

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

---

Fakulta Elektrotechnická  
Katedra Řídicí Techniky



# Diplomová práce

*Vzdálené řízení modelu technologie*

Martin Žoldák  
2004

## Prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona č.121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne .....

.....  
Martin Žoldák

## Poděkování

Tato práce by nevznikla bez pomoci, cenných rad, podnětů a připomínek vedoucího diplomové práce Doc. Ing. Jana Bílka Csc., dále pak Ing. Pavla Burgeta, Ing. Zdeňka Šebka a Doc. Ing. Bohuslava Kirchmanna Csc. V neposlední řadě bych rád poděkoval Ing. Ondřeji Novotnému a p. Čmelíkovi za výpomoc při zhotovení mechanické části třídícího mechanismu.

Můj vděk patří také všem, kteří mě při práci podporovali i jinak než odbornými radami.

## Abstrakt

V této diplomové práci je popsán způsob vzdáleného řízení modelu technologie, který lze použít také k distančnímu vzdělávání. Aktuální stav modelu je monitorován webkamerou. Model technologie je reprezentován třídícím mechanismem, který třídí míčky podle hmotnosti. Míčky jsou váženy digitální vahou firmy Sartorius. Použitý distribuovaný řídicí systém firmy Siemens, včetně vloženého řídicího systému, je napojen na Terminal Server, který umožňuje vzdálený přístup. Terminal server zároveň slouží k vizualizaci průběhu řízení. Na závěr je pomocí statistických metod vyhodnocena spolehlivost třídění míček.

## Abstract

In these diploma thesis is described way of remote control of the scale model technology, which can be used also to point of distance education. Actual state of the model is monitoring by web - camera. Model of technology is represented with sorting mechanism, which sorts the ball according to weight. The ball is weighted by precision digital balance Sartorius. In this model is used distributed control system Siemens including embedded control system that are connected to a Terminal Server. The Terminal server allows distant access and there is used to visualization of control process. At the end is analyzed the reliability of the ball sorting, by the help of a statistical methods.

# Obsah

<b>Obsah</b>	<b>iv</b>
<b>Úvod</b>	<b>v</b>
<b>1 Vzdálené řízení</b>	<b>1</b>
1.1 Profibus	3
1.1.1 Profibus DP	3
1.2 Internet	5
1.2.1 Protokol IP	6
1.2.2 Protokol TCP	7
<b>2 Realizace vzdáleného řízení</b>	<b>9</b>
2.1 Model třídícího mechanismu	11
2.1.1 Popis modelu	12
2.1.2 Akční členy – Solenoidy	14
2.1.3 Akční člen – Fúkar	16
2.1.4 Snímače	17
2.1.5 Digitální váha Sartorius	19
2.2 Řídicí systém	24
2.2.1 Struktura řídicího systému	24
2.2.2 PLC S7-300	25
2.2.3 PLC S7-200	26
2.2.4 Gateway DP/RS 232C Link	28
2.3 Terminal Server	29
2.3.1 Operační systém Windows 2000 Server	30
2.3.2 Vizualizační software	32
2.3.3 WWW stránky	34
2.4 Webová kamera AXIS 2100	35
2.4.1 Připojení k síti Ethernet	35
2.4.2 Frekvence posílání snímků	36
<b>3 Softwarové řešení řídicího systému</b>	<b>39</b>
3.1 PLC S7-200	39
3.1.1 Ochrana akčních členů	41
3.1.2 Komunikace s modulem EM 277	45

---

3.2	PLC S7-300 . . . . .	46
3.2.1	Hardwarová konfigurace . . . . .	46
3.2.2	Převod dat z váhy na údaj o hmotnosti – FC50 . . . . .	47
3.2.3	Třídící algoritmus . . . . .	48
<b>4</b>	<b>Vizualizace třídícího mechanismu</b>	<b>52</b>
4.1	Aplikace v InTouch . . . . .	53
4.2	Zobrazení aplikace pomocí SuiteVoyager . . . . .	56
<b>5</b>	<b>Statistika – Určení hmotnostních intervalů u míčků</b>	<b>58</b>
5.1	Určení $\mu$ a $\sigma^2$ . . . . .	59
5.2	Určení intervalů z naměřených dat . . . . .	60
	<b>Závěr</b>	<b>63</b>
	<b>Literatura</b>	<b>64</b>
	<b>A Struktura přiloženého CD ROM</b>	<b>66</b>
	<b>B Konfigurace služby Terminal Services</b>	<b>67</b>

# Úvod

Vzdálené řízení je v dnešní době nejpoužívanějším způsobem řízení různých technologií nezávisle na tom, zda se jedná o technologickou nebo manažerskou úroveň. V této diplomové práci je popsán jeden z možných způsobů jak realizovat vzdálené řízení.

Byl zkonstruován model demonstrující technologii, která se má vzdáleně řídit. Jedná se o model třídícího mechanismu, umožňující třídít pingpongové míčky podle váhy. Míčky se váží na velmi přesné digitální váze firmy Sartorius. Jako řídicí systém jsou použity programovatelné automaty firmy Siemens, konkrétně PLC řady S7-200, S7-300 a komunikační modul gateway DP/RS-232C. Jedná se o distribuovaný řídicí systém, ve kterém PLC S7-200 a gateway DP/RS-232C zastávají funkci vloženého řídicího systému. Řídicí systém je na dálku zpřístupněn pomocí PC, které plní roli serveru. Na tomto serveru je možné naprogramovat třídící algoritmus a nahrát ho do řídicího systému. Dále je možné sledovat a ovlivňovat průběh algoritmu pomocí vizualizačního software od firmy Wonderware.

Celý model třídícího mechanismu je koncipován tak, aby mohl sloužit také k distančnímu vzdělávání. Proto jeho součástí je webkamera firmy AXIS, která monitoruje aktuální stav modelu. K serveru je možné se vzdáleně připojit pomocí sítě Internet a tudíž lze třídící algoritmus programovat odkudkoliv, kde je tato síť přístupná.

Tato diplomová práce vznikla za podpory grantu, bez kterého by nemohly být realizovány důležité součásti modelu. Jedná se o grant: *Vložené řídicí systémy a jejich vnitřní komunikace*, Grantová Agentura České Republiky, číslo projektu 102/02/1032.

Text je rozdělen do pěti kapitol, které pojednávají o problematice vzdálené komunikace, realizaci modelu třídícího mechanismu, návrhu řídicího systému a jeho programování, zpřístupnění serveru pomocí sítě Internet a vizualizaci třídícího mechanismu. V poslední kapitole je pomocí statistických metod vyhodnocena úspěšnost třídění míčeků.

# Kapitola 1

## Vzdálené řízení

S rozvojem technologií, zvláště v oblasti IT, se zvýšily možnosti vzdáleného řízení. Existuje mnoho způsobů realizace. Je nutné si předem uvědomit, na jaké úrovni se bude vzdálené řízení používat a co se od něho očekává. Hlavním kritériem je úspěšně zvládnutá vzdálená komunikace. Základní požadavky na vzdálenou komunikaci v řízení jsou především tyto:

- **Časová odezva** – tj. čas<sup>1</sup>, který uplyne od vzniku požadavku do vykonání skutečné akce v technologii. Tato vlastnost určuje, zda je možné řídit technologii v reálném čase.
- **Spolehlivost** – zaručení vykonání vzniklého požadavku a zobrazení správných hodnot z technologie. Tím je myšleno zamezení ztráty dat nebo jejich poškození při přenosu. To lze řešit použitím vhodných protokolů, které jsou založeny např. na potvrzování zpráv nebo takové, které zamezují kolizím na sběrnici.
- **Bezpečnost** – zamezení zneužití vzdálené komunikace nežádoucími subjekty. Obvykle se řeší šifrováním přenosu dat, přičemž zvláště důležité je šifrování přihlašovacích údajů do systému.
- **Dostupnost** – možnost využití již hotových veřejných sítí bez jakýchkoliv omezení<sup>2</sup>.

Splnění všech těchto požadavků zároveň není s dnešními dostupnými prostředky možné, neboť hlavním limitujícím faktorem je požadovaná časová odezva versus dostupnost připojení. V případě použití veřejných sítí, např. Internet, je dostupnost zajištěna téměř po celém světě, ale vesměs potřebné časové odezvy nelze vůbec zajistit. Z těchto důvodů lze rozdělit vzdálené řízení do dvou kategorií<sup>3</sup>:

1. **Technologická úroveň**, kde se řídicí systém (ŘS) decentralizuje od jednoho celku do více dílčích subjektů a obvykle je ovládán z jednoho centrálního místa (velín).

---

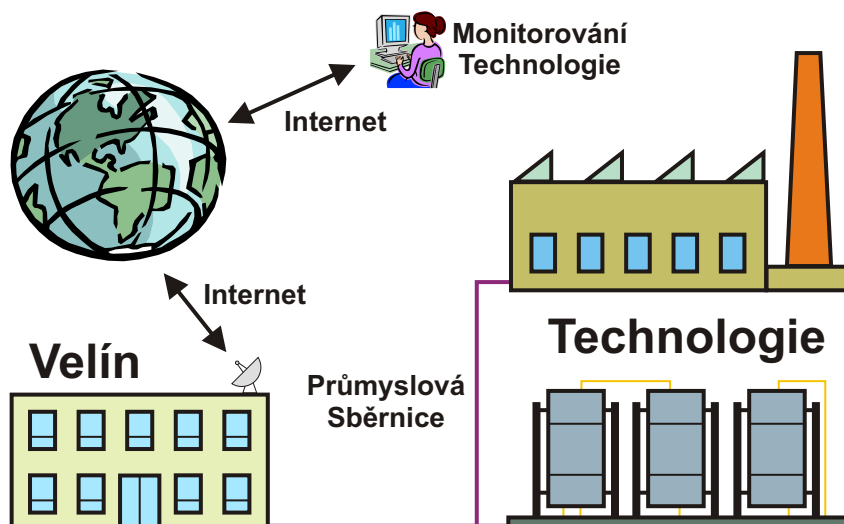
<sup>1</sup>Tento čas by měl být co nejmenší.

<sup>2</sup>Bez zásahu do již vytvořených struktur sítí.

<sup>3</sup>V rámci řídicího systému se jedná o úrovně.

2. **Monitorovací úroveň**, resp. manažerská, kde je potřeba zobrazovat všechny stavy, případně i měnit stavy technologie z jiných míst než je velín.

Na obrázku 1.1 jsou vidět pozice obou úrovní.



Obrázek 1.1: Technologická a monitorovací úroveň

První případ se řeší nasazením průmyslových sběrnic, které jsou speciálně vyvinuty pro použití v technologické úrovni. Hlavním kritériem je krátká časová odezva takového spojení a jeho odolnost proti poruchám nebo přetížení. Nejrozšířenějšími zástupci jsou Profibus, Interbus apod. V druhém případě existuje mnoho způsobů řešení a zaleží, kde mají být použity. Pokud jde o přístup ke všem datům technologie v rámci nějakého areálu nebo daného celku přístupových míst, je možné si vybudovat vlastní privátní síť. Ale pokud je potřeba přistupovat k technologii pomalu odkudkoliv, je lepší využít již vybudovaných veřejných sítí jako Internet (používající standardu Ethernet) případně GSM atd. než si budovat vlastní rozsáhlou síť.

Privátní síť je možné realizovat buď pevnou strukturou (pokud jsou přesně známy přístupové body, ze kterých se má přistupovat k technologii) nebo pomocí bezdrátového spojení, je-li třeba nezávislost na poloze v rámci areálu. Bezdrátové spojení se realizuje vesměs různými rádiovými sítěmi. V tomto případě je zvláště nutné zajištění bezpečnosti sítě proti odposlechu nebo přímo proti vniknutí do systému, protože je snadný a hlavně těžko lokalizovatelný odposlech rádiových vln. Efektivní ochrana proti odposlechu rádiových vln, kromě šifrování<sup>4</sup> přenášených dat, je častá a hlavně náhodná (resp. pseudonáhodná) změna kanálu, po kterém se přenášejí data. Na tomto principu funguje síť GSM, kde se náhodně přepíná mezi 128 kanály několikrát za vteřinu.

<sup>4</sup>Je nutné si uvědomit, že bezpečné šifrování dat je náročné na výpočetní výkon zařízení, která budou data šifrovat resp. dešifrovat.



V případě použití veřejných sítí, zvláště Internetu v kombinaci s GSM, je možno přistupovat k technologii pomalu odkudkoliv z "civilizovaného" světa. Avšak zde platí dvojnásob zabezpečení přístupu k datům technologie.

V této diplomové práci jsem na technologické úrovni ŘS použil sběrnici Profibus, což je ovlivněno použitým ŘS. Pro vzdálenou komunikaci (resp. monitorovací úroveň) s ŘS, s přihlédnutím na budoucí využití této diplomové práce, jsem zvolil veřejnou síť Internet, která je nejrozšířenější a nejdostupnější síť na světě. V následujících dvou kapitolách jsou popsány základní charakteristiky Profibusu a Internetu.

## 1.1 Profibus

Profibus patří mezi jednu z nejrozšířenějších průmyslových sběrnic. Je to dáno tím, že se jedná o otevřený standard, který je dostupný všem výrobcům automatizačních zařízení. Norma definuje tři varianty sběrnice Profibus. Rozdíl mezi nimi je v oblasti použití:

**Profibus FMS** (*Fieldbus Message Specification*) je historicky nejstarší. Je koncipován pro acyklickou komunikaci (jen na žádost) mezi automatizačními zařízeními jako jsou PLC, PC, HMI. Není zde kladen důraz na krátkou dobu odezvy. Není vhodný pro řízení v reálném čase.

**Profibus DP** (*Decentralized Periphery*) je určen pro komunikaci mezi automatizačním zařízením (aktivním účastníkem komunikace) a distribuovanými vstupy-výstupy (pasivními účastníky). Aplikační oblast této sběrnice se označuje jako technologická. Je vhodná pro řízení v reálném čase.

**Profibus PA** (*Process Automation*) využívá komunikačního protokolu Profibus DP, ale navíc může být nasazen v prostředí s nebezpečím výbuchu.

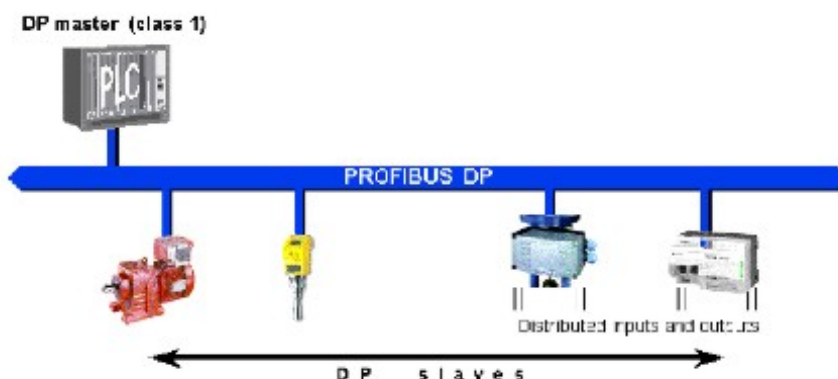
V této diplomové práci je použit Profibus DP.

### 1.1.1 Profibus DP

Profibus DP byl původně čistě firemní protokol firmy Siemens, dříve označovaný jako SINEC L2. Protokol Profibus DP je součástí normy EN50170. Oblastí použití je výrobní automatizace. Jedná se tedy o protokol určený pro rychlou cyklickou komunikaci (výměnu údajů) mezi automatizačními systémy a distribuovanými periferiemi. Jako automatizační zařízení se běžně používají programovatelné automaty (angl. PLC – *Programmable Logic Controller*), označované jako **master**. Této stanici se také říká aktivní účastník. Jako distribuované periferie se pak používají vstupně-výstupní (I/O) moduly, senzory (čidla), akční členy a inteligentní zařízení. Ty se označují jako **slave**. Stanici slave se také říká pasivní účastník. Každé stanici na sběrnici je přidělena adresa s tím, že stanice master by měly mít nižší hodnotu adresy než stanice slave. Maximální počet stanic na sběrnici je 126 (adresy 0-125). Profibus DP využívá topologie typu Bus<sup>5</sup> viz obrázek 1.2.

---

<sup>5</sup>Bus je lineární liniová sběrnice.



Obrázek 1.2: Komunikace typu Master–Slave

Protokol Profibus DP je rozdělen do tří vrstev. V rámci ISO-OSI se jedná o vrstvy fyzickou a spojovou (nazývanou též linkovou). Třetí vrstva je uživatelská<sup>6</sup>, která definuje strukturu přenášených dat. Jako rozhraní mezi uživatelskou a spojovou vrstvou jsou definovány přístupové body, zvané SAP<sup>7</sup>, k jednotlivým službám Profibusu DP. Spojová vrstva u Profibusu DP, značená FDL<sup>8</sup>, zajišťuje spolehlivost přenášených údajů fyzickou vrstvou. Fyzická vrstva může být realizována buď pomocí RS-485 nebo optické přenosové cesty (označované FO–*Fibre Optic*). Rozdíl mezi RS-485 a FO je především v maximální přípustné délce vedení v rámci jednoho segmentu sběrnice.

Vrstva FDL zajišťuje komunikaci mezi zařízeními typu master a slave. Rozlišuje se mezi komunikací typu **master-slave** a **multi-master** s tím, že mohou být obě současně kombinovány. Komunikaci typu master-slave si lze představit tak, že na sběrnici je připojena jen jedna stanice typu master a několik stanic typu slave viz obrázek 1.2. Hlavním koordinátorem na sběrnici je stanice typu master. Ta má seznam všech stanic slave, které jsou připojeny ke sběrnici. Tento seznam si vždy v průběhu cyklické komunikace obnovuje. Stanice master vysílá telegramy stanicím slave a ty na tyto telegramy odpovídají. Telegramy jsou datové rámce (max. 244 bytů) posílané po sběrnici, pomocí kterých komunikují stanice mezi sebou. Jsou přesně definovány časy, do kdy musí resp. kdy ještě může stanice slave vyslat telegram. Stanice slave nemůže sama od sebe posílat telegramy, aniž by k tomu nebyla vyzvána stanicí master. Tudiž komunikace mezi slave-slave není z výše popsaného principu komunikace přípustná. Tedy stanice slave pouze odpovídá na otázky masteru a nemůže sama od sebe začít komunikovat bez vyzvání. Tímto mechanismem je zajištěno, že na sběrnici nemůže současně přistupovat více stanic a tudíž nedochází ke kolizím.

Profibus DP umožňuje také současné připojení více stanic master na sběrnici. Pak se jedná o komunikaci typu multi-master, kde se využívá deterministického přístupu ke sběrnici pomocí tzv. token-pass. To znamená, že na sběrnici v daném časovém

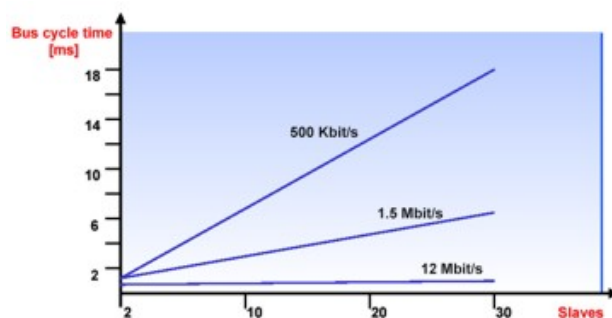
<sup>6</sup>Sice v rámci ISO-OSI není definována, ale je nejbližší aplikační vrstvě.

<sup>7</sup>Service Access Points.

<sup>8</sup>Fieldbus Data Link.

okamžiku může přistupovat pouze jedna stanice master, a to ta, která má pověření (má v daném okamžiku přidělen token). Vzhledem k tomu, že po sběrnici koluje jen jeden token, tak může ke sběrnici přistoupit v daném okamžiku jen jedna stanice. Tím je zabráněno kolizním stavům, kdy by současně přistoupilo ke sběrnici více stanic master. Pokud jsou ještě v tomto případě na sběrnici stanice slave, pak každá stanice master může provádět komunikaci typu master-slave popsanou v předchozím odstavci. Ale není zaručeno, že každá stanice master dostane odpověď na svůj telegram od každé stanice slave (nebo stanice slave zareaguje na telegram zapsáním příslušných hodnot na výstup), neboť je možné, že si některá ze stanic master uzamkla stanici slave pro svou potřebu. Pak uzamčená stanice slave reaguje pouze na telegramy stanice master, kterou byla uzamčena.

