127 / max. 8 KByte
Programmbearbeitung	• freier Zyklus (OB1)
	• zeitgesteuert (OB35)
	• uhrzeitgesteuert (OB10)
	• alarmgesteuert (OB40)
	• Anlauf (OB100)
Schachtelungstiefe der Bausteine	8 für jede Programmbearbeitungsebene
Klammerebenen	8
Operationsvorrat	Binäre Verknüpfungen, Klammerbefehle, Ergebniszuzuweisung, Speichern, Zählen, Laden, Transferieren, Vergleichen, Schieben, Rotieren, Komplement bilden, Bausteine aufrufen, Festpunktarithmetik, Gleitpunktarithmetik, Sprungfunktionen
Anwenderprogrammschutz	Passwortschutz
Systemfunktionen (SFC)	Alarm- und Fehlerbearbeitung, Daten kopieren, Uhrenfunktionen, Diagnosefunktionen, Baugruppenparametrierung, Betriebszustandübergänge
Bearbeitungszeiten	
• Bitoperationen	0,3 bis 0,6 μ s
• Wortoperationen, etwa	1 μ s
• Zeit-/Zähloperationen	12 μ s
• Festpunktadditionen	2 μ s
• Gleitpunktadditionen	50 μ s
Zykluszeitüberwachung	150 ms (voreingestellt), einstellbar 1 bis 6000 ms

Merker	2048
• davon remanent mit Batterie	0 bis 2048 (M 0.0 bis M 255.7, einstellbar)
• davon remanent ohne Batterie	0 bis 1152 (M 0.0 bis M 143.7, einstellbar)
Zähler	64
• davon remanent mit Batterie	0 bis 63, einstellbar
• davon remanent ohne Batterie	0 bis 63, einstellbar
• Zählbereich	1 bis 999
Zeiten	128
• davon remanent mit Batterie	0 bis 71, einstellbar
• davon remanent ohne Batterie	0 bis 71, einstellbar
• Zeitbereich	10 ms bis 9990 s
Integrierte Funktionen	
• Zähler	1 Zähler mit 4 Eingängen oder 2 Zähler mit 2 Eingängen und 2 richtungsabhängige Vergleicher für jeden Zähler; Zählfrequenz 10 kHz; 32 Bit (incl. Vorzeichen)
• Frequenzmessung bis 10 kHz	1 Kanal bis max. 10 kHz; Messzeiten 0,1 s, 1 s, 10 s; Messverfahren: Berechnung von Impulszahl je Messzeit
• Gesteuertes Positionieren	1 Kanal; Wegerfassung über einen 24 V-asymmetrischen Inkrementalgeber; 3 digitale Eingänge werden vom Geber belegt (Spur A, Spur B, Referenzpunkt); Einfachauswertung der Zählimpulse (10 kHz)
• Integrierte Funktionsbausteine "Regeln"	PID-Regelungsbausteine
	• kontinuierliche Stellgrößenänderung

	• binäre Stellgrößenausgänge
	• Automatik/Handbetrieb
	• Stellwertbegrenzung
MPI-Schnittstelle	
• Teilnehmerzahl, max.	32 Teilnehmer am MPI-Bus; PG/PC, OP, weitere S7-300 [®] /400 [®] , C7; je CPU max. 4 statische/ 8 dynamische Verbindungen
	• PG/OP-Kommunikation
	• Globale Datenkommunikation
	• S7-Basis-Kommunikation
	• S7-Kommunikation
• Übertragungsgeschwindigkeit	187,5 kbit/s
• Entfernung zwischen 2 benachbarten Teilnehmern max.,	ohne Repeater: 50 m; mit 2 Repeatern: 1100 m; mit 10 Repeatern in Reihe: 9100 m; über Lichtwellenleiter: 23,8 km (mit 16 Sternkopplern oder OLM)
PG/PCs mit STEP [®] 7	über MPI-Schnittstelle anschließbar
Integrierte Ein-/Ausgänge	
• Digitaleingänge	20; DC 24 V davon 4 Kanäle für Prozessalarme oder integrierte Funktionen verwendbar
• Digitalausgänge	16; DC 24 V, 0,5 A
• Analogeingänge / Auflösung (Bit)	4; ± 10 V, ± 20 mA / 11 + Vz
• Analogausgänge / Auflösung (Bit)	1; ± 10 V, ± 20 mA / 11 + Vz
Echtzeituhr	ja
Gesamtadressraum E/A	512/512 Byte
Prozessabbild F/A	128/128 Rvte