Každá stanice master má shora omezenou dobu (lze ji nastavit), po kterou může přistupovat na sběrnici v rámci jednoho cyklu. Po tuto dobu může posílat telegramy ostatním stanicím. Stanice slave pouze odpoví na přijaté telegramy. Tím je zajištěna maximální délka jednoho cyklu sběrnice a tudíž i **zaručené časové odezvy**, což je **obrovskou výhodou sběrnice Profibus DP**. Při maximální přenosové rychlosti<sup>9</sup> sběrnice je pro přenos 512 bitů vstupních a 512 bitů výstupních dat zapotřebí doba pouze kolem 1 ms. Závislost doby cyklu sběrnice a tedy i maximální doby odezvy na přenosové rychlosti je na obrázku 1.3.



Obrázek 1.3: Závislost doby cyklu sběrnice na počtu připojených stanic slave

## 1.2 Internet

Síť internet využívá komunikačního standardu Ethernet. Ethernet je technologie, která byla vynalezena a uznána jako efektivní prostředek přenosu dat na poli kancelářské komunikace a informačních technologií, přičemž během krátké doby zaznamenal prudký vzestup v oblasti lokálních počítačových sítí na celém světě. Ethernet je vybaven vyšší úrovní komunikačního softwaru v souvislosti se standardem IEE 802.3, tj. TCP/IP (*Transmission Control Protocol/Internet Protocol*) pro umožnění komunikace mezi rozdílnými systémy. TCP/IP protokol nabízí vysoký stupeň spolehlivosti

<sup>9</sup>Maximální rychlost je 12 MBit/s.

přenosu informace. Pro přenos dat v síti Ethernet se používá několik technologií s různými parametry, které se liší v typu přenosového média, přenosové rychlosti<sup>10</sup>, délce segmentu a typu přenosu. Topologie sítě je závislá na použité přenosové technologii.

Ve standardu Ethernet mohou stanice přistupovat na sběrnici užitím metody CSMA/CD (*Carrier Sense Multiple Access/Collision Detection*).

- *Carrier Sense*: Vysílač odposlouchává provoz na sběrnici.
- *Multiple Access*: Na sběrnici může přistupovat několik vysílačů.
- *Collision Detection*: Jsou detekovány kolize.

Pracovní stanice sítě Ethernet může zahájit vysílání kdykoli, pokud zjistí, že žádná jiná pracovní stanice současně nevysílá. Pracovní stanice tak poslouchá "éter", a pokud je prázdný, zahájí vysílání. Odtud také název Ethernet - "éterová síť". V průběhu přenosu, pracovní stanice neustále naslouchá a zjišťuje, jestli mezitím nezačala vysílat jiná stanice. Pokud dojde u vysílání ke kolizi (tedy pokud vysílají dvě stanice současně), počká každá pracovní stanice určitý náhodně zvolený časový interval a poté se o vysílání pokusí znovu. Při opakovaném vysílání může pochopitelně také dojít k další kolizi, pak nastupují různé speciální algoritmy. Jeden z takových algoritmů spočívá v čekání o exponenciálně narůstající interval.

Bohužel metoda CSMA/CD není pro průmyslové požadavky dostatečně spolehlivá, zvláště nelze vůbec zaručit časovou odezvu. Spolehlivost do jisté míry zajišťují až různé komunikační protokoly jako IP, TCP, UDP. Obrázek 1.4 zobrazuje pro přehlednost pozice protokolů ve vrstvách ISO-OSI.

Transportní vrstva	TCP a UDP
Síťová vrstva	IP
Spojová vrstva	Ethernet
Fyzická vrstva	

Obrázek 1.4: Pozice protokolů ve vrstvách v rámci ISO-OSI

### 1.2.1 Protokol IP

Internet se skládá z celé řady hostitelských počítačů a také zařízení, kterým se říká *směrovače (routers)*. Propojením některých počítačů vznikají sítě, přičemž některé ze sítí tvoří ostrovy. Směrovače pak propojují tyto ostrovy sítí. Každý jednotlivý hostitelský počítač i směrovač musí mít jedinečnou adresu, podle které se na toto zařízení a

<sup>10</sup>Výhodou Ethernetu je jeho vysoká přenosová rychlost, řádově i 10 GBit/s.

zpět z něj dají posílat datové pakety. Je nutné vhodně směřovat zprávy sítí, aby došly do cíle v co možná nejkratší době<sup>11</sup>. To má na starosti protokol IP (*Internet Protocol*).

Formát pro přiřazování adres popisuje protokol IP. Současně definuje mechanismus pro nespolehlivý přenos dat, kterému se říká *datagram*. Datagramy IP jsou v protokolech nižších úrovní zapouzdřeny jako součást dat. Těmito protokoly jsou například Ethernet, Token-ring atd. Protokol IP verze 4 (*IPv4*) definuje síťové adresy jako 32 bitové číslo. Adresy IP přiděluje úřad Internet Assigned Numbers Authority (*IANA*). Adresy IP se dělí do celé řady kategorií, kterým se říká *třídy*. Avšak pro samotnou komunikaci mezi aplikacemi na dvou různých počítačích samotná IP adresa nestačí. Každá aplikace musí mít možnost identifikovat svůj protějšek, se kterým chce komunikovat, protože na jednom počítači běží obvykle souběžně několik různých aplikací. Tyto aplikace mohou zpravidla komunikovat s různými aplikacemi na různých vzdálených počítačích nebo i na jednom stejném počítači. Protokoly UDP a TCP identifikují odesílající a přijímající aplikaci pomocí portů, zdrojová a cílová IP adresa pak identifikuje počítače, na kterých tyto aplikace běží. Datový proud mezi dvěma aplikacemi potom identifikuje množina následujících čtyř hodnot:

- Zdrojová IP adresa
- Číslo zdrojového portu
- Cílová IP adresa
- Číslo cílového portu

Některým důležitým a známým aplikacím jsou čísla portů již pevně přidělena. Tato předem přiřazená čísla portů se používají pouze na serveru. Čísla portů na klientech se alokují dynamicky. Jediný port na serveru může mít otevřených několik proudů, protože se na něj může připojit i několik klientů. Čísla portů získávají postupně na důležitosti, protože na jejich základě totiž často probíhá filtrování firewallů<sup>12</sup>.

### 1.2.2 Protokol TCP

Protokol IP poskytuje poměrně nespolehlivý přenos dat. Protokol TCP oproti tomu představuje téměř již spolehlivý, bezchybný, plně duplexní kanál mezi dvěma počítači. Pro přenos se využívá služeb protokolu IP, současně však definuje určité další mechanismy, které se starají o ztracené a zdvojené datagramy IP. Protokol TCP zabezpečuje také správné složení jednotlivých datagramů IP, které dorazí mimo správné pořadí. Pro zabezpečení těchto funkcí rozděluje protokol TCP vstupní proud dat do paketů, které musí být natolik malé, aby se vešly dovnitř datagramu IP. Těmto paketům se říká segmenty. Datagramy IP se očíslovají a odešlou se. Jestliže z cíle nedorazí potvrzení o jejich přijetí, odesílají se znovu. Odesílatel vždy po určitou dobu čeká

<sup>11</sup>Internet je v dnešní době tak rozsáhlá síť, že obvykle existuje několik paralelních cest, které vedou do stejného cíle.

<sup>12</sup>Jedná se obvykle o zařízení nebo software, které povoluje přístup jen k daným portům a jen příslušným aplikacím. Tímto lze zajistit poměrně vysoký stupeň ochrany počítače proti vniknutí nežádoucích subjektů.

a pokud neobdrží potvrzení o přijetí, odešle datagram znovu. Avšak když na sběrnici přistupuje současně velké množství počítačů (stanic), dochází k tak velkému množství kolizí, že ani protokol TCP není schopen vždy zabezpečit přenos dat, natož zaručit určité časové odezvy.

**Pozn.:** Pokud se přeci jen používá standard Ethernet v průmyslové praxi pro řízení, tak je jeho využití spolehlivé v málo vytížené síti, tím je myšleno, že se komunikace v daném segmentu účastní malý počet PC, aby se minimalizoval počet kolizí. Dále je využito nadstavbového protokolu (např. Profinet<sup>13</sup>) nad protokoly TCP/IP, který zabezpečuje přenos tak, aby vždy nastal, pokud existuje přenosová cesta. Některé nadstavbové protokoly používají místo protokolu TCP protokol UDP, který je více vhodný pro použití v řízení, protože nepoužívá různé mechanismy přeposílání zpráv jako TCP a tím více otevírá prostor pro nadstavbový protokol (např. RT Profinet<sup>14</sup> nebo viz [4]), který si zajistí spolehlivost přenosu dat sám. Avšak ani přes všechny výše zmíněné mechanismy se v praxi nedosahuje takové doby odezvy jako u průmyslových sběrnic (např. Profibus DP), ale výhodou je rychlejší přenos souvislých bloků dat ve prospěch Ethernetu.

---

<sup>13</sup>Dosahuje se doby odezvy řádově 10 ms.

<sup>14</sup>Softwarová "Real Time" verze Profinetu. Dosahuje se doby odezvy menší jak 10 ms.

# Kapitola 2

## Realizace vzdáleného řízení

Na základě zadání diplomové práce byl realizován model, na kterém lze demonstrovat vzdálené řízení. Jedná se mechanismus třídící pingpongové míčky dle jejich hmotnosti. Tento model bude sloužit, v rámci mezinárodního projektu LABLINK (vzdálené vzdělávání v rámci virtuálních laboratoří), studentům a nejen jim při výuce řídicích systémů. Celý model je koncipován tak, aby studenti nemuseli být přítomni v laboratoři, kde je umístěn. Jedná se vlastně o způsob distančního (vzdáleného) vzdělávání. Studenti mohou programovat řídicí algoritmus<sup>1</sup> odkudkoliv, kde mají přístup k internetu (z domova, koleje, ale i internetové kavárny). Tímto se stává výuka mnohem zajímavější a hlavně časově flexibilní, neboť virtuální laboratoř má otevřeno nonstop (součástí modelu je i osvětlení, které lze v případě potřeby dálkově zapnout).

Celý model třídícího mechanismu je koncipován tak, aby nepotřeboval pro svou funkci fyzický zásah člověka, tím splňuje podmínku vzdáleného řízení. Jelikož uživatel není fyzicky přítomen v laboratoři, je nutné publikovat aktuální informaci o stavu třídícího mechanismu. To zajišťuje webová kamera AXIS 2100, která funguje jako web-server, to znamená, že pro distribuci snímků v síti není potřeba žádného nadřazeného počítače. Celá problematika je zobrazená na obrázku 2.1. Veškeré potřebné informace o modelu jsou k dispozici na webových stránkách.

Jako řídicí systém (ŘS) jsou zvoleny programovatelné automaty firmy Siemens řady S7. Vzhledem k tomu, že se tyto PLC programují z firemního prostředí STEP7, na které jsou vázány licenční povinnosti<sup>2</sup>, bylo nutné zpřístupnit toto vývojové prostředí uživatelům na dálku. Proto je součástí modelu server (na obrázku 2.1 je zobrazen jako Terminal Server), na kterém je s licencí nainstalováno vývojové prostředí STEP7. Po řádném zaregistrování a přihlášení do rezervačního systému<sup>3</sup> se uživatelé mohou k tomuto serveru připojovat jako klienti. Tímto je zaručeno, že k serveru budou mít přístup pouze zaregistrovaní uživatelé a v daném okamžiku pouze jeden přihlášený uživatel. Rezervační systém funguje na principu firewallu, který zpřístupní server pouze přihlášenému uživateli.

Na serveru je zprovozněn vizualizační software (včetně aplikace vizualizace modelu)

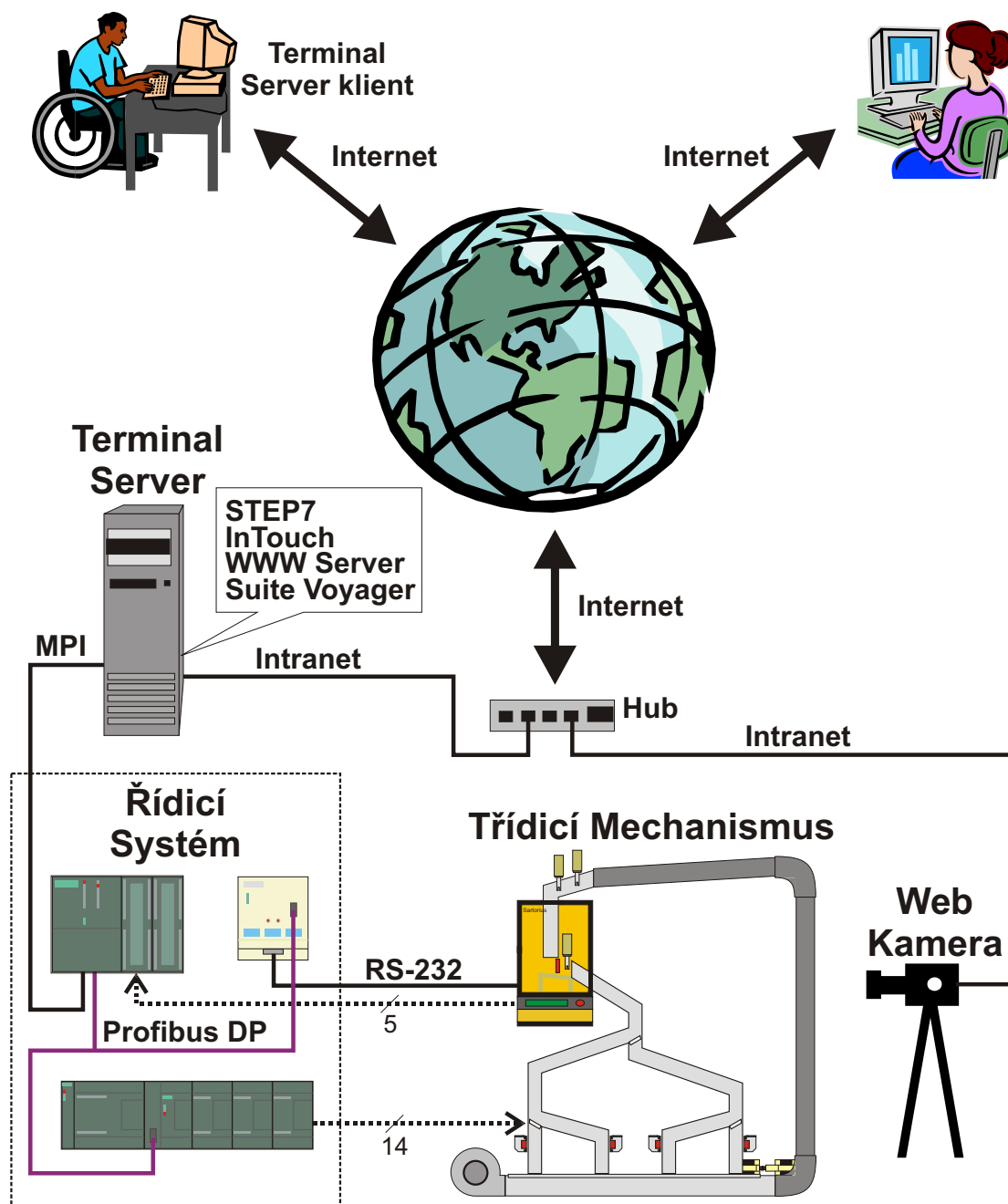
---

<sup>1</sup>Cílem studenta je naprogramovat algoritmus, který roztřídí míčky dle jejich hmotnosti.

<sup>2</sup>Cena licence je cca 50000,-.

<sup>3</sup>Rezervační systém je součástí projektu LABLINK na katedře řídicí techniky.





Obrázek 2.1: Celá koncepce modelu vzdáleného řízení

INTOUCH firmy Wonderware pro usnadnění průběhu programování řídicího algoritmu. Další forma vizualizace stavu modelu je pomocí webového prohlížeče. To zajišťuje software SUITEVOYAGER rovněž od firmy Wonderware.

Hlavní součásti modelu vzdáleného řízení (model třídícího mechanismu, řídicí systém, terminal server a web kamera) viz obrázek 2.1 jsou popsány v následujících kapi-

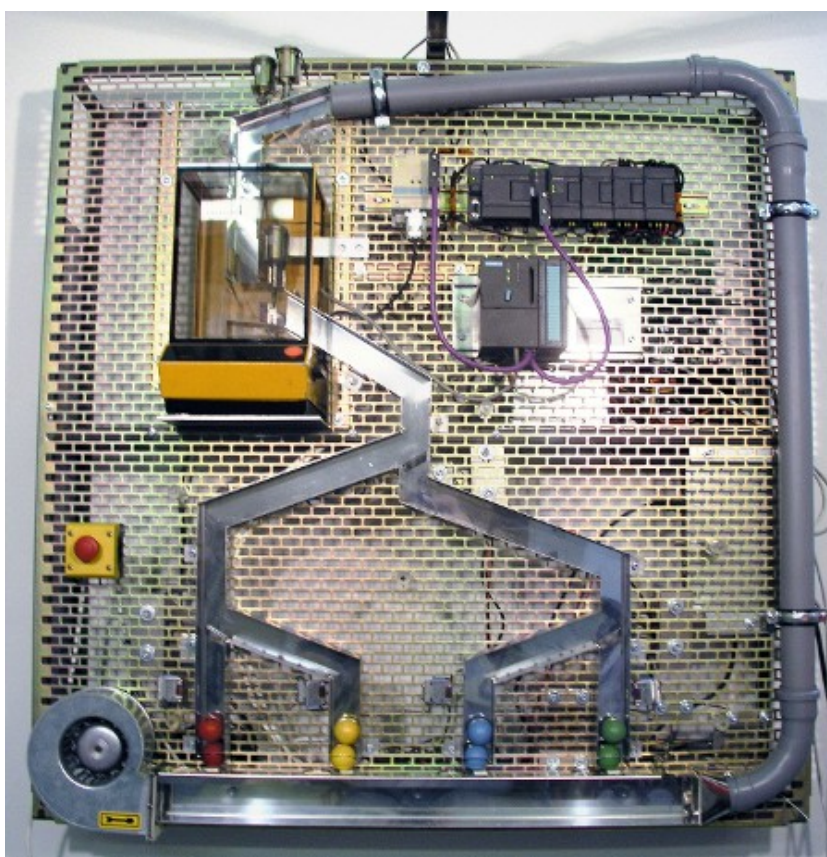


tolách.

**Pozn.:** Všechny výše zmíněné hlavní součásti jsou umístěny přímo na konstrukci modelu třídícího mechanismu.

## 2.1 Model třídícího mechanismu

Model třídícího mechanismu byl zkonstruován za účelem demonstrace algoritmu třídění míček podle jejich váhy<sup>4</sup>. Vzhledem k využití modelu pro účely distančního vzdělávání byl model navrhnut tak, aby v zapnutém stavu nevyžadoval žádnou vnější manuální obsluhu.



Obrázek 2.2: Třídící mechanismus

Do počátečního stavu se celý mechanismus dostane pouhým vyfoukáním míček do horního zásobníku, který je tvořen horní rourou viz obrázek 2.2. Z horního zásobníku padají míčky již samovolně s tím, že lze usměrnit nebo zastavit jejich pohyb pomocí zářezek a mechanických závor. Tím je splněna základní podmínka pro vzdálené řízení bez nutného zásahu člověka přímo na místě. Cílem je roztřídit všech osm míček tak,

<sup>4</sup>Aby se daly míčky opticky rozlišit, tak jsou míčky o stejné hmotnosti označeny stejnou barvou.

aby byly vždy dva přibližně stejné hmotnosti v jedné ze čtyř šachet. Roztříděné míčky v šachtách jsou vidět na obrázku 2.1 resp. 2.3.

### 2.1.1 Popis modelu

Model se skládá ze spodního zásobníku pro míčky (obrázek 2.3), který je zpřístupněn mechanickou uzávěrou. Uzávěra je ovládána pomocí dvou proti sobě umístěných velkých solenoidů. Když je uzávěra otevřená, tak veškeré míčky v šachtách propadnou do spodního zásobníku. Z tohoto zásobníku lze míčky vyfouknout pomocí fukaru do horního zásobníku, pokud je spodní zásobník uzavřen<sup>5</sup>.



Obrázek 2.3: Spodní zásobník a fukar

Z horního zásobníku jsou míčky dávkovány pomocí dvou<sup>6</sup> mechanických závor (obrázek 2.4) na váhu. Na váze je míček zastaven další mechanickou závorou (obrázek 2.5), aby mohl být zvážen. Mechanická závora je tvořena malým solenoidem, jehož jádro je v klidové poloze vysunuto a brání propadnutí míčku. Po přivedení napětí se jádro solenoidu zasune a míček může samovolně pokračovat v pohybu.



Obrázek 2.4: Mechanické závory u horního zásobníku

<sup>5</sup>Uzavření proto, aby z něj neunikal proudící vzduch z fukaru do šachet.

<sup>6</sup>Dvě závory jsou použity proto, aby šlo míčky dávkovat jednotlivě.

Přítomnost míčku na váze je identifikována pomocí snímače. Samozřejmě lze zjistit přítomnost míčku na váze změnou údaje hmotnosti, snímač je zde redundantní<sup>7</sup>.



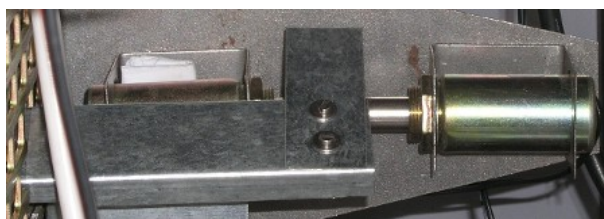
Obrázek 2.5: Mechanická závora u váhy

Po zvážení míčku je nutné nastavit zarážky (obrázek 2.6) podle toho, do které ze čtyř šachet se má umístit daná hmotnost (barva) míčku. Vysunutá zarážka způsobí vychýlení míčku ze směru jeho přirozeného pádu. Pohyb zarážky je umožněn pomocí dvou malých solenoidů, které jsou umístěny proti sobě (viz obrázek 2.7) tak, že první způsobí vysunutí a druhý zasunutí zarážky. Pohyb zarážky by šel zajistit i jedním solenoidem ve spojení s pružinou. Pak by solenoid musel překonávat odpor pružiny a tudíž by musel být výkonnější.



Obrázek 2.6: Zarážka způsobující vychýlení míčku z jeho přirozené dráhy

<sup>7</sup>Snímač umožní zjednodušit třídící algoritmus.



Obrázek 2.7: Pozice solenoidů umožňujících pohyb zarážky

Po nastavení všech zarážek může míček opustit vážící mechanismus pomocí mechanické závory (obrázek 2.5) a padat volným pádem do zvolené šachty.

### 2.1.2 Akční členy – Solenoidy

Převážnou část akčních členů tvoří solenoidy. V předchozí kapitole při popisu třídícího mechanismu byly zmíněny dva typy. Malý solenoid, který tvoří buď mechanickou závoru nebo umožňuje pohyb zarážky. Jedná se o typ "Tube/Push CD" od výrobce Emessem. Dalším zástupcem je velký solenoid starající se o uzavření resp. otevření spodního zásobníku. Výrobce je Mechetronics a jedná se o typ "AC laminated TT4".

Důležitou a relativně nepříjemnou vlastností solenoidů je jejich nelineární závislost vyvinuté síly při zasouvání jádra na vysunutí jádra. Tato závislost je vesměs exponenciální nebo kvadratická, záleží na typu. Při maximálním povoleném vysunutí jádra je síla vzniklá zasouváním jádra minimální a při úplném zasunutí jádra je síla maximální. Tento fakt bylo nutné vzít v úvahu při nasazení solenoidů na třídící mechanismus. Použitá činná dráha vysunutí jádra je 40 – 60% z maximálního rozsahu pohybu jádra. Při větším vysunutí jádra byla síla již nedostatečná k vykonání potřebného přesunu jádra se zátěží.

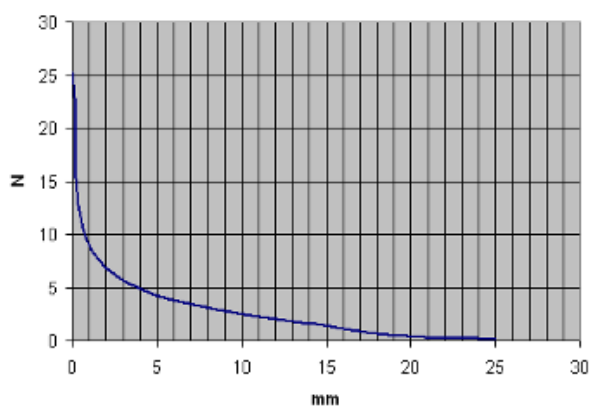
Malých solenoidů je použito celkem 9 (3 jako mechanické závory a 6 pro pohyb tří zarážek). Velké solenoidy jsou dva, umístěné proti sobě, starající se o pohyb uzávěry spodního zásobníku. Vzhledem k použitému napájení řídicího systému, byly oba typy solenoidů zvoleny pro napájecí napětí 24 V.

#### Emessem – Tube/Push CD

Jedná se o kompaktní typ s poměrně vysokým zdvihem jádra. Dva zástupci jsou na obrázku 2.4. Maximální vysunutí jádra může dosáhnout 25 mm, přičemž dosahovaná síla je 0.1 N. Závislost síly na vysunutí jádra je na obrázku 2.8. Pro funkci mechanické závory je využito zdvihu 12 mm a pro přesun zarážky jen 10 mm.

Odpor vinutí je 103  $\Omega$ , což odpovídá příkonu 5,5 W při napájecím napětí 24 V. Proto pro tento typ solenoidu je s výhodou využito společného napájení s řídicím systémem.

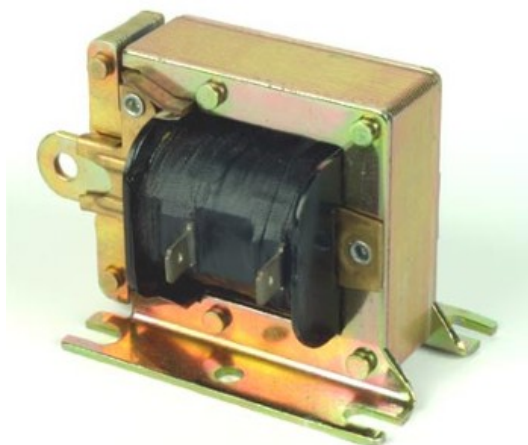




Obrázek 2.8: Závislost síly na vysunutí jádra malého solenoidu

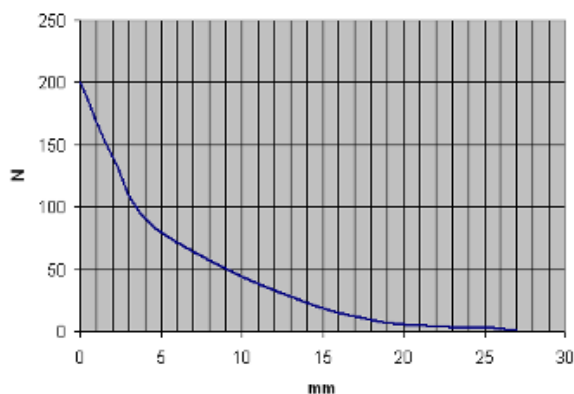
### Mechetronic – AC laminated TT4

Jedná se o dosti výkonný solenoid (obrázek 2.9) s relativně vysokou silou vznikající při zasunutí jádra. Maximální zdvih je 27 mm. Závislost síly na vysunutí jádra je na obrázku 2.10. Pro manipulaci s uzávěrem spodního zásobníku je využito zdvihu max. 14 mm, kdy je síla ještě dostatečná. Jelikož je třeba s uzávěrem pohybovat o šířku míčku, tj. přibližně 40 mm, bylo nutné použít převodní páku, která umožní při pohybu jádra solenoidu o 14 mm posunutí uzávěrem o 40 mm.



Obrázek 2.9: Velký solenoid firmy Mechetronic

Tento typ solenoidu je předurčen pro střídavé napájení. Vinutí je nakrátko, což odpovídá stejnosměrnému odporu 1.25  $\Omega$ . Při střídavém napájení se při zasouvání jádra projevuje zvyšující se indukčnost, která výrazně zvýší impedanci vinutí a tím sníží spotřebu. Avšak střídavé napájení (o síťovém kmitočtu 50 Hz) se v praxi ukázalo jako nevhodné, protože zasouvání jádra bylo doprovázeno nepříjemným hlukem a vi-



Obrázek 2.10: Závislost síly na vysunutí jádra velkého solenoidu

bracemi. Proto bylo zvoleno napájení stejnosměrné, což ale vedlo k potřebě použití výkonného napájecího zdroje. Ten by musel při napětí 24 V dodávat výkon přes 400 W a protékající proud solenoidem by činil 17 A. Takto vysoké proudové zatížení by solenoid dlouhou dobu zaručeně nevydržel (výrobce udává max. přípustnou teplotu vinutí 80 °C). Naštěstí se v praxi ukázalo, že pro přesun uzávěru spodního zásobníku postačí velké solenoidy napájet napětím 12 V. Při tomto napětí je příkon solenoidu 100 W a protékající proud 10 A. Pro tento účel byl navrhnout a zkonstruován napájecí zdroj s výkonem 160 W, 12 V/12 A.

Dalším omezením solenoidů je doba, po kterou mohou být aktivní<sup>8</sup>. Tato doba by měla být maximálně taková, aby nedošlo ke zničení (shoření) vinutí solenoidu. U malých solenoidů výrobce uvádí maximální dobu sepnutí 3-5 minut (dobu, po které lze cyklus sepnutí opakovat neuvádí). U velkých solenoidů výrobce uvádí kritickou teplotu vinutí max. 80 °C, přičemž vlivem použití stejnosměrného napájení prochází vinutím proud 10 A a tím se rychle zahřívá. Proto je žádoucí, aby solenoidy byly sepnuty jen po nezbytně nutnou dobu k vykonání úkolu, který mají plnit (tímto se prodlouží jejich životnost na maximum). Jelikož třídící algoritmus budou programovat studenti, kteří se budou teprve učit porozumět technice programování PLC, není možné spoléhat se na to, že budou spínat solenoidy po nezbytně nutnou dobu k vykonání potřebného pohybu. Proto bylo potřeba zajistit ochranu těchto akčních členů externě<sup>9</sup>, a to je dosaženo strukturou řídicího systému, který je popsán v kapitole 2.2.

### 2.1.3 Akční člen – Fukar

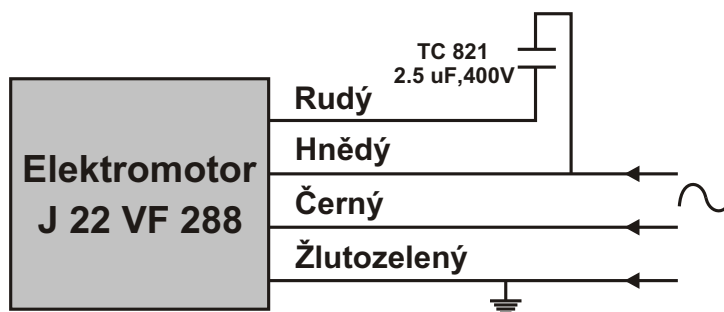
Jedná se o konstrukci<sup>10</sup>, která je při relativně malých rozměrech schopna zajistit velký objemový průtok vzduchu. Jádrem fukaru je jednofázový asynchronní kondenzátorový motorek s oběžným rotorem typu "J 22 VF 288". Elektromotorek má jmenovitý

<sup>8</sup>Vinutím protéká proud.

<sup>9</sup>Nezávisle na programu, který vytvoří studenti.

<sup>10</sup>Anglicky zvanou blower.

výkon 35 W při jmenovitém napětí 220 V a kmitočtu 50 Hz. Jmenovité otáčky jsou 2000 ot/min. a objemový průtok vzduchu při jmenovitých otáčkách  $60 \text{ dm}^3/\text{s}$ . Schéma připojení fukaru k síti je na obrázku 2.11.



Obrázek 2.11: Schéma připojení fukaru k síti

Jelikož je fukar napájen přímo ze sítě, je jeho spínání zajištěno pomocí externího relé, které spíná všechny živé vodiče. Veškeré napájení ze síťového napětí je řešeno interně (mezi přední částí modelu a zdí je odstup 20 cm, který je vytvořen rámovou konstrukcí) v třídícím mechanismu tak, aby k němu nebyl volný přístup<sup>11</sup>. Tento postup bylo nutné dodržet, protože model je umístěn v laboratoři, kde je přístupný všem studentům. Tímto je zajištěno, že na I/O svorky řídicího systému je přivedeno jen jedno společné napájecí napětí a to 24 V, které je bezpečné.

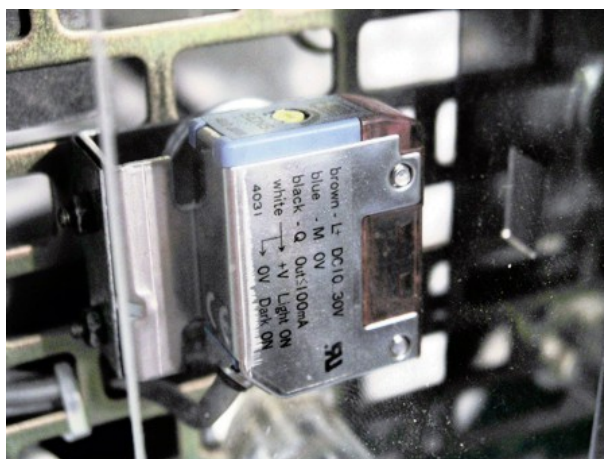
Vzhledem k objemovému průtoku vzduchu, bylo nutné zvolit váhu míčeků tak, aby je fukar byl schopen všechny naráz<sup>12</sup> vyfouknout do horního zásobníku. Váha míčeků je v rozmezí 2.2-3.4 g, přičemž odstupňování mezi jednotlivými váhovými kategoriemi je v rozmezí 150-350 mg. Další problém byl nalézt vždy dva přibližně stejně těžké míčky, aby z nich šla vytvořit dvojice (míčky se ve dvojici od sebe liší i o více jak 100 mg). Odlišení hmotnosti míčeků bylo docíleno tak, že byly použity nejlevnější vyráběné série pingpongových míčeků a v těchto sériích se stejné míčky od sebe lišily až o 0.8 g. Další odlišení hmotnosti bylo docíleno použitím míčeků různých výrobců.

### 2.1.4 Snímače

V třídícím mechanismu jsou použity fotoelektrické snímače pro kontrolu polohy míčku viz obrázek 2.12. Jedná se o typ WT 170 od výrobce Sick. Snímač funguje na principu světelné závory, která vysílá světelný paprsek a v případě odrazu paprsku od předmětu vyhodnotí odražený paprsek jako přítomnost předmětu viz obrázek 2.13. Pro správnou funkci snímače je nutné nastavit správnou citlivost, což je vzdálenost, ve které se očekává detekce předmětů. Další podmínkou je, aby se paprsek od detekovaného

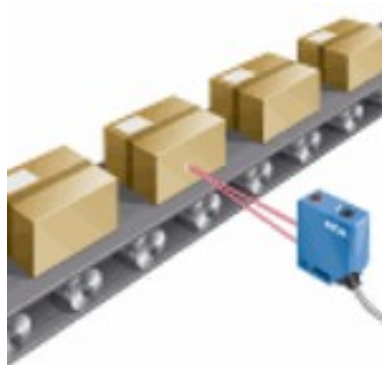
<sup>11</sup>Takto jsou ošetřeny všechny součásti modelu, které vyžadují napájení přímo ze sítě.

<sup>12</sup>Jelikož tlak vzduchu v rouře působí jen na jeden míček, tak se hmota všech míčeků počítá a dá se nahradit jakoby jedním míčkem o hmotnosti všech míčeků.



Obrázek 2.12: Snímač Sick WT 170

předmětu mohl odrážet. Tento předpoklad pingpongové míčky detekované tímto čidlem splňují. Citlivost tohoto snímače lze nastavit v rozmezí 10-550 mm.

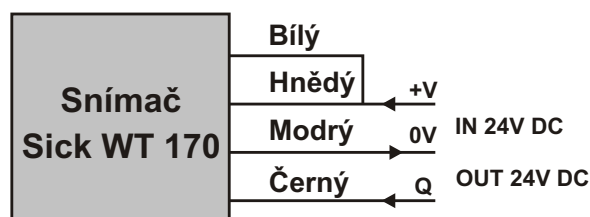


Obrázek 2.13: Funkce snímače

Snímač umožňuje celkem čtyři způsoby zapojení:

1. detekce předmětu je signalizována na výstupu *vysokou* logickou úrovní a výstup je realizován PNP tranzistorem s otevřeným kolektorem
2. detekce předmětu je signalizována na výstupu *nízkou* logickou úrovní a výstup je realizován PNP tranzistorem s otevřeným kolektorem
3. detekce předmětu je signalizována na výstupu *vysokou* logickou úrovní a výstup je realizován NPN tranzistorem s otevřeným kolektorem
4. detekce předmětu je signalizována na výstupu *nízkou* logickou úrovní a výstup je realizován NPN tranzistorem s otevřeným kolektorem





Obrázek 2.14: Schéma připojení snímače

Byl zvolen první způsob zapojení, který je zobrazen na obrázku 2.14. Základní technické parametry jsou v tabulce 2.1.

Vlastnost	Hodnota
Napájecí napětí	DC 10...30 V
Max. výstupní proud	100 mA
Doba odezvy (max.)	0.7 ms
Frekvence spínání (min.)	700 s <sup>-1</sup>

Tabulka 2.1: Parametry snímače Sick

### 2.1.5 Digitální váha Sartorius

K rozlišení hmotnosti míčků je použita digitální váha firmy Sartorius, typ 1612MP8 viz obrázek 2.15, na kterém je plocha vážícího mechanismu již upravená pro model třídícího mechanismu. Jedná se o velmi přesnou laboratorní váhu s rozlišovací schopností<sup>13</sup> 0.01 mg. Rozsah je 0-160 g. Údaj o hmotnosti je možné číst přímo z displeje váhy nebo pomocí sériového rozhraní RS-232.

Váha umožňuje změnit rozsáhlé firemní nastavení. Změnu lze provést ve speciálním režimu váhy, který se vyvolá následujícím postupem:

1. váha ve vypnutém stavu
2. stisknutí a přidržení tlačítka Tare<sup>14</sup>
3. zapnutí váhy
4. když se zobrazí na displeji kód CH5, spuštění tlačítka Tare

Po této sekvenci je možné změnit nastavení váhy tak, že se stiskem tlačítka Tare zvolí číslo kódu vlastnosti, kterou je třeba změnit. Příslušná čísla vlastností a všechny možnosti nastavení jsou v [13]. Vlastnosti, které lze nastavit jsou v tabulce 2.2.

<sup>13</sup>Tato váha váží i prach, který se postupem času usadí na vážícím mechanismu.

<sup>14</sup>Na obrázku je jako červené tlačítko vpravo dole.



Obrázek 2.15: Laboratorní váha Sartorius 1612MP8

Vlastnost	Popis
Digitální filtr	stupeň filtrace: normální, zesílený, velký a extrémní
Stabilní rozsah	lze zvolit v rozmezí 0.25-64 digitů
Formát zobrazení	poslední pozice na displeji zapnuta/vypnuta; poslední pozice resp. všechny pozice na displeji zobrazeny jen v případě stabilního údaje
Nulování váhy	buď jako stabilní údaj nebo nestabilní údaj
Modulační rychlost	možné nastavení v rozmezí 150-9600 Bd
Paritní bit	sudá, lichá, znaménková parita nebo bez parity
Posílání výstupních dat	buď na vyžádání jako stabilní/nestabilní údaj nebo automatické vysílání stabilního/nestabilního údaje o hmotnosti

Tabulka 2.2: Vlastnosti váhy Sartorius

Nulovat resp. resetovat váhu lze pomocí tlačítka Tare.

Stabilní údaj (vyhodnotí ho sama váha) nastane tehdy, když se ustálí vážící mechanismus váhy s váženým předmětem, což v ideálním případě trvá 5-10 vteřin. V případě vážení míčků váha nedojde ke stabilnímu údaji v reálně konečném čase<sup>15</sup>, protože se vážící mechanismus nemůže ustálit kvůli vzniklému tření<sup>16</sup> mezi míčkem, stěnami z plexiskla a mechanickou závorou, která zadržuje míček na vážícím mechanismu (míček je při vážení na šikmé plošince, aby mohl po zvednutí mechanické závoru samovolně opustit vážící mechanismus). Proto bylo nutné zvolit pro vážení míčků zobrazení nestabilního údaje (stejný údaj se posílá po sériové lince). Nestabilním údajem je myšleno, že se v daném okamžiku zobrazuje resp. vysílá aktuální údaj o hmotnosti, i

<sup>15</sup>Tím je myšleno, že se údaj neustálil ani po dvou minutách, což je pro třídění míčků nepoužitelné.

<sup>16</sup>Toto velmi malé tření vzniká vlivem pohybu vážícího mechanismu nahoru a dolů, při kterém se váha snaží ustabilizovat údaj o hmotnosti.

když ještě není definitivně správný. Při vážení míčků bylo zjištěno, že optimální doba vážení je někde mezi 3 až 5 vteřinami (podrobněji viz kapitola 5).