Summe digitale Kanäle	max. 992
Summe analoge Kanäle	max. 248 E bzw. 124 A
Anzahl Baugr. je System	31
Anzahl ZG/EG	1/3
Anzahl DP-Stränge je CPU (integrierte Schnittst./CP 342-5)	-/1
DP-Master-Stationen je CPU (integrierte Schnittst./CP 342-5)	-/16
Adressraum je Station	122 Byte
Bgrp. je ET 200M	8
DP-Anschluss (Master/Slave)	1 (CP 342-5)
Kommunikationsfunktionen über CPs	
• PG/OP-Kommunikation	ja
• Erweiterte Kommunikation	ja (Server)
• S5-kompatible Kommunikation	ja (über nachladbare Bausteine)
• Standard-Kommunikation	ja (über nachladbare Bausteine)
Anz. Verbindungen statisch/dynamisch	4/8
Versorgungsspannung	
• Nennwert	DC 24 V
• zulässiger Bereich	20,4 bis 28,8 V
Stromaufnahme, typ.	1 A
Einschaltstrom, typ.	8 A
Verlustleistung, typ.	16 W
Maße (B x H x T) in mm	160 x 125 x 130
Gewicht	
• CPU	900 g
• Speicherkarte	-
Betreibbare Baugruppen (Empfehlung)	
• FM	4

• CP, Punkt zu Punkt	2
• CP, LAN, (K-Bus)	1
Betreibbare Software	
• Software-Regler	abhängig von benötigtem Speicherplatz und sich ergebender Laufzeit
• Prozessdiagnose	ja
• S7-GRAPH	-
• S7-HiGraph®	-
• S7-SCL	ja
• CFC	-
Integrierte Digitaleingänge	20
Eingangsspannung	
• Nennwert	DC 24 V
• bei Signal "1"	15 bis 30 V
• bei Signal "0"	-3 bis +5 V
Potentialtrennung	
• Sondereingänge / in Gruppen zu	nein / 4
• Eingänge / in Gruppen zu	ja / 16
Eingangsstrom	
• bei Signal "1", min. / typ.	2 mA / 7 mA
Eingangsverzögerung (bei Nennwert der Eingangsspannung)	
• für Standardeingänge, typ./max.	3 ms/5 ms
• für Interrupteingänge, max.	50 µs
• für Zähler, max.	50 µs
Anschluss von 2-Draht-BERO	
• zulässiger Ruhestrom	1,5 mA
Leitungslängen	
• ungeschirmt	600 m

• geschirmt	1000 m (100 m für Alarm- und Zählengänge)
Integrierte Digitalausgänge	16
Lastnennspannung L+/L1	DC 24 V
• zulässiger Bereich	20,4 bis 28,8 V
Ausgangsspannung	
• bei Signal "1", max.	L+ - 0,8 V
Potentialtrennung / in Gruppen zu	ja / 8
Ausgangsstrom maximal	
• bei Signal "1"	
- Nennwert bei 40 °C	0,5 A
- Nennwert bei 60 °C	0,5 A
- Mindeststrom	5 mA
• bei Signal "0", max.	0,5 mA
Gesamtbelastbarkeit	
• bei 40 °C	50 %
• bei 60 °C	25 %
Schaltfrequenz der Ausgänge	
• bei ohmscher Last	100 Hz
• bei induktiver Last	0,5 Hz
Begrenzung der induktiven Abschaltspannung	30 V
Kurzschluss-Schutz	elektronisch taktend
Leitungslängen	
• ungeschirmt	600 m
• geschirmt	1000 m
Integrierte Analogeingänge	4
Eingangsbereiche (Nennwerte)/ Eingangswiderstand	
• Spannung	±10 V/50 kΩ
• Strom	±20 mA/105,5 kΩ
Potentialtrennung / in Gruppen	Ja / 4

zu	
Auflösung bipolar	11 Bit + VZ
Wandlungszeit	
• je Kanal	100 μ s
• je Baugruppe	400 μ s
Grundfehlergrenze (Gebrauchsfehlergrenze bei 25 °C bezogen auf den Eingangsbereich), max.	$\pm 0,9\%$
Integrierte Analogausgänge	1
Ausgangsbereiche (Nennwerte)	
• Spannung	± 10 V
• Strom	± 20 mA
Potentialtrennung / in Gruppen zu	Ja / 1
Wandlungszeit je Kanal	40 μ s
Grundfehlergrenze (Gebrauchsfehlergrenze bei 25 °C bezogen auf den Ausgangsbereich), max.	$\pm 0,9\%$

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 užití své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla.

Datum: 21. 05. 2002

Podpis: 