Pro nulování váhy je nastavena vlastnost nulování se stabilním údajem. To znamená, že po stisku tlačítka Tare se váha snaží ustabilizovat vážící mechanismus a teprve poté dojde k vynulování údaje. Jak již bylo psáno v předchozím odstavci, je nutné, aby na vážícím mechanismu nebyl míček, jinak se mechanismus nedostane do stabilního stavu.

Pro přenos dat po sériové lince byla zvolena modulační rychlost 1200 Bd, přenos dat bez paritního bitu a vysílání dat automaticky bez stabilního údaje. V režimu automatického vysílání bez stabilního údaje vyšle váha údaj o hmotnosti dvakrát až čtyřikrát za vteřinu. Pomocí sériového rozhraní lze také měnit nastavení váhy zasíláním zpráv příslušného formátu<sup>17</sup>, avšak této možnosti není využito, protože není třeba měnit nastavení váhy v průběhu třídění míčků.

### Sériové rozhraní RS-232

Jedná se o velmi rozšířené rozhraní standardizované Electronic Industry Association EIA. Je vhodné pro připojení typu point to point (nemá charakter sběrnice) na krátké vzdálenosti s malou přenosovou rychlostí. Má příliš volnou implementaci co se počtu a výběru signálů používaných konkrétním přístrojem týče, což přináší mnohé problémy zejména softwarového rázu. Přenosové médium není přesně definováno a je možno použít obyčejného dvou-vodičového vedení, tím ale není zaručena odolnost vůči rušení z okolí.

Modulační rychlost přenosu je nutné před každou komunikací mezi dvěma zařízeními nastavit na stejnou hodnotu, neboť určuje dobu existence jednoho bitu na lince. Přenosová jednotka je Bd (Baud rate), která je úměrná počtu změn signálových úrovní na lince za sekundu. Obvykle je v případě dvou možných signálových úrovní (nízká a vysoká logická úroveň) přenosová rychlost (bit/s) rovna modulační rychlosti.

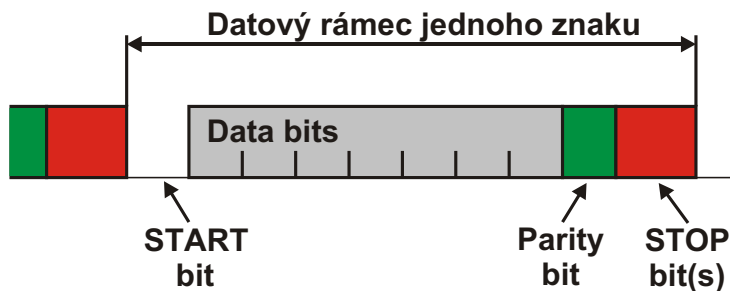
Často používané označení RS-232C definuje napěťové úrovně. Log. 1 (vysoká úroveň) odpovídá napěťové úrovni -3 až -15 V, log. 0 (nízká úroveň) úrovni +3 až +15 V. Obvody rozhraní jsou nesymetrické, proto se uvedené úrovně vztahují vůči potenciálu nulového signálového vodiče.

Součástí každého zařízení komunikujícího po sériové lince standardu RS-232 je vysílač (Transmitter- signál TxD), přijímač (Receiver- signál RxD) a řídicí signály, často realizované obvodem 8250. Jedná se o plně duplexní komunikaci umožňující současné vysílání a přijímání dat. Počet řídicích signálů není přesně definován a záleží na implementaci. Nejčastěji se používají RTS (Request To Send), CTS (Clear To Send), DSR (Data Set Read) a DTR (Data Terminal Ready).

Přenos dat po médiu se děje na úrovni znaků, což je zároveň pevný přenosový rámec, který se používá po celou dobu. Kromě datové informace je s každým znakem vysílána i informace synchronizační. Přenosový rámec začíná START bitem, který má nízkou logickou úroveň, za ním pak následují datové bity, paritní bit a STOP bity s vysokou logickou úrovní viz obrázek 2.16. Počet datových bitů (5 až 8, často 7 resp. 8),

<sup>17</sup>Tento formát je popsán v [13].

přítomnost paritního bitu, typ parity a počet STOP bitů (1 nebo 2) lze nakonfigurovat. Jedná se o asynchronní přenos využívající pro svou synchronizaci pouze START a STOP bity, proto jsou jejich logické úrovně odlišné, aby přijímač rozpoznal, zda se jedná o začátek rámece (každý rámeček oznámí svůj začátek přijímači nízkou úrovní signálu TxD).



Obrázek 2.16: Datový rámeček přenášený po RS-232

Váha umožňuje připojení k sériovému rozhraní pomocí 25 pinového D-Sub konektoru (DB25S). Popis všech signálů je v tabulce 2.3.

Pin	Vlastnost	Pin	Vlastnost
1	ground	14	signal ground
2	Data output TxD	15	Print-key
3	Data input RxD	16	Tare-key
4	external ground	17	connected
5	CTS-input	18	connected
6	connected	19	connected
7	connected	20	DTR-output
8	key	21	connected
9	connected	22	connected
10	connected	23	connected
11	connected	24	connected
12	connected	25	connected
13	connected	-	-

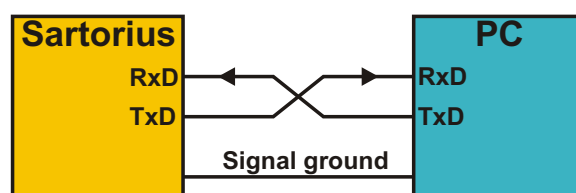
Tabulka 2.3: Význam pinů u konektoru sériového rozhraní

Aby váha mohla vysílat data, je nutné propojit na straně váhy piny 4 a 7 (nutná podmínka daná výrobcem). Dále je nutné, aby byl řídicí signál CTS v logické jedničce, což je zajištěno připojením CTS na straně váhy na pin 25. Signál CTS slouží pro synchronizaci mezi váhou a přijímací stranou<sup>18</sup> (PC atd.). V případě propojení vodičů

<sup>18</sup>Přijímací strana dává vysokou logickou úrovní signálu CTS vysílací straně najevo, že je připravena přijímat data.

key a Tare-key dojde k simulovanému stisknutí tlačítka Tare a tím k resetování váhy. Této vlastnosti je využito pro vzdálené nulování váhy propojením pinů key a Tare-key pomocí relé. V případě propojení pinů key a Print-key je dán požadavek váze o vyslání údaje o hmotnosti, pokud je váha přepnuta do režimu posílání dat na vyžádání. Ale tyto piny nejsou využity, protože váha je využívána v automatickém režimu posílání dat.

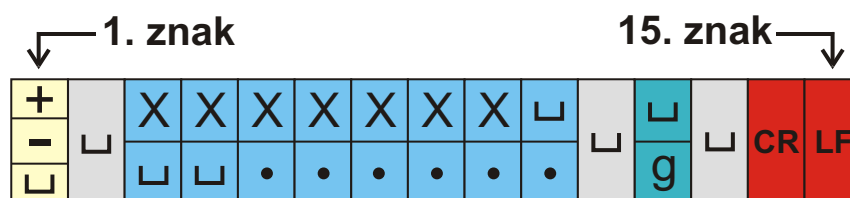
Možné připojení<sup>19</sup> váhy se zařízením, které bude přijímat data je na obrázku 2.17. Obdobné propojení váhy je použito pro komunikaci s řídicím systémem (ŘS), avšak přijímač RxD na straně váhy není připojen, protože není potřeba měnit nastavení váhy.



Obrázek 2.17: Příklad propojení pomocí RS-232

### Protokol výstupních dat

Protokol, pomocí kterého posílá váha údaj o hmotnosti, je tvořen 15-ti znaky viz obrázek 2.18. Každý přenášený znak<sup>20</sup> se skládá z jednoho start bitu, sedmi datových bitů, jednoho paritního bitu (v nastavení váhy není parity použito) a jednoho stop bitu. Tedy každý přenášený znak se skládá z deseti bitů.



Obrázek 2.18: Protokol výstupních dat

První znak protokolu může být + (2B Hex), - (2D Hex) nebo mezer<sup>21</sup> (20 Hex). Druhý je vždy mezer. Třetí a čtvrtý znak je buď mezer nebo číslice<sup>22</sup> 0-9 (30-39 Hex). Pátý až devátý znak může být buď číslice nebo desetinná tečka (2E Hex). Desátý znak je mezer nebo desetinná tečka. Jedenáctý a třináctý znak je

<sup>19</sup>Za předpokladu propojení pinů 4 se 7 a 5 s 25 na straně váhy.

<sup>20</sup>Jedná se o datový rámeček.

<sup>21</sup>Na obrázku 2.18 je vyznačená symbolem □.

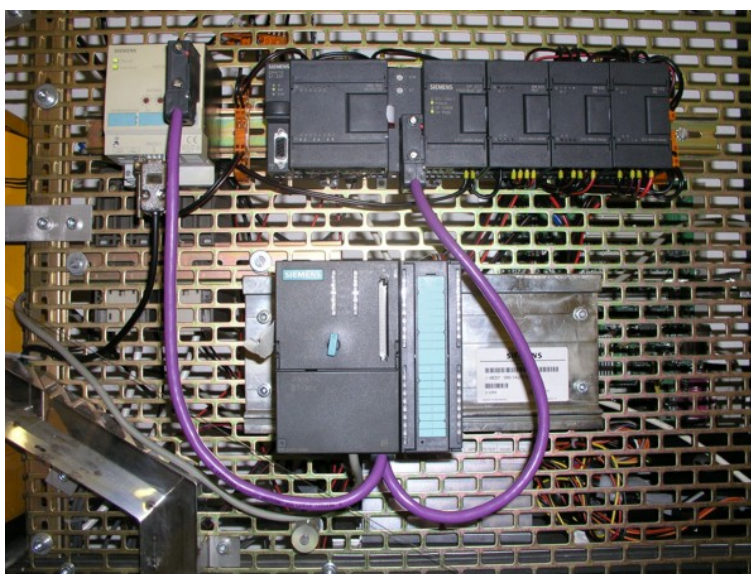
<sup>22</sup>Na obrázku 2.18 vyznačena jako X.

mezera. Dvanáctý znak je buď *mezera* nebo *g* (67 Hex) a to v případě stabilního údaje o hmotnosti. Čtrnáctý znak je *CR* (0D Hex) a patnáctý *LF* (0A Hex), přičemž tyto dva znaky jsou pomocné a slouží k identifikaci konce protokolu.

Údaj o hmotnosti je reprezentován třetím až desátým znakem protokolu s libovolně přípustnou kombinací znaků na těchto pozicích. Údaj o hmotnosti je ještě reprezentován prvním znakem, což je znaménko. Znaménko má smysl tehdy, pokud dojde k vynulování váhy s nějakým předmětem na vážícím mechanismu. Po odejmutí tohoto předmětu se musí vážený údaj logicky dostat do záporných hodnot. Stabilní údaj o hmotnosti je reprezentován dvanáctým znakem protokolu a je signalizován symbolem *g*.

## 2.2 Řídicí systém

Použitý řídicí systém (ŘS), obrázek 2.19, zajišťuje kompletní řízení celého třídícího mechanismu. Jsou k němu připojeny veškeré akční členy a snímače. Struktura ŘS je decentralizovaná, aby mohla být zajištěna ochrana akčních členů. Skládá se z programovatelných automatů a modulů SIMATIC řady S7 firmy Siemens.



Obrázek 2.19: Řídicí systém třídícího mechanismu

### 2.2.1 Struktura řídicího systému

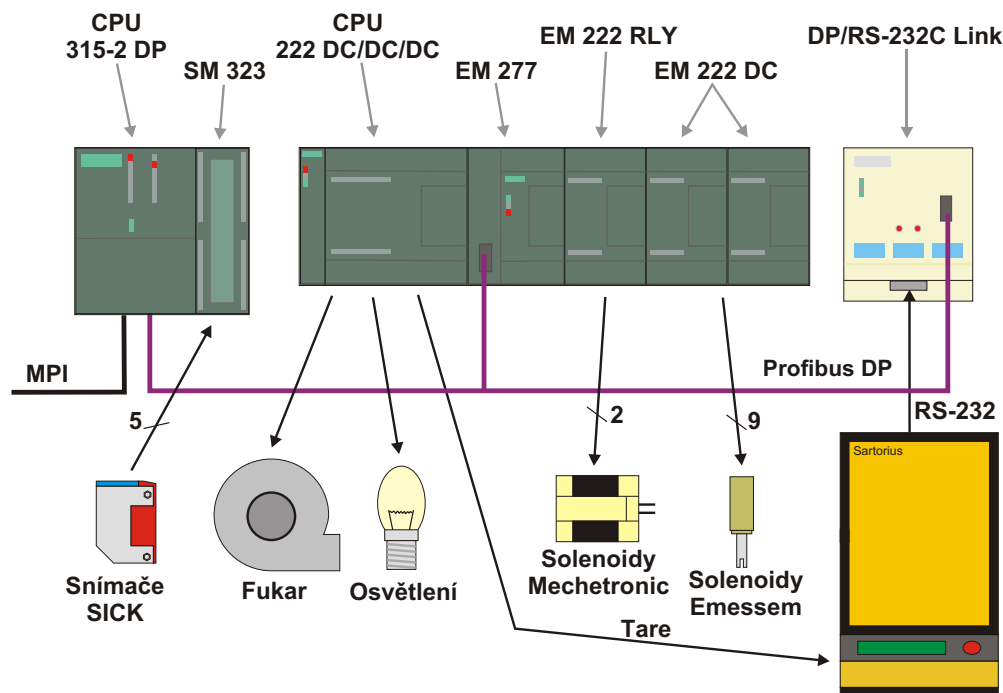
Řídicí systém lze rozdělit do tří hlavních částí propojených sběrnici Profibus:

- PLC řady S7-300
- PLC řady S7-200



### • Gateway DP/RS-232

Struktura a propojení je na obrázku 2.20. PLC řady S7-300 je na sběrnici jako DP-Master. PLC řady S7-200 a gateway DP/RS-232 jsou DP-Slave a zároveň plní funkci vloženého ŘS. Před připojením stanic DP-Slave ke sběrnici Profibus je nutné každé stanici nastavit adresu<sup>23</sup>, pomocí které bude identifikována.



Obrázek 2.20: Struktura řídicího systému

### 2.2.2 PLC S7-300

Skládá se z CPU 315-2 DP a vstupně-výstupního (I/O) modulu SM 323. Toto CPU je připojeno k PC (v našem případě Terminal Server) rozhraním MPI<sup>24</sup>, které umožňuje nahrání programu do CPU a také sběr dat z CPU. Toto PLC je hlavním článkem ŘS, neboť je zároveň stanicí DP-Master, tudíž se může dotazovat a dávat příkazy stanicím DP-Slave. Je k němu připojeno všech 5 snímačů firmy Sick přes I/O modul SM 323. K akčním členům a datům z váhy Sartorius již ale nemá přímý přístup. Může k nim přistupovat jen pomocí výše zmíněných DP-Slave stanic. Tímto oddělením akčních členů, přes DP-Slave stanice, lze zajistit jejich ochranu nezávisle na programu běžícím v CPU 315-2 DP.

<sup>23</sup>Obvykle se mechanicky nastavuje přímo na stanici.

<sup>24</sup>Multi Point Interface.

PLC řady 300 slouží ke vzdálenému naprogramování (k tomuto účelu je použit Terminal Server) třídícího algoritmu a následně k ověření funkčnosti algoritmu na modelu.

### CPU 315-2 DP

Jedná se vlastně o hlavní jednotku provádějící veškeré úkony dle naprogramovaného algoritmu. Dalším hlavním úkolem je starat se o veškerou komunikaci na sběrnicích, v našem případě Profibus<sup>25</sup> a MPI. Jeho přímou součástí nejsou žádné I/O periferie.

Disponuje celkem 64 kB pracovní paměti. Je-li použita záložní baterie a dojde-li k výpadku napájení, obsah celé pracovní paměti zůstane zachován. Pokud není použito záložní baterie, zachová se 4096 bytů paměti. Přehled nejdůležitějších parametrů je v tabulce 2.4.

Parametr	Hodnota
Pracovní paměť	64 kB
S7 časovače	128 (T0-T127)
S7 čítače	64 (C0-C63)
Datová bloky DB	128 (DB0 vyhrazen)
Funkce FC	192
Funkční bloky FB	192

Tabulka 2.4: Přehled základních parametrů CPU 315-2 DP

### SM 323

SM 323 je přídavný I/O modul pro PLC řady 300. Konkrétně je použit 16DI/16DO s 16 binárními vstupy a 16 binárními výstupy. Pracuje s napájecím napětím 24 V a proudové zatížení výstupů může být maximálně 0.5 A. Přenos vysoké logické úrovně (ve spojení s vysokorychlostními čítači) na výstup lze opakovat s frekvencí maximálně 100 Hz. Veškeré vstupy a výstupy jsou galvanicky odděleny pomocí optoelektronických polovodičových členů.

### 2.2.3 PLC S7-200

Součástí je CPU 222 DC/DC/DC<sup>26</sup>, modul EM 277 a I/O moduly EM 222 DC resp. RLY. K těmto modulům jsou připojeny veškeré akční členy včetně osvětlení, tudíž nad nimi má plnou kontrolu CPU 222. Toto CPU je připojeno ke sběrnici Profibus pomocí rozšiřujícího modulu EM 277 jako DP-Slave. EM 277 je modul, který splňuje kritéria stanice DP-Slave, a který umožňuje připojení ke sběrnici Profibus pomocí RS-485.

<sup>25</sup>Připojení ke sběrnici Profibus je pomocí RS-485.

<sup>26</sup>Dále jen ve zkratce CPU 222.



PLC řady 200 slouží k ochraně akčních členů před jejich přetížením. V případě překročení povoleného časového okamžiku sepnutí akčního členu ze strany PLC řady 300, dojde k odpojení akčního členu ze strany PLC řady 200. S7-200 zde plní roli inteligentního I/O DP-Slave zařízení. Podrobněji je této problematice věnována kapitola 3.

### CPU 222

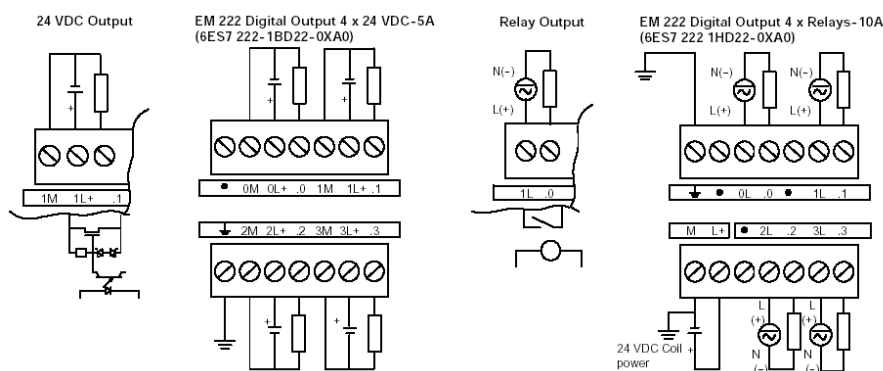
Jedná se kompaktní typ, který má v sobě integrovány I/O periferie. Konkrétně 8 binárních<sup>27</sup> vstupů a 6 binárních výstupů. Přehled nejdůležitějších parametrů je v tabulce 2.5.

Parametr	Hodnota
Paměť programu	4096 bytů
Paměť dat	2048 bytů
Časovače	256 (T0-T255)
Čítače	256 (C0-C255)
Volání/Podprogramy	0-63

Tabulka 2.5: Přehled základních parametrů CPU 222

### EM 222 DC a EM 222 RLY

Jedná se o rozšiřující I/O moduly, konkrétně EM 222 Digital Output 4 x 24VDC-5A a EM 222 Digital Output 4 x Relays-10A. Veškeré výstupy jsou galvanicky odděleny. Způsob zapojení výstupů je na obrázku 2.21. Koncovka DC znamená, že je na výstupu použit MOSFET tranzistor, který snese proudové zatížení 5 A a koncovka RLY nám říká, že jako výstupní člen je použito relé, které snese proudové zatížení až 10 A.



Obrázek 2.21: Způsob zapojení výstupů u modulů EM 222

<sup>27</sup>Proto označení DC za číslem CPU.

EM 222 RLY je použito pro přímé spínání velkých solenoidů firmy Mechetronic, kterými za provozu protéká proud 10 A. Obecnou nevýhodou použití reléově spínaných výstupů oproti polovodičovým je jejich nižší frekvence spínání a menší opakovatelnost (omezená životnost mechanických kontaktů).

### 2.2.4 Gateway DP/RS 232C Link

Jedná se o DP-Slave zařízení tvořící logickou bránu mezi sériovou linkou RS-232C a sběrnici Profibus. U rozhraní RS-232C podporuje maximální modulační rychlost 19.2 kBd. Připojení k sériové lince je pomocí 9-pinového SUB-D konektoru. Popis signálů je v tabulce<sup>28</sup> 2.6.

Pin	Vlastnost	Pin	Vlastnost
1	-	6	-
2	Data input RxD	7	RTS
3	Data output TxD	8	CTS
4	-	9	-
5	signal ground	-	-

Tabulka 2.6: Význam pinů u konektoru sériového rozhraní

U komunikace po sériové lince lze nastavit modulační rychlost a volbu paritního bitu (sudá, lichá parita nebo bez parity). Ale nelze nastavit počet datových bitů a jako implicitní hodnota jich je 8! Tento fakt je nutné brát v úvahu při čtení dat z váhy Sartorius<sup>29</sup> a vhodně ho softwarově ošetřit. Dále je možné zvolit ze čtyř komunikačních módů (podrobně popsanych v [8]), v jakém stavu bude stanice DP-Slave, po vyžádání stanici DP-Master, posílat přijatá data z RS-232. Jedná se o následující módy:

- **The 3964R Driver**
- **Start-End Delimiter Protocol**
- **Character Delay Mode**
- **Number of Character-Protocol**

Pro komunikaci s váhou je použit mód **Number of Character**, který přímo přenáší data bez jakékoliv časové konzistence. Pak je nutné zajistit rozlišení dat pomocí vhodného mechanismu (tímto mechanismem je například komunikační protokol váhy).

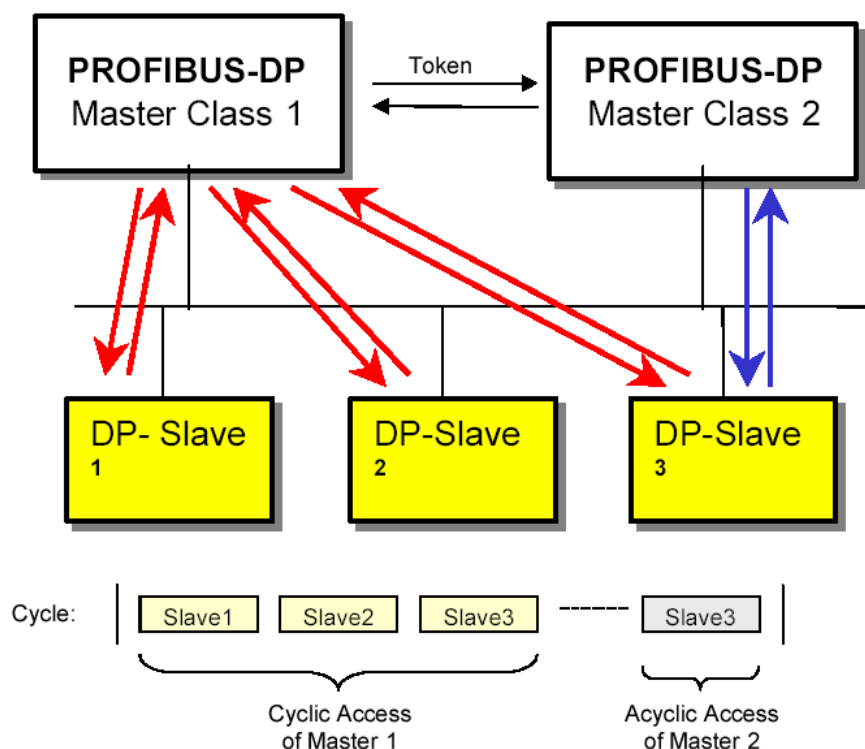
<sup>28</sup>Výrobce uvádí, že signály RTS a CTS ke komunikaci nepoužívá.

<sup>29</sup>Připomínám, že její datový rámec se skládá ze sedmi datových bitů.

## 2.3 Terminal Server

Terminal Server<sup>30</sup> je PC s operačním systémem, které umožňuje vzdálené připojení a je prostředníkem ŘS pro vzdálenou komunikaci. Umožňuje vzdáleně programovat CPU 315-2 DP a zároveň z něj číst data pro případnou vizualizaci. Terminal Server je připojen k veřejné síti Internet a s řídicím systémem komunikuje prostřednictvím PCI karty Siemens CP 5611 přes rozhraní MPI viz obrázek 2.1.

Komunikační karta CP 5611 tvoří tzv. PG/PC<sup>31</sup> rozhraní mezi PC a ŘS. Kromě komunikačního protokolu MPI dovoluje komunikovat pomocí protokolu Profibus. V tomto případě bude PC jako stanice DP-Master třídy 2 (DPM2). To znamená, že na sběrnici nepřistupuje nepřetržitě, nýbrž jen v potřebných okamžicích (acyklická komunikace). Používá se jako manažerská stanice k monitorování dat nebo k diagnostice průběhu dat na sběrnici. Protějškem k DPM2 je DPM1, který přistupuje na sběrnici nepřetržitě a stará se o cyklickou komunikaci. Zástupcem je obvykle PLC. Příklad cyklické (DPM1) a acyklické (DPM2) komunikace ilustruje obrázek 2.22.



Obrázek 2.22: DPM1 a DPM2 na jedné sběrnici

Použitý operační systém (OS) vychází z požadavku aplikace, pomocí které se programuje CPU 315-2 DP. Jedná se vývojové prostředí STEP7 firmy Siemens, které

<sup>30</sup>Dále je též používán ve zkratce název server.

<sup>31</sup>PG-Programming device.

běží pod OS WINDOWS firmy Microsoft. STEP7 je nainstalováno na Terminal Serveru včetně všech nutných licencí. K vizualizaci dat (a zároveň ke komfortnímu ovládání třídícího mechanismu) bylo zvoleno prostředí INTOUCH a SUITEVOYAGER firmy Wonderware, které rovněž běží pod OS WINDOWS. Jako OS byl konkrétně zvolen WINDOWS 2000 SERVER, který kromě výše zmíněné podpory potřebných aplikací umožňuje vzdálenou správu a vzdálené připojení bez nutnosti instalace dalšího software. Tím je splněna základní podmínka pro vzdálené řízení.

Server je umístěn přímo v konstrukci třídícího mechanismu a není k němu přímý přístup. K okolí je připojen pouze pomocí sítě Internet a nejsou k němu připojeny klávesnice, myš ani monitor<sup>32</sup>. Umístění serveru přímo do konstrukce modelu bylo z důvodů kompaktnosti modelu, aby byl nezávislý na okolí. Server se dispozičně nachází za ŘS.

### 2.3.1 Operační systém Windows 2000 Server

Hlavní důvod volby této verze OS WINDOWS je integrovaná služba **Terminal Services**<sup>33</sup>. Služba je podpora OS, která umožňuje spouštět aplikace na pozadí právě jako služby, a to i tehdy, pokud není nikdo přihlášen do systému. To znamená, že při startu systému se automaticky spouští všechny povolené služby, které běží na pozadí aniž by se někdo přihlásil do systému.

Server je nakonfigurován tak, aby se k němu dalo kdykoliv připojit jako *TSE Client*. Jsou vytvořeny dva základní účty:

- **student** – je členem skupiny *Power Users* a bude sloužit studentům k přihlášení k TSE. Má omezená přístupová práva na disk a nemůže provádět změny v systémovém nastavení OS. Může spouštět všechny potřebné aplikace.
- **spravce** – je členem skupiny *Administrators* a slouží ke vzdálené správě serveru. Uživatelský účet Administrator nebyl ke vzdálenému přístupu použit z bezpečnostních důvodů, neboť hackeři se chytají klíčových slov Administrator apod.

WINDOWS 2000 SERVER je možné provozovat v jednom ze dvou režimů:

- **Režim vzdálené administrace** – umožňuje omezenému počtu administrátorů vzdáleně spravovat server. Vzdálení uživatelé mohou být pouze členy skupiny Administrators. Toto nastavení minimalizuje dopad na výkon serveru.
- **Režim aplikačního serveru** – umožňuje uživatelům vzdáleně spouštět jednu nebo více aplikací. Vzdálený přístup je povolen uživatelům ze všech skupin. Toto nastavení optimalizuje dobu odezvy programů.

Pro provoz serveru byl zvolen režim aplikačního serveru, neboť se bude používat ke vzdálenému spouštění aplikací a je také nutné povolit přístup uživatelům (*student*),

<sup>32</sup>Samozřejmě ke konfiguraci serveru a k instalaci softwaru byly tyto periferie použity, ale pro funkci serveru v rámci třídícího mechanismu již nejsou potřebné.

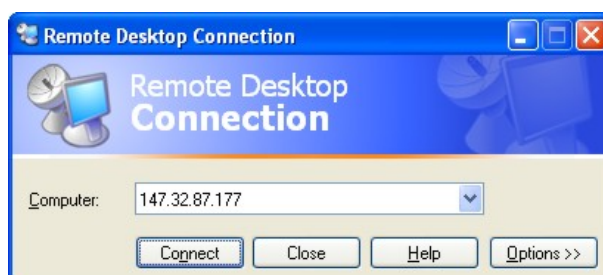
<sup>33</sup>Ve zkratce TSE.

kteří nejsou ze skupiny Administrators. Avšak pro konfiguraci a instalaci aplikací (i když je prováděno lokálně) je lepší použít režim vzdálené administrace, protože v aplikačním režimu není standardně povoleno instalovat nové aplikace. To je povoleno jen v případě otevření prostředí *Přidat nebo odebrat programy*, ale i tak není zaručeno správné nainstalování všech aplikací, zvláště těch, které potřebují v průběhu instalace resetovat počítač (např. STEP7).

### Terminal Services

Tato služba umožňuje vzdálený přístup k ploše a téměř<sup>34</sup> ke všem aplikacím pomocí klientské aplikace TERMINAL SERVER CLIENT (TSE Client), která je přímo součástí OS nebo se dá stáhnout na stránkách firmy Microsoft nebo na webových stránkách modelu. To znamená, že pokud je spuštěna služba Terminal Services a není blokován firewallem port 3389, je možné se k serveru připojit odkudkoliv ze světa, kde je přístup k Internetu.

Služba TSE při přihlášení vytvoří každému uživateli vlastní pracovní plochu, jako kdyby pracoval na svém počítači. WINDOWS 2000 SERVER umožňují vzdáleně se přihlásit najednou více uživatelům, kde každému uživateli je přidělena vlastní vzdálená pracovní plocha, ale všichni pracují na jednom počítači a sdílejí jeho výpočetní výkon. K tomuto provozu je potřeba výkonný hardware a zvláště velká kapacita operační paměti.



Obrázek 2.23: Aplikace TERMINAL SERVER CLIENT

Přihlášení k serveru na vzdálené stanici probíhá v následujících krocích:

1. Spuštění aplikace TSE Client, obrázek 2.23.
2. Volba nastavení<sup>35</sup>: přenos grafických detailů (vzhledem k rychlosti sítě) a velikost pracovní plochy (rozlišení v pixelech). Velikost pracovní plochy je omezena pouze ze strany klientského počítače a to maximálním použitým rozlišením.

<sup>34</sup>Některé aplikace, zvláště mladšího data, hlídají, jestli nejsou spuštěny v režimu vzdálené plochy, a pokud ano, nedovolí jejich spuštění. Potom je obvykle potřeba použít TSE instalaci dané aplikace, na kterou se většinou vztahují jiné licenční podmínky, neboť služba TSE umožňuje současně přistupovat k serveru více uživatelům.

<sup>35</sup>Tlačítko Options>>.

3. Zadání IP adresy nebo názvu (včetně domény) serveru do políčka nazvaného Computer, ke kterému je třeba se připojit. Stisknutí tlačítka Connect.
4. Otevře se okno zvoleného rozlišení s běžným přihlašovacím dialogem. Po zadání příslušného jména a hesla je možné přihlásit se ke vzdálené ploše a začít pracovat. V případě, že uživatel zvolil v nastavení maximální rozlišení, tak bude celá lokální pracovní plocha nahrazena (překryta) vzdálenou pracovní plochou.

Služba TSE používá ke komunikaci mezi serverem a klientským počítačem protokol RDP (*Remote Desktop Protocol*) ve spojení s protokolem TCP. RDP je založen na rozhraní standardu rodiny protokolů ITU T.120. Jedná se o vícekanálový protokol, který dovoluje po více virtuálních kanálech šifrovaný přenos obrazu, kliknutí myši a úhozy do klávesnice. Podporuje až 64000 oddělených kanálů pro přenos dat. RDP obsahuje vlastní ovladač obrazovky, který na straně serveru poskytuje data pro přenos obrazu klientovi. Na serveru RDP se používá pro příjem dat od klienta vlastní virtuální ovladač klávesnice a myši. Pro přenos dat protokolem RDP bylo zvoleno šifrování 128 bitovým klíčem (maximální možná volba). Šifrování je obousměrné, tzn. z klienta na server a opačně. Tímto šifrováním je dostatečně zabezpečena ochrana v přístupu k serveru, který zároveň umožňuje přístup k ŘS.

Výhodou protokolu RDP je malý transfer dat po síti, protože posílá jen změny v obraze (a ještě k tomu komprimované) místo neustálého posílání statického obrazu celé plochy. V praxi se to projeví tak, že i přes modemové spojení<sup>36</sup> nedochází ke znatelným prodlevám v ovládání vzdálené pracovní plochy.

Nevýhodou použité verze protokolu RDP u WINDOWS 2000 SERVER je přenos obrazu s barevnou hloubkou pouze 256 barev. S tímto faktem se muselo počítat při realizaci vizualizační aplikace. Tento nedostatek je odstraněn až u OS WINDOWS XP, kde je hloubka barev omezena pouze ze strany klientského počítače a lze ji zvolit v nastavení aplikace TSE Client. Avšak omezení ze strany WINDOWS XP je v tom, že umožňuje současně se přihlásit pouze jednomu uživateli.

### 2.3.2 Vizualizační software

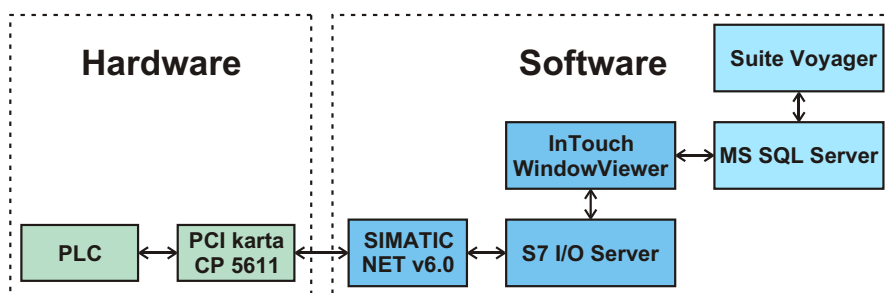
K vizualizaci dat je použit software INTOUCH v8.0 firmy Wonderware. InTouch v sobě zahrnuje aplikace WINDOWMAKER a WINDOWVIEWER. WINDOWMAKER slouží k naprogramování vizualizační aplikace a WINDOWVIEWER poskytuje runtime prostředí pro vizualizační aplikaci, ve kterém probíhá ovládání ŘS.

K ovládání ŘS samotný INTOUCH nestačí. Je potřeba zprostředkovat data z PLC do WINDOWVIEWERU. K tomuto účelu slouží aplikace S7 I/O SERVER firmy Wonderware ve spojení se SIMATIC NET v6.0 od firmy Siemens. Logická návaznost vizualizačního software je na obrázku 2.24.

K datům z PLC má nejbližší SIMATIC NET přes kartu CP 5611. Aby mohl přistupovat k datům S7 I/O SERVER, je nutné nakonfigurovat SAPI<sup>37</sup> databázi u SIMATIC

<sup>36</sup>Za předpokladu nastavení minimálních detailů u klientské aplikace. Minimálními detaily je myšleno, že se přenáší běžná pracovní plocha bez případného obrázku na pozadí a přenos bitmap není v plné kvalitě.

<sup>37</sup>Speech Application Programming Interface.



Obrázek 2.24: Hierarchie softwaru pro vizualizaci

NET. Konfigurace spočívá ve vytvoření virtuální PC Station, která běží jako služba v rámci SIMATIC NET. Konfigurace se provádí v prostředí STEP7, kde se vytvoří PC Station, která je připojena k rozhraní MPI. Podrobný postup jak nakonfigurovat SIMATIC NET je v [16]. Pokud je SAPI správně nakonfigurována, tak je možné nadefinovat přístupový bod (S7 Topic Definition) u S7 I/O SERVERU.

Aplikace WINDOWVIEWER přistupuje k datům z PLC pomocí S7 I/O SERVERU, se kterým komunikuje DDE<sup>38</sup> protokolem. K ovládání PLC z prostředí WINDOWVIEWER je nutné, aby běžely aplikace S7 I/O SERVER a SIMATIC NET. Ty jsou spouštěny jako služby, tudíž studentům, kteří se přihlásí ke vzdálené ploše, stačí spustit WINDOWVIEWER a mohou pomocí vizualizační aplikace<sup>39</sup> ovládat PLC, což jim usnadní práci při programování třídícího mechanismu.

### SuiteVoyager

Je další alternativou firmy Wonderware k vizualizaci dat. V tomto případě k prohlížení dat postačuje normální MS Internet Explorer. SUITEVOYAGER v 2.0 je webové rozhraní, které běží na webovém serveru, konkrétně IIS<sup>40</sup> 5.0 firmy Microsoft. Žádný jiný webový server není podporován. IIS 5.0 je služba OS WINDOWS 2000 SERVER a je přímo jeho součástí. Pro správnou funkci SUITEVOYAGERU je nutné u IIS nastavit přístup na základě ověřování (uživatelské jméno a heslo).

Veškerá konfigurace SUITEVOYAGERU se provádí pomocí webového prohlížeče viz obrázek 2.25. Pro přístup je nutné se vždy přihlásit, ale uživatelské jméno a heslo se přenáší jako prostý text, což je pro hackery snadné zachytit. Implicitně je přístup povolen pouze Administratorovi a dalším uživatelům by měl jít nastavit přímo v konfiguraci. To se ale nepovedlo, protože při přidělování oprávnění uživatelům prohlížet stránky SUITEVOYAGERU se předpokládá, že síť, ve které je zprovozněn, je doménového typu. Ale síť ČVUT není doménového typu, tudíž musel být zvolen jiný způsob a to pomocí služby **Component Services**<sup>41</sup>. Byly vytvořeny celkem tři speciální uživatel-

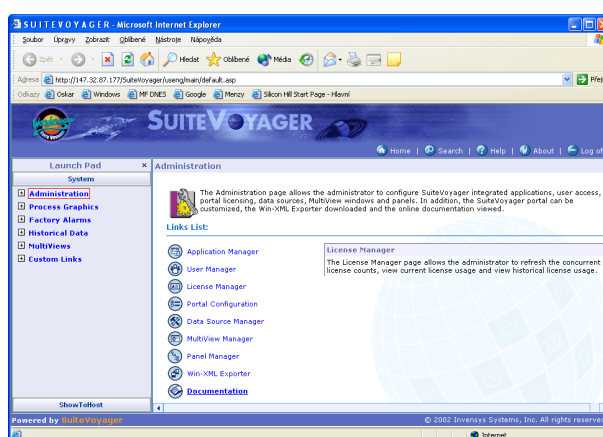
<sup>38</sup>Dynamic Data Exchange protokol vyvinutý společností Microsoft pro výměnu dat a instrukcí mezi aplikacemi v rámci prostředí Windows.

<sup>39</sup>Podrobněji je popsána v kapitole 4.

<sup>40</sup>Internet Information Services.

<sup>41</sup>Umožňuje konfigurovat a spravovat aplikace modelu COM+, jehož součástí je i SUITEVOYAGER.





Obrázek 2.25: Systémové prostředí SUITEVOYAGERU

ské účty<sup>42</sup> pro účel SUITEVOYAGERU. Těmto účtům není povolen přístup ke vzdálené ploše, protože se jejich údaje přenáší po síti nešifrovaně a tudíž by mohly být jednoduše zneužity.

Aplikaci vytvořenou v prostředí INTOUCH je nutné pomocí Win-XML Exportéru<sup>43</sup> nahrát do webového prostředí SUITEVOYAGERU. Aby byla zobrazovaná data SUITEVOYAGERem aktivní, je nutné, aby na serveru běžel MS SQL SERVER, ze kterého webové rozhraní SUITEVOYAGER čerpá data. Dále je nutné, aby byl na serveru aktivní prohlížeč WINDOWVIEWER a v něm běžela vizualizační aplikace. Stav připojení k datům z WINDOWVIEWERU je na webovém prohlížeči zobrazen ve stavové řádce. WINDOWVIEWER lze spouštět jako službu (nastavení v prostředí INTOUCH), ale pak ho nelze zároveň spustit jako aplikaci na stejném serveru.

Vizualizace dat pomocí SUITEVOYAGERU má tu výhodu, že ke sledování technologických dat postačuje webový prohlížeč, který je dnes součástí každého počítače. Nevýhodou je relativně dlouhá doba odezvy, a proto se spíše používá k vizualizaci dat na manažerské úrovni, kde není časová odezva tak důležitá.

### 2.3.3 WWW stránky

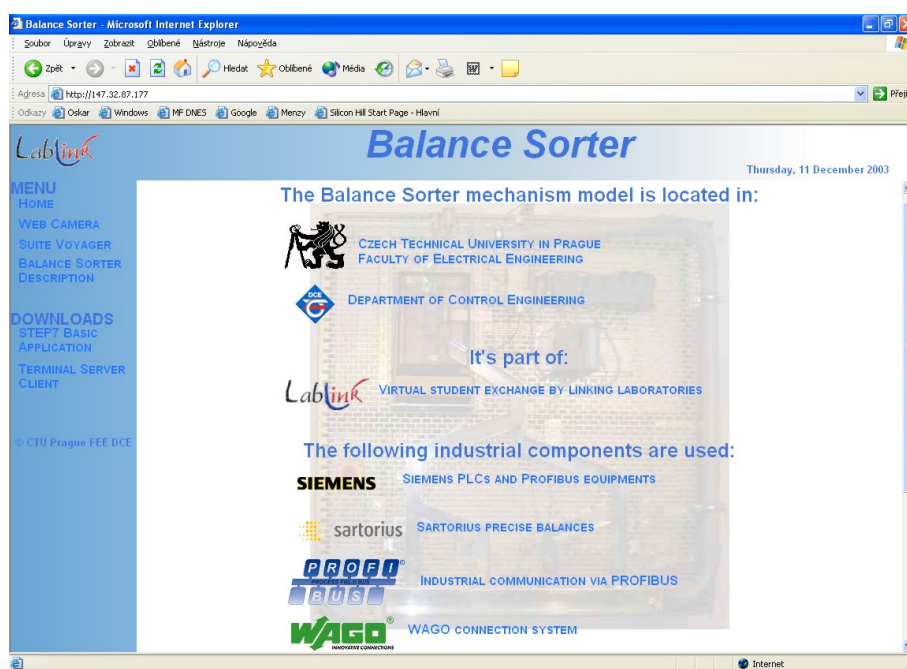
Jelikož součástí serveru je i webový server (IIS 5.0), tak byly vytvořeny a zprovozněny webové stránky<sup>44</sup> podávající informace o modelu třídícího mechanismu viz obrázek 2.26. Na těchto stránkách jsou veškeré potřebné informace k naprogramování třídícího algoritmu. Je zde také odkaz na SUITEVOYAGER a webovou kameru AXIS popsanou v následující kapitole 2.4. Jsou zde také uloženy TSE Client potřebný k přístupu ke vzdálené ploše a výchozí aplikace ve STEP7, která je odrazovým můstkem studentů k naprogramování třídícího mechanismu.

<sup>42</sup>Jeden pro účel administrace, druhý pro prohlížení aplikace s právem zápisu dat a třetí pouze pro sledování dat vizualizace bez možnosti zásahu do řízení.

<sup>43</sup>Aplikace, která se stáhne v prostředí SUITEVOYAGERU.

<sup>44</sup>Ve standardu dynamického HTML.





Obrázek 2.26: Úvodní stránka

## 2.4 Webová kamera AXIS 2100

Model třídícího mechanismu je sledován síťovou kamerou AXIS 2100 (obrázek 2.27). Kamera má vestavěný webserver, proto nepotřebuje žádný další nadřazený počítač na zpracování a distribuci snímků v síti. Po zadání IP adresy do prohlížeče se objeví výrobcem vytvořené webové stránky, které je možno modifikovat.

### 2.4.1 Připojení k síti Ethernet

Kameru lze připojit k síti buď pomocí modemu nebo k síti LAN kabelem UTP s konektory RJ45. V druhém případě lze nastavení IP adresy provést dvěma způsoby.

#### Nastavení IP adresy příkazem ARP

Nastavení příkazem ARP se provádí z příkazového řádku:

```
arp -s <IP adresa kamery> <Adresa Ethernet> <IP adresa mého počítače>
```

kde <IP adresa kamery> je požadovaná IP adresa kamery, <Adresa Ethernet> je fyzická MAC adresa kamery a je shodná se sériovým číslem uvedeným na výrobním štítku kamery. <IP adresa mého počítače> je adresa počítače, ze kterého je spuštěn příkaz ARP. V tomto případě tedy vypadá daný příkaz takto:

```
arp -s 147.32.87.178 00-40-8c-10-00-86 147.32.87.176
```



Obrázek 2.27: Webová kamera AXIS 2100

Poté kameru připojíme k síti kabelem UTP a zapneme napájení. Přibližně za 10-15 sekund se objeví kladná odpověď příkazu:

```
ping -t 147.32.87.178
```

### Nastavení IP adresy pomocí aplikace AXIS IP Installer

AXIS IP Installer je aplikace pro Windows 95/98/NT/2000 a lze ji získat na adrese <http://www.axis.com> nebo na CD dodávaném ke kameře. Po instalaci a spuštění se zobrazí dialogové okno. Po vypnutí a následném zapnutí napájení kamery se kamera restartuje a v dialogovém okně se objeví její sériové číslo, které je shodné s její MAC adresou. Zbývá již jen nastavit požadovanou IP adresu a zkontrolovat správnost nastavení pohledem na webové stránky kamery.

Pro zdařilé načtení webových stránek kamery je však nejprve nutné nastavit v prohlížeči zákaz používání proxy serveru pro adresu 147.32.87.178, protože kamera ještě nemá kromě IP adresy nastavené žádné síťové parametry, tedy ani IP adresu routeru. Teprve po nastavení zákazu proxy je možno zadáním URL `http://147.32.87.178` načíst požadované stránky z kamery. Nastavení dalších síťových parametrů se provádí již pouze na webových stránkách zadáním URL `http://147.32.87.178/admin`.

Na této stránce je možno změnit již nastavenou IP adresu, nastavit masku pod-sítě, IP adresu routeru a primárního a sekundárního DNS serveru, zadat doménový název kamery, rychlost sítě a číslo HTTP portu. Po nastavení IP adresy routeru lze již v prohlížeči zrušit zákaz používání proxy serveru.

Další vlastnosti jako například rozlišení a orientaci snímku, vlastní design stránek, přístupová práva a mnoho dalších vlastností lze nastavovat po kliknutí na příslušné odkazy.

### 2.4.2 Frekvence posílání snímků

Frekvence posílání snímků po síti je závislá na:

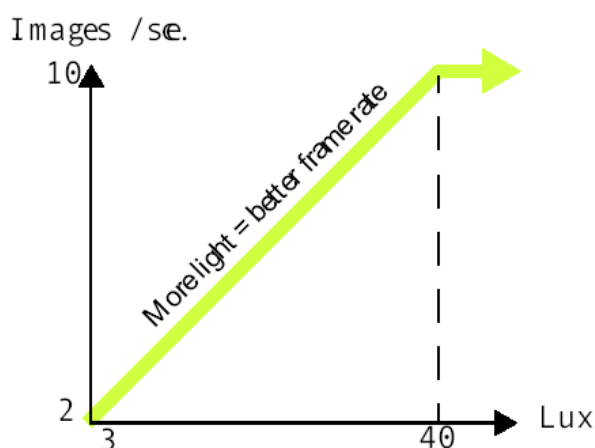
- Rychlosti počítače, který snímky z kamery čte
- Rozlišení posílaných obrázků
- Světelných podmínkách v místě instalace kamery
- Rychlosti sítě

Typické hodnoty datových objemů snímků popisuje obrázek 2.28. Závislost frekvence

Resolutio n	At lowest compression	At low compression	At medium compression	At high compresson
640 x 480	250 kb	20 kb	13 kb	8 kb
320 x 240	70 kb	8 kb	5 kb	3 kb

Obrázek 2.28: Datové objemy snímků v závislosti na rozlišení a kompresi

snímku na okolním osvětlení ukazuje obrázek 2.29.



Obrázek 2.29: Frekvence snímků v závislosti na okolním osvětlení

Pohled na třídící mechanismus webkamerou, v noci za umělého osvětlení, je na obrázku 2.30.



Obrázek 2.30: Pohled na třídící mechanismus webkamerou AXIS 2100

# Kapitola 3

## Softwarové řešení řídicího systému

Řídicí systém je rozdělen do dvou hlavních celků viz kapitola 2.2. PLC S7-200 plní roli supervizora a ochraňuje solenoidy třídícího mechanismu před přetížením resp. jejich zničením. Toto PLC bylo naprogramováno v laboratoři z lokálního PC a není možné vzdáleně měnit jeho program. Přijímá signály z PLC S7-300 a dle jejich úrovně zapne resp. vypne daný solenoid nebo jiný akční člen.

PLC S7-300 slouží k řízení třídícího mechanismu dle naprogramovaného třídícího algoritmu. Dále pomocí Gateway DP/RS-232C získává informaci z váhy Sartorius o hmotnosti váženého míčku. Toto PLC je možné programovat vzdáleně z prostředí STEP7, které je nainstalováno na Terminal Severu.

### 3.1 PLC S7-200

Je naprogramováno ve vývojovém prostředí STEP7-MICRO/WIN 32<sup>1</sup>, vyvinutého speciálně pro tuto řadu programovatelných automatů. Lze si vybrat jeden ze tří způsobů programování:

- **STL** *Statement List* – programuje se na úrovni instrukcí, je nejbližší assembleru.
- **LAD** *Ladder* – jedná se o žebříčkové grafické prostředí, kde jsou znázorněny cesty toku logického signálu.
- **FBD** *Function Block Diagram* – programování na úrovni funkčních bloků, kde každý blok reprezentuje určitou instrukci s patřičnými vstupy a výstupy. Tyto bloky jsou mezi sebou propojeny logickými cestami.

K programování je využito STL, protože veškerý program lze zapsat pomocí instrukcí a není nutné podstoupit zdoluhavou klikací proceduru s myší jako v případě LAD a FBD. Strukturu programu v MICROWINu lze rozdělit do následujících částí<sup>2</sup>:

---

<sup>1</sup>Dále jen zkráceně MICROWIN.

<sup>2</sup>Nazývaných též *Program Organizational Unit-POU*.

**Hlavní část** – označována jako Main nebo OB1<sup>3</sup>. Zde se program cyklicky opakuje. Vykonávání programu od první po poslední instrukci se nazývá programový cyklus neboli scan cyklus.

**Subroutines** – jedná se o podprogramy, které přeruší běh programu odkud byly zavolány a po jejich ukončení pokračuje původní běh programu. Jsou označovány jako SBR + pořadové číslo a jejich počet je omezen použitým CPU, v našem případě 64. Tyto podprogramy lze volat i se vstupními parametry jejichž počet může být maximálně 16, tím částečně zastupují funkce.

**Interrupt subroutines** – podprogramy přerušení, které jsou zavolány v případě nějaké události. Podprogramy přerušení se neprovádějí v programovém cyklu, ale provedou se ve chvíli, kdy je splněna podmínka pro přerušení (což může nastat v libovolném okamžiku programového cyklu).

### Rozdělení paměti

Paměť je rozdělena do oblastí podle funkce, pro kterou je určena. Oblast binárních vstupů a výstupů je označena jako I resp. Q a jejich rozsah je max. po 16-ti bytech. Oblasti M (*Internal memory bits*) a V (*Variable memory*) reprezentují oblast paměti s bitovým přístupem, kterou lze použít k libovolným výpočtům a uchování mezivýsledků. Rozsah oblasti M je 32 bytů a V 2047 bytů v případě CPU 222. Rozdíl mezi M a V je v tom, že při výpadku elektrického napájení CPU lze<sup>4</sup> trvale uchovat stav prvních 14-ti bytů oblasti M v paměti EEPROM. Další oblasti jsou T (časovače) a C (čítače), jejichž rozsahy jsou uvedeny v tabulce 2.5. Oblasti AI (analogové vstupy), AQ (analogové výstupy) a HC (vysokorychlostní čítače) nejsou v této práci použity.

Další významnou oblastí paměti je L, která reprezentuje lokální paměť v rámci bloku OB1 nebo podprogramů. Rozsah je 64 bytů. Je důležité si uvědomit, že oblast L reprezentuje dočasnou paměť, která je aktivní pouze po dobu vykonávání podprogramu nebo hlavního programu (záleží kde je využita) a nelze zaručit, jaký bude mít počáteční<sup>5</sup> obsah! Například pro všechny podprogramy je obvykle použita jedna společná oblast paměti L, tudíž při volání podprogramu obsahuje data uložená v předchozím podprogramu.

Poslední důležitou oblastí je paměť SM nebo-li *Special Memory*. SM bity slouží pro výměnu informací mezi CPU a programem. Tyto bity se používají pro nastavení a řízení některých speciálních funkcí CPU S7-200 jako jsou: bit prvního programového cyklu nebo bit stavu matematických nebo provozních instrukcí. Rozsah paměti je 300 bytů. Přehled nejdůležitějších částí SM, které jsou použity při programování, je v tabulce 3.1.

Součástí programu může být také datový blok, který umožňuje inicializaci paměťové oblasti V podle kritérií zvolených programátorem.

Celý program se skládá z pěti podprogramů:

---

<sup>3</sup>Organization Block 1.

<sup>4</sup>Pokud jsou MB0 až MB13 nastaveny jako remanentní, jsou uloženy trvale v EEPROM.

<sup>5</sup>Tím je myšlen obsah na počátku programového cyklu OB1 nebo při zavolání podprogramu.

SM	Popis
SM0.0	Tento bit má vždy hodnotu 1.
SM0.1	Tento bit je v log. 1 v prvním programovém cyklu. Jedním z použití je volání inicializačního podprogramu.
SMW22	Doba posledního programového cyklu v milisekundách.
SMB222 až SMB249	Informace specifické danému modulu.

Tabulka 3.1: Vlastnosti některých částí Special Memory

- **SBR1–*CompleteRestart*** - je volán jen při startu (zajištěno bitem SM0.1) a provádí potřebnou inicializaci programu.
- **SBR2–*TimerTON*** - zajišťuje funkci časovače a je volán v SBR3 až SBR5.
- **SBR3–*SingleControl*** - zajišťuje ochranu akčních členů.
- **SBR4–*MultipleControl*** - zajišťuje ochranu akčních členů.
- **SBR5–*LightControl*** - zajišťuje zapnutí a vypnutí osvětlení třídícího mechanismu.

SBR3 až SBR5 jsou volány cyklicky v OB1.

### 3.1.1 Ochrana akčních členů

Vzhledem k tomu, že má být zajištěna stejná ochrana pro několik akčních členů, je nejvhodnější program napsat jako funkci (v našem případě se jedná o podprogramy), která se bude volat několikrát s jinými parametry. Z hlediska ochrany jsou akční členy rozděleny do dvou skupin:

1. **Akční členy ovládané úrovní** - do této skupiny jsou zařazeny mechanické závory (obrázky 2.4 a 2.5) a fukar.
2. **Akční členy ovládané náběžnou hranou** - do této skupiny patří zarážky<sup>6</sup> (obrázek 2.7) a dva velké solenoidy starající se o pohyb uzávěru spodního zásobníku (obrázek 2.3).

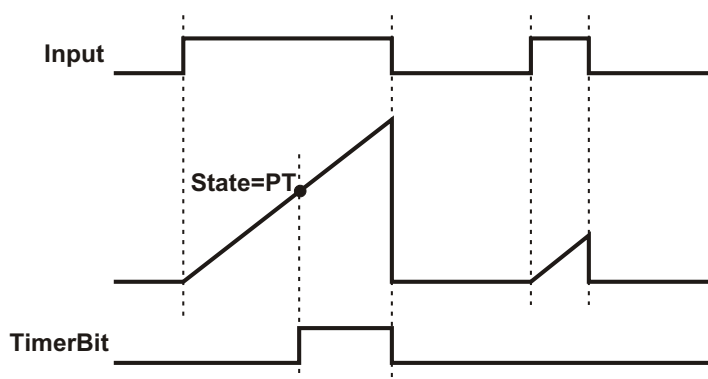
Byly naprogramovány dva podprogramy (SBR3 a SBR4) zajišťující ochranu výše zmíněných dvou skupin akčních členů, jejichž funkce je popsána v následujících podkapitolách. Jelikož součástí podprogramu musí být i časovač pro hlídání doby sepnutí akčního členu, tak je nutné paměťovou oblast časovače předávat jako parametr. Ale standardní časovače využívají paměťové oblasti T, kterou nelze předávat jako parametr podprogramu, tudíž byl naprogramován vlastní časovač jako podprogram (jehož parametry mohou být paměťová oblast M nebo V).

<sup>6</sup> Jsou tři a o jejich pohyb se stará celkem šest solenoidů.



## SBR2 - Časovač TON

Jedná se o standardní *On Delay Timer* (bez použití reset bitu), jehož funkce je zobrazena na obrázku 3.1. Náběžná hrana **Input** bitu zajistí spuštění časovače a ten po dosažení žádaného času (proměnná **PT-32** bit integer) nastaví **TimerBit** do vysoké logické úrovně. V případě nízké logické úrovně Input bitu dojde k vynulování časovače.



Obrázek 3.1: Funkce časovače

Odměrování času se děje pomocí **SMW22**, ve kterém je vždy informace o délce trvání<sup>7</sup> posledního scan cyklu programu v ms. Tudíž rozlišení časovače je závislé na délce scan cyklu. Maximální čas, kterého může časovač dosáhnout je  $2^{31}$  ms ( $2^{31}$ , protože se jedná o double integer se znaménkem), což je 2147483 vteřin nebo 596 hodin resp. 24 dní.

Pro funkci časovače je nutné detekovat náběžnou hranu. K tomuto účelu slouží v prostředí MICROWIN instrukce **EU** bez parametru. Její použití v rozsáhlejšímu programu se ukázalo jako nemožné, neboť byla důvodem nečekaného chování programu. Tato instrukce nejspíš používá pro detekci hrany omezený paměťový buffer, a když dojde k jeho překročení (přetečení), detekce náběžných hran pomocí **EU** není spolehlivá<sup>8</sup>! Proto k detekci náběžné hrany je použit pomocný bit, který uchovává hodnotu signálu v daném scan cyklu. V následujícím scan cyklu je jeho hodnota porovnávána s aktuálním stavem signálu a při různých logických úrovních pomocného bitu a signálu je detekována změna úrovně (tedy i náběžná hrana).

Syntaxe zápisu volání podprogramu časovače je

**CALL** TimerTON, Input,PT,State,TimerBit,EdgeBit,

kde **TimerTON** je název podprogramu časovače, **Input** je vstupní bit spouštějící časovač, **PT** je hodnota v ms, po které se nastaví výstup (**TimerBit**) do vysoké logické úrovně. **State** obsahuje aktuální dobu v ms, která uplynula od spuštění časovače. **EdgeBit** je pomocný bit pro detekci náběžné hrany.

<sup>7</sup>Pro správnou funkci časovače je nutné, aby scan cyklus byl delší než 1 ms, což je v našem případě s rezervou splněno.

<sup>8</sup>To byl i můj případ.

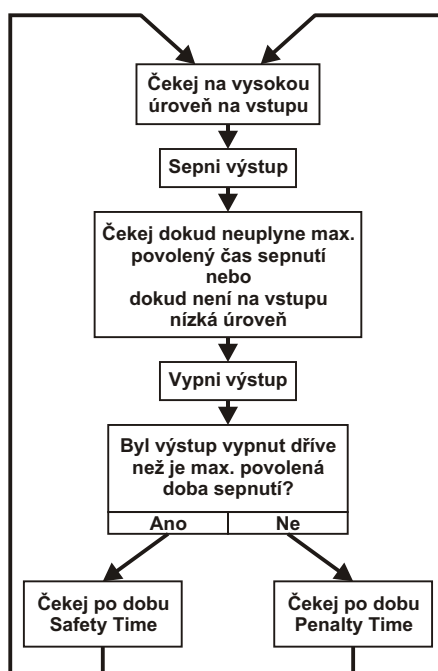
### SBR3 - Ochrana akčních členů ovládaných úrovní

Vývojový diagram je na obrázku 3.2. Vstupním signálem lze sepnout resp. vypnout akční člen. V případě překročení povolené doby sepnutí dojde k vypnutí akčního členu nezávisle na vstupním signálu. Po vypnutí je nastaven čas, po který nebude možné akční člen zapnout. Rozlišuje se mezi dvěma časy v závislosti na tom, jak došlo k vypnutí:

**Safety Time** – doba, po kterou nepůjde sepnout akční člen v případě, že byl vypnut včas, tedy nebyla překročena maximální povolená doba sepnutí.

**Penalty Time** – doba, po kterou nepůjde sepnout akční člen v případě, že nebyl vypnut včas, tedy byla překročena maximální povolená doba sepnutí.

Penalty Time je oproti Safety Time několikanásobně vyšší<sup>9</sup> a je to časové penále (trest) za nedodržení maximální povolené doby sepnutí akčního členu. Safety Time je bezpečnostní doba nečinnosti akčního členu po jeho aktivním stavu. Pokud na vstup přijde vysoká úroveň vstupního signálu (pokyn k sepnutí akčního členu) v okamžiku vykonávání Penalty resp. Safety Time, tak dojde k vynulování příslušného Time a ten poběží znovu od začátku!



Obrázek 3.2: Vývojový diagram SBR3

<sup>9</sup>Například u mechanických závor je max. povolená doba sepnutí 2 s, Safety Time 2 s a Penalty Time 20 s.

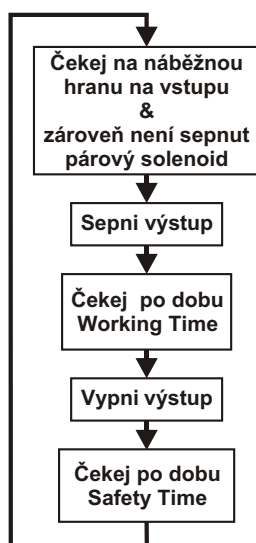
Syntaxe zápisu volání SBR3 je

```
CALL   SingleControl, ControlInput, MaxOnTime, SafetyTime,
        PenaltyTime, ControledOutput, SafetyState,
        PenaltyState, MemForTimer1, BitForTimer1,
        BitForTimer2, MemForEdgeMonitor,
        MemForEdgeMonitor, MemForEdgeMonitor,
```

kde `SingleControl` je název podprogramu SBR3, `ControlInput` je vstupní bit (signál) pro sepnutí akčního členu. `MaxOnTime` je max. povolená doba sepnutí akčního členu. `ControledOutput` je hlídáný výstup, pomocí kterého je sepnut akční člen. Ostatní parametry jsou paměťové oblasti pro použité časovače (SBR2) volané uvnitř SBR3 a pomocné bity sloužící k detekci náběžné resp. sestupné hrany.

### SBR4 - Ochrana akčních členů ovládaných náběžnou hranou

Do této skupiny jsou zařazeny solenoidy umístěné po párech proti sobě zajišťující posuvný pohyb (např. zarážky) v obou směrech. Proto je nutné zajistit, aby bylo povoleno zapnout solenoid jen tehdy, pokud je jeho "párový solenoid" vypnut, aby se solenoidy mezi sebou nepřetahovaly a nedocházelo k jejich zbytečnému zatížení.



Obrázek 3.3: Vývojový diagram SBR4

Vývojový diagram je na obrázku 3.3. Sepnutí solenoidu nastane po náběžné hraně vstupního signálu a za podmínky, že vypršel jeho Safety Time a zároveň není zapnut jeho párový solenoid. Po vykonání pohybu do žádané polohy (solenoid je po dobu Working Time v aktivním stavu, aby se mohl pohyb vykonat) je automaticky vypnut a pak ho po dobu Safety Time nelze sepnout. Penalty Time zde není aplikován, protože doba aktivního stavu solenoidu je pevně stanovena. Pokud na vstup přijde náběžná

hrana vstupního signálu (pokyn k sepnutí solenoidu) v okamžiku vykonávání Safety Time, tak dojde k jeho vynulování a poběží znovu od začátku!

Syntaxe zápisu SBR4 (pro dva solenoidy v páru se musí volat dvakrát s příslušnými parametry) je

```
CALL   MultipleControl, ControlInput, WorkingTime, SafetyTime,
        Output1, Output2, SafetyState, MemForTimer1,
        MemForTimer2, BitForTimer1, BitForTimer2,
        WorkingState1, WorkingState2,
        MemForEdgeMonitor, MemForEdgeMonitor,
        MemForEdgeMonitor, MemForEdgeMonitor,
```

kde `MultipleControl` je název podprogramu, `ControlInput` je řídicí vstup pro sepnutí akčního členu. `WorkingTime` udává dobu, po kterou bude solenoid sepnut. `Output1` je výstup, pomocí kterého je solenoid sepnut a `Output2` je výstup párového solenoidu. Ostatní parametry jsou paměťové oblasti pro použité časovače (SBR2) volané uvnitř SBR4 a pomocné bity sloužící k detekci náběžné resp. sestupné hrany.

Původně byla snaha vytvořit pro oba párové solenoidy podprogram tak, aby se volal jen jednou, ale to nebylo možné z důvodů omezeného počtu vstupních parametrů podprogramu.

**Pozn.:** K rozsvícení osvětlení je použit podprogram SBR5, který je jednodušší variantou SBR4 (není použit Safety Time a závislost na párovém solenoidu). Osvětlení se rozsvítí náběžnou hranou vstupního signálu na 55 minut a potom je zhasnuto. Pak ho lze znovu náběžnou hranou zapnout. Tímto postupem je zvýšena životnost osvětlení, která se snižuje častým rozsvícením a zhasínáním. Veškeré nastavené časy (SafetyTime, PenaltyTime, MaxOnTime, WorkingTime) u všech akčních členů jsou v tabulce 3.2 na straně 49.

### 3.1.2 Komunikace s modulem EM 277

Aby mohlo CPU 222 posílat a přijímat data na sběrnici Profibus, je nutné zajistit komunikaci s externím inteligentním modulem EM 277. Komunikace je zajištěna pomocí SM<sup>10</sup> (oblasti Special Memory). Postup komunikace je shrnut v následujících krocích:

1. Kontrola, zda je EM 277 v režimu výměny dat porovnáním bytu SMB224 s hodnotou 2 (jsou použity jen první dva bity - mohou nastat čtyři režimy, podrobněji viz [11]).
2. V případě splnění první podmínky lze provádět správnou výměnu dat pomocí SMW226, SMB228 a SMB229.

<sup>10</sup>Konkrétní hodnoty SM jsou závislé na pozici inteligentního modulu za CPU.

Word SMW226 obsahuje adresu výstupního zásobníku v paměti V jako posunutí od VB0. V tomto zásobníku jsou za sebou umístěny nejprve výstupní data a za nimi vstupní data. SMB228 obsahuje počet bytů výstupních dat a SMB229 obsahuje počet bytů vstupních dat. Počet bytů vstupních resp. výstupních dat je ovlivněno hardwarovou konfigurací stanice DP-Master, v našem případě CPU 315-2 DP.

Původně byla tato komunikace zapsána jako kód podprogramu, ale při vysílání více jak jednoho bytu výstupních dat se poslal vždy z nepochopitelných důvodů jen první byte. Proto je tento kód přímo součástí OB1, kde funguje již bez problémů.

## 3.2 PLC S7-300

Tato řada programovatelných automatů se programuje v prostředí STEP7. Jsou umožněny stejné způsoby programování (STL, LAD, FBD) jako u MICROWINu, přičemž bylo zvoleno opět STL. Strukturu programu lze rozdělit do následujících částí:

- **Organization Blok (OBx)** – je jich zhruba kolem stovky a jsou určeny k různé činnosti. OB1 je hlavní cyklus programu, OB10 až OB47 se vykonávají v případě nějakého přerušení, OB100 se vykoná pouze v prvním programovém cyklu atd.
- **Data Block (DBx)** – slouží k uchování dat. Jejich velikost a počet je omezen.
- **Function (FCx)** – jedná se normální funkce jejichž součástí je i dočasná lokální paměť (viz L paměť u S7-200).
- **Function Block (FBx)** – je obdobou FC, ale navíc může uchovávat stavy vnitřních proměnných i do dalších programových cyklů ⇒ nutnou součástí je DB.

Přesný počet DB, FB a FC u CPU 315-2 DP je v tabulce 2.4. Paměť je rozdělena do oblastí I, Q, M, T, C, L obdobně jako u S7-PLC. Navíc je DB a P, což je oblast vyhrazená perifériím.

### 3.2.1 Hardwarová konfigurace

Před samotným programováním je nutné provést hardwarovou konfiguraci celého systému. Nejprve se volí typ CPU a k němu přidružené zařízení (I/O periferie a inteligentní moduly). Dále je případně nutné nakonfigurovat parametry komunikace po sběrnici Profibus jako přenosovou rychlost, adresy připojených stanic atd. U modulu gateway DP-RS232C je nutné nastavit velikost konzistence dat, což je velikost I/O oblasti, pomocí které jsou zobrazeny data z váhy do CPU. Konkrétně v našem případě je nejvýhodnější nastavit konzistenci dat stejně velkou jako je délka protokolu vysílaného z váhy (15 bytů).

Po úspěšném nakonfigurování hardwaru je možné začít se samotným programováním. Struktura programu se dá podle funkce rozdělit do tří částí:

- Čtení posloupnosti znaků (protokolu) z váhy a převést je na údaj reprezentující hmotnost. To zajišťuje funkce FC50–*ReadBalance*.

- Zprostředkování proměnných a informací vizualizačnímu softwaru INTOUCH. To zajišťuje funkce FC51–*InTouch*.
- Samotný třídící algoritmus a obslužná rutina ovládání třídícího mechanismu. To zajišťuje kód napsaný v OB1 v kombinaci s funkcemi FC1–*IdentifyBallColour*, FB1–*BatchMeter*, FB2–*SetWay* a k nim přidružené DB1 a DB2.

Význam funkce FC51 je popsán v kapitole 4.

### 3.2.2 Převod dat z váhy na údaj o hmotnosti – FC50

Váha posílá údaj o hmotnosti pomocí protokolu popsaného v kapitole 2.1.5. Tento protokol se skládá z patnácti bytů, přičemž údaj o hmotnosti poskytují třetí až desátý znak v předem neznámém tvaru.

Funkce FC50 má jako vstupní parametr integer číslo reprezentující počáteční adresu<sup>11</sup> oblasti IB, na které se nachází data z modulu gateway DP/RS232C. Tímto je zajištěna nezávislost na hardwarové konfiguraci. Výstupem je reálné číslo reprezentující údaj o hmotnosti v gramech a bit signalizující vysokou logickou úrovní stabilní údaj. Následující postup naznačuje jak funguje FC50 při čtení protokolu.

1. Transformace vstupního integer údaje o adrese IB na pointer a načtení celého protokolu do lokální paměti, přičemž se u každého bytu extra nuluje nejvýznamnější bit, protože váha posílá znaky o délce 7 bitů. Kdyby se nejvýznamnější bit nevyvnuoval, tak není vůbec zaručeno jakou bude mít hodnotu.
2. Nalezení pozice znaku LF a zkopírování znaků za ním (pokud za ním jsou) na začátek lokálního pole. Na zbylé pozice lokálního pole jsou zapsány všechny znaky umístěné před znakem LF.
3. Zjištění znaménka v prvním bytu lokálního pole.
4. Zjištění pozice tečky v lokálním poli, která se může nacházet na čtvrté až desáté pozici.
5. Přečtení všech znaků reprezentujících číslici a vynásobení těchto číslic příslušnou dekadickou váhou podle pozice vůči tečce. Výsledkem této operace je reprezentace každé číslice již jako reálné číslo.
6. Sečtení všech číslic (reprezentovaných jako reálné číslo) z předchozího bodu a vynásobení výsledku příslušným znaménkem z bodu 3.

Výše popsany postup předpokládá, že je v HW konfiguraci u modulu gateway DP/RS232C nastavena konzistence dat v délce protokolu (15 bytů), pak je možné funkci FC50 používat bez problémů. V bodě 2 se provádí transformace vstupních znaků z gateway DP/RS232C do tvaru na obrázku 2.18. Může se stát, že znak LF není na poslední pozici ve vstupním bufferu modulu gateway DP/RS232C, ale na libovolné

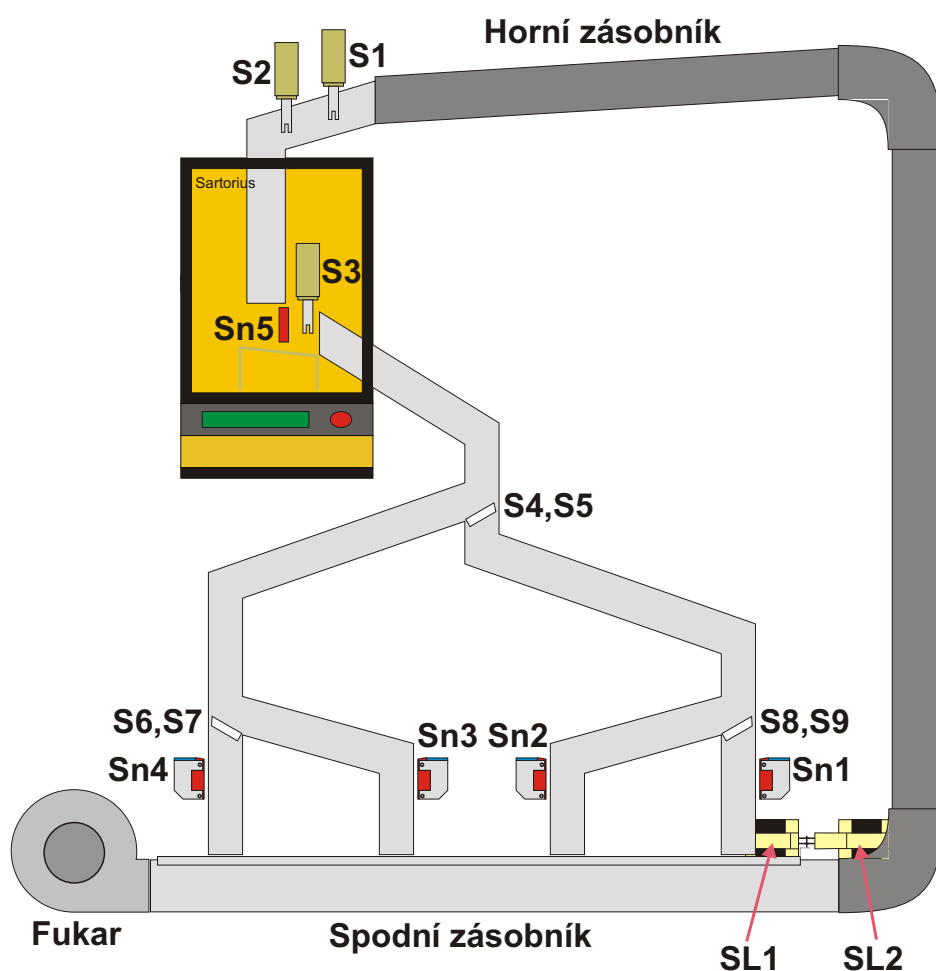
<sup>11</sup>Tato adresa se nastavuje v HW konfiguraci.

z patnácti pozic a ostatní znaky jsou cyklicky posunuty. Tento případ nastává při špatné synchronizaci protokolu mezi přijímačem a vysílačem (např. když v okamžiku nahrávání HW konfigurace do CPU 315-2 DP váha vyšle po RS-232 údaj o hmotnosti).

Informace o stabilním údaji je zjišťována z dvanácté pozice protokolu porovnáním s hodnotou 67 Hex.

### 3.2.3 Třídící algoritmus

Třídící algoritmus je napsán tak, aby ho bylo možné jednoduše povelovat z vizualizace (viz následující kapitola 4). Lze ho spustit, pozastavit a úplně přerušit uvedením do výchozího stavu. Použité značení akčních členů je na obrázku 3.4, kde **S1–S3** jsou mechanické závory, **S4–S9** zajišťují pohyb zarážek, **SL1** a **SL2** pohybují s uzávěrem spodního zásobníku, **Sn1–Sn5** jsou snímače a **Fukar** je označován též jako Blower.



Obrázek 3.4: Použité značení akčních členů

Při práci s akčními členy je nutné počítat s maximální dobou sepnutí (viz tabulka 3.2, kde Tare je nulování váhy) a také se Safety Time. V případě spínání akčních členů



reagujících na náběžnou hranu nestačí u CPU 315-2 DP v jednom scan cyklu nastavit výstup do vysoké logické úrovně a v následujícím scan cyklu do nízké, protože scan cyklus CPU 222 je zhruba 30 ms a CPU 315-2 DP 10 ms. Pro spolehlivé sepnutí akčního členu je nutné nastavit výstup do vysoké logické úrovně aspoň po dobu dvou scan cyklů CPU 222! S přehledem postačuje 100 ms.

Akční člen	MaxOn Time [s]	Safety Time [s]	Penalty Time [s]	Working Time [s]
S1–S3	2	2	20	-
S4–S9	-	3	-	1
SL1 a SL2	-	8	-	1.5
Fukar	8	20	60	-
Tare	2	15	60	-
Osvětlení	-	-	-	3300

Tabulka 3.2: Přehled všech časů u akčních členů hlídaných CPU 222

Výchozím stavem samotného algoritmu třídění je umístění míčků v horním zásobníku (obrázek 2.4). Do tohoto stavu může míčky dovést automaticky proces Inicializace<sup>12</sup>, jehož vývojový diagram je na obrázku 3.5. Nejprve je otevřen uzávěr spodního zásobníku (solenoid SL2) a zároveň mechanická závora u váhy (S3), která je otevřena pouze na 1.8 vteřiny, aby nedošlo k překročení max. povoleného času sepnutí. Tímto je zajištěno uvolnění vážícího mechanismu váhy a propadnutí míčků do spodního zásobníku. Potom dojde k uzavření spodního zásobníku (SL1) a zapnutí fukaru na 4 vteřiny, což s přehledem postačuje na vyfouknutí všech míčků do horního zásobníku. Po vypnutí fukaru a následně po uplynutí třiceti<sup>13</sup> vteřin proces inicializace končí.

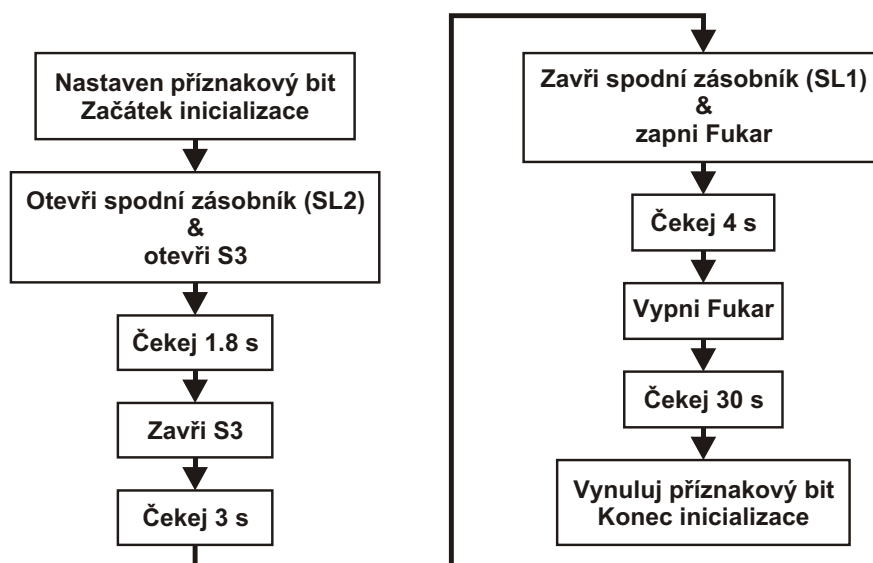
Po vyfouknutí míčků do horního zásobníku je nutné zajistit vhodné posílání (dávkování) míčků na vážící mechanismus. To zajišťuje funkční blok FB1, jehož funkci znázorňuje vývojový diagram na obrázku 3.6. Nejprve jsou sepnuty solenoidy S1 a S3, což způsobí posun míčku z horního zásobníku před solenoid S2. S3 způsobí opuštění míčku z vážícího mechanismu. Po uzavření S3 a S1 je otevřen S2, přičemž míček propadne na vážící mechanismus, kde je zvážen.

Po uplynutí doby nutné ke zvážení míčku (v našem případě 3 vteřiny) dojde k vyhodnocení, o jaký míček (barvu) se jedná (funkce FC1). Potom se nastaví zarážky (FB2) podle barvy (hmotnosti) míčku tak, aby míček spadl do příslušné šachty. Celý proces třídícího algoritmu je na obrázku 3.7.

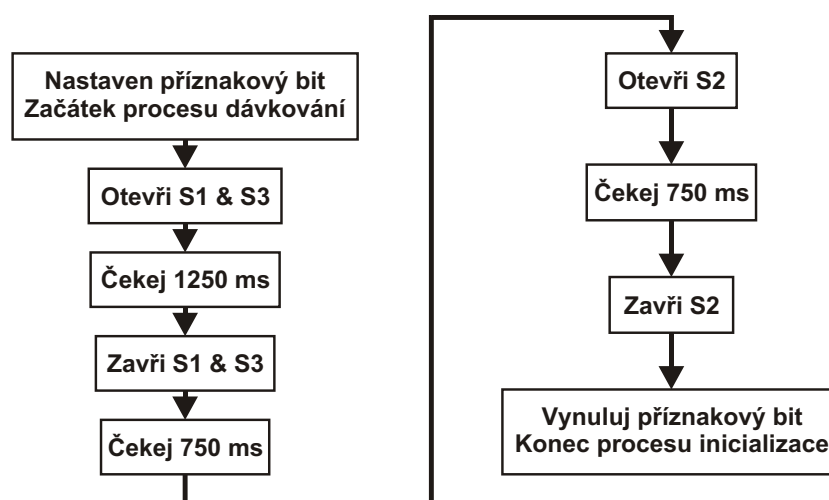
Funkce FC1 je volána cyklicky v OB1, tudíž vyhodnocuje barvu míčku nepřetržitě. Výstupem FC1 je barva míčku v podobě integer proměnné, která může nabývat hodnot 0 až 4 podle barvy míčku. O jakou barvu míčku se jedná je rozhodnuto pomocí dvou

<sup>12</sup>Nebo samozřejmě ručně povelováním příslušných akčních členů.

<sup>13</sup>Fukar má po vypnutí poměrně dlouhou setrvačnost než z něj přestane proudit vzduch, který působí na vážící mechanismus, a tím znemožňuje korektní vážení míčků. Po třiceti vteřinách je proud vzduchu z fukaru již minimální.



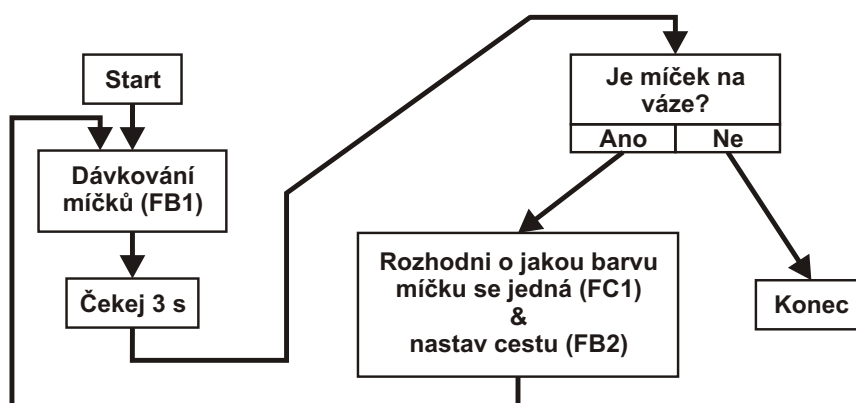
Obrázek 3.5: Vývojový diagram průběhu inicializace



Obrázek 3.6: Vývojový diagram dávkování míčků na váhu

mezí, mezi kterými by se měla nacházet hmotnost míčku. Tabulka 3.3 znázorňuje přiřazení čísla barvě míčku a hmotnostních mezí, podle kterých je rozhodnuto o barvě.

V případě, že se vážený míček (míčky) nenachází ani v jednom váhovém rozmezí z tabulky 3.3, pak výstupem FC1 je nula. Každé ze čtyř šachet je přidělena integer



Obrázek 3.7: Vývojový diagram třídícího algoritmu

Míček	Integer	Dolní mez [g]	Horní mez [g]
červený	1	2.0	2.50
žlutý	2	2.51	2.91
modrý	3	2.92	3.30
zelený	4	3.31	4.0

Tabulka 3.3: Přehled označení míčků a rozmezí jejich hmotností

hodnota<sup>14</sup> od jedné do čtyř podle toho, kam se má umístit daná barva míčku. Funkční blok FB2 porovná výstup FC1 s hodnotami příslušných šachet a nastaví zarážky tak, aby míček spadl do příslušné šachty. Jakmile jsou nastaveny zarážky podle váženého míčku, pak míček může opustit vážící mechanismus, což se provede zavoláním FB1.

Proces třídícího algoritmu se opakuje do té doby, dokud jsou signalizovány míčky na třídícím mechanismu (pomocí snímače umístěného u vážícího mechanismu nebo nenulovou hmotností).

**Pozn.:** Kompletní zdrojové kódy pro S7-200 a S7-300 jsou na příloženém CD.

<sup>14</sup>Lze ji změnit viz vizualizace v následující kapitole 4.

# Kapitola 4

## Vizualizace třídícího mechanismu

Jak již bylo napsáno v kapitole 2.3.2, k vizualizaci je použit INTOUCH. Jedná se o objektově orientovaný grafický generátor určený pro vytváření HMI (*Human Machine Interface*) rozhraní. Obsahuje předkonfigurované grafické objekty (Wizards).

Umožňuje vytvářet vlastní objekty a přiřadit jim animační vlastnosti, které jsou reakcí na změnu nějaké proměnné. Také je možné vytvářet vlastní skripty, které se dělí do šesti skupin:

1. **Aplikační** – tento skript běží na pozadí aplikace, vykonává se cyklicky, přičemž lze volit periodu opakování.
2. **Stisku klávesnice** – při stisku klávesy se vykoná definovaný skript.
3. **Logické podmínky** – v závislosti na výsledku logické podmínky se vykoná daný skript. Ten se může vykonat buď jednou<sup>1</sup> při splnění nebo nesplnění log. podmínky, anebo cyklicky.
4. **Změny dat**
5. **Okna** – každému aplikačnímu oknu může být přiřazen vlastní skript, který se vykonává, pokud je okno aktivní.
6. **Procedury**

Veškeré proměnné z PLC jsou do aplikace INTOUCH mapovány jen<sup>2</sup> pomocí paměťové oblasti M, konkrétně MB200 až MB225. Aby bylo možné přenášet i stavy vstupů a výstupů je použita funkce FC51. Tento způsob byl zvolen proto, aby byla dosažena nezávislost na hardwarové konfiguraci, tedy na přiřazení adres<sup>3</sup> vstupům a výstupům.

---

<sup>1</sup>Je to jako reakce na náběžnou nebo sestupnou hranu podmínky.

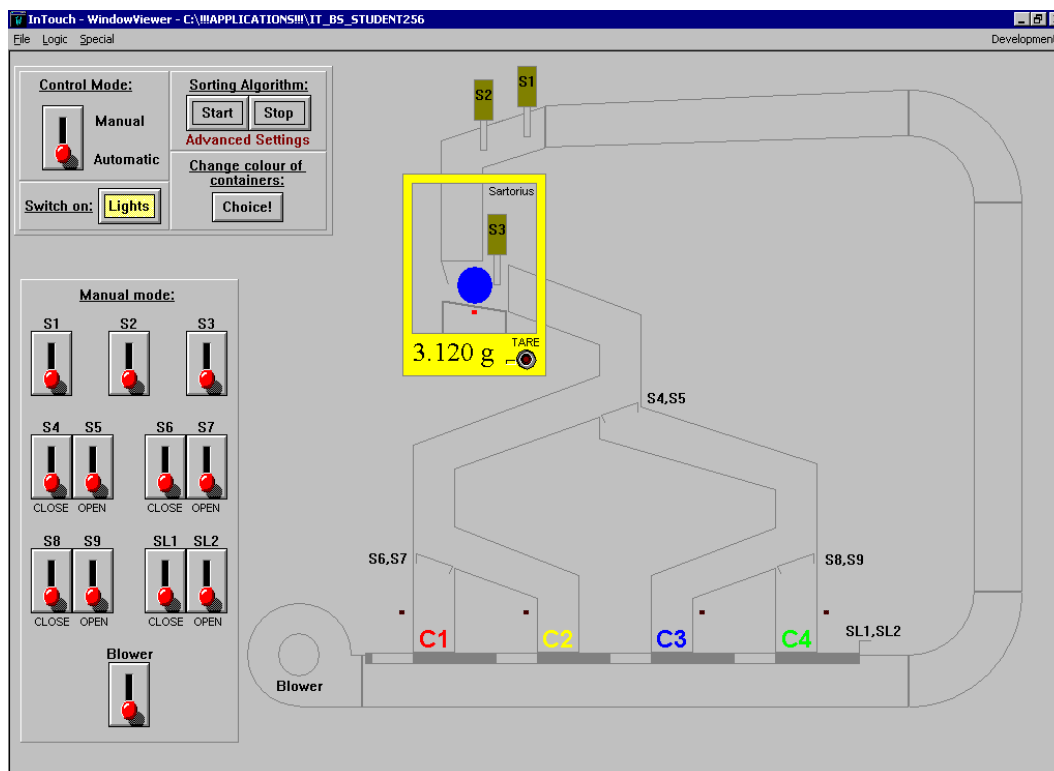
<sup>2</sup>Samozřejmě lze použít i jiné paměťové oblasti.

<sup>3</sup>Parametrem FC51 jsou adresy I/O.

## 4.1 Aplikace v InTouch

V prostředí WINDOWMAKER byla vytvořena aplikace znázorňující model třídícího mechanismu, obrázek 4.1. Hotová aplikace se spouští z prostředí WINDOWVIEWER, odkud je možné (ve spojení s FC51) ovládat třídící mechanismus. Lze volit mezi dvěma režimy ovládání:

1. **Manuální** – v tomto režimu lze ovládat veškeré akční členy pomocí přepínačů v levé části na obrázku 4.1.
2. **Automatický** – v tomto režimu je možné pomocí tlačítek **Start** a **Stop** ovládat průběh třídícího algoritmu. Není možné ovládat akční členy manuálně (tento zákaz paralelně zajišťují aplikační skript a FC51).

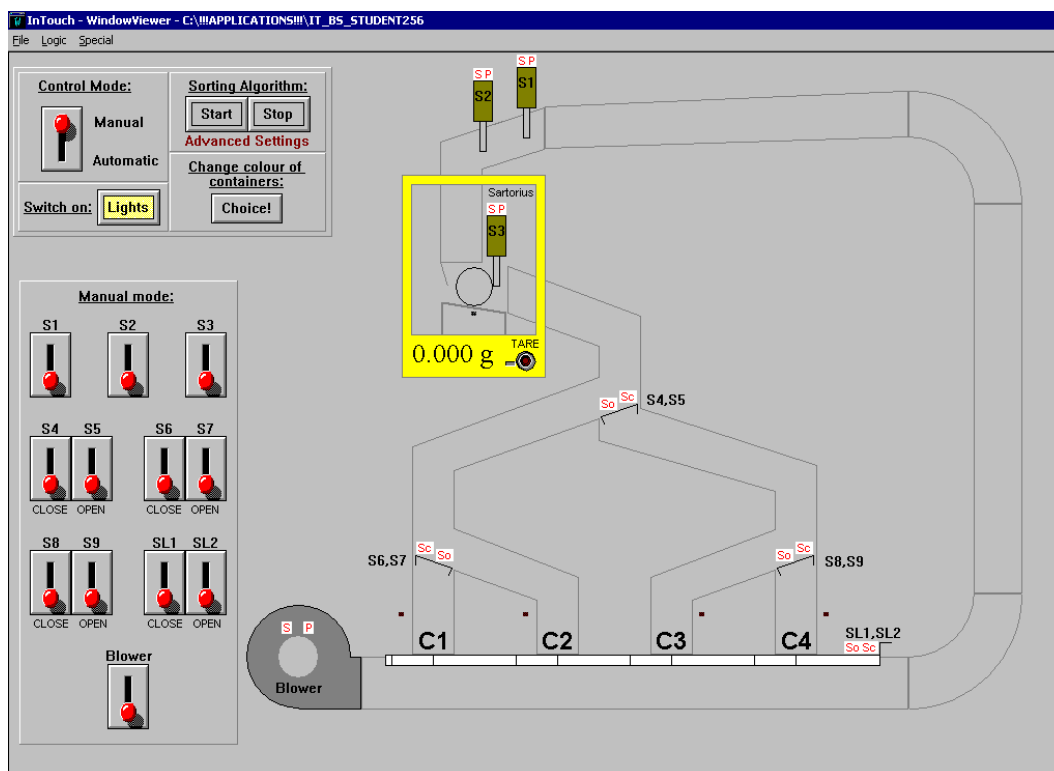


Obrázek 4.1: Vizualizační aplikace v INTOUCH

Z této aplikace lze ovládat veškeré akční členy<sup>4</sup> třídícího mechanismu a sledovat v jakém stavu se nacházejí. Osvětlení lze aktivovat pomocí tlačítka **Lights** nezávisle na volbě režimu (manuální nebo automatický). V případě výpadku spojení s PLC nebo pokud PLC přejde do STOP režimu, veškeré aktivní členy vizualizace začnou blikat, což je znázorněno<sup>5</sup> na obrázku 4.2.

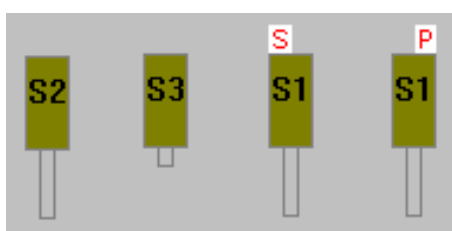
<sup>4</sup>V případě volby manuálního režimu.

<sup>5</sup>Blikání si lze představit jako periodické střídání obrázků 4.1 a 4.2 .



Obrázek 4.2: Vizualizační aplikace v INTOUCH – výpadek spojení s PLC

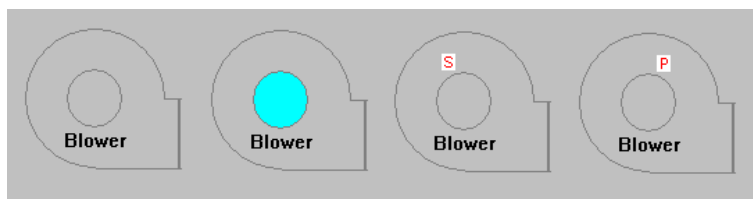
Mechanické závory jsou na obrázku 4.1 označeny jako S1 až S3. Veškeré možné aktivní stavy jsou znázorněny na obrázku 4.3, přičemž stavy (zleva) jsou: klidový stav, sepnutý stav, vypnutý stav a probíhá Safety Time, vypnutý stav a probíhá Penalty Time.



Obrázek 4.3: Aktivní stavy mechanických závor

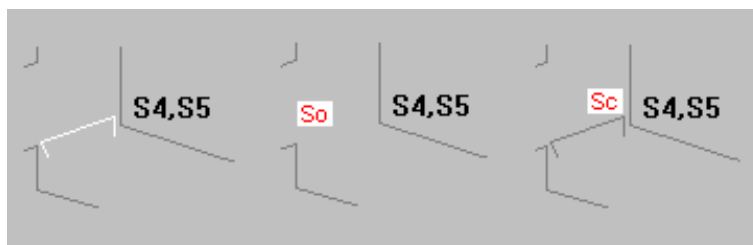
Fukar je na obrázku 4.1 označen jako Blower. Jeho všechny možné aktivní stavy jsou znázorněny na obrázku 4.4, přičemž význam stavů je shodný s mechanickými závory.

Stavy zářezek (S4 až S9) a uzávěru spodního zásobníku (SL1 a SL2) jsou na obrázcích 4.5 a 4.6. Bílá barva znázorňuje počáteční stav po zapnutí vizualizace, pro-

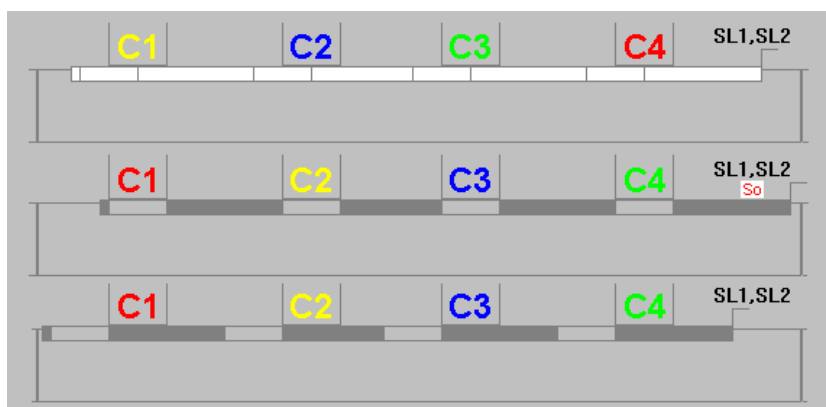


Obrázek 4.4: Aktivní stavy fukaru

tože není možné určit<sup>6</sup>, v jakém stavu se zarážky resp. uzávěr nachází. Dále jsou zobrazeny otevřená a zavřená zarážka resp. uzávěr. So resp. Sc znázorňují Safety Time po otevření resp. zavření zarážky nebo uzávěru.



Obrázek 4.5: Aktivní stavy zarážek



Obrázek 4.6: Aktivní stavy uzávěru spodního zásobníku

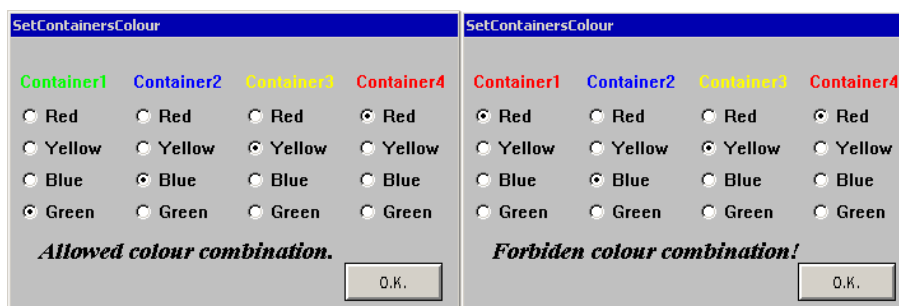
Na obrázku 4.1 je hmotnost míčku v gamech zobrazena přímo na váze. Pokud je hmotnost v jednom ze čtyř intervalů (viz tabulka 3.3), tak je míček na váze zobrazen přímo jeho barvou (na obrázku 4.1 to je modrý míček). Pomocí tlačítka umístěného

<sup>6</sup>Nejsou použity žádné koncové snímače. Zarážky resp. uzávěr se mohou nacházet v jednom ze dvou klidových stavů, narozdíl od mechanických závor resp. fukaru, kde je klidový stav jasně definován.



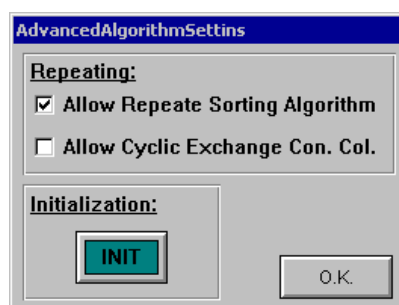
pod nápisem TARE lze váhu vynulovat, přičemž stabilní údaj hmotnosti je zobrazen obdélníčkem<sup>7</sup> umístěným vedle tlačítka TARE.

Do které šachty přijde jaká barva míčku je znázorněno pomocí C1 až C4 přímo barvou. Tyto barvy lze změnit vyvoláním nového okna (obrázek 4.7) pomocí tlačítka **Choice!**. V tomto okně lze každé šachtě přiřadit libovolnou barvu. Aby šla změna barvy potvrdit<sup>8</sup>, tlačítko **O.K.**, tak je nutné přiřadit každé šachtě jinou barvu! Změna barev, kterou lze resp. nelze potvrdit, je v tomto okně znázorněna pomocí informativního textu jako *Allowed colour combination.* resp. *Forbidden colour combination!*



Obrázek 4.7: Změna nastavení barev u šachet - vlevo povolená a vpravo zakázaná kombinace barev

Po stisknutí textu **Advanced Settings** (umístěného pod tlačítka **Start** a **Stop**) se otevře okno viz obrázek 4.8. V tomto okně lze zvolit periodické opakování třídícího algoritmu a automatickou změnu barev u šachet po každém roztřídění míčků. Dále zde lze spustit proces inicializace, který uvede míčky do výchozí polohy.



Obrázek 4.8: Pokročilé nastavení třídícího algoritmu

## 4.2 Zobrazení aplikace pomocí SuiteVoyager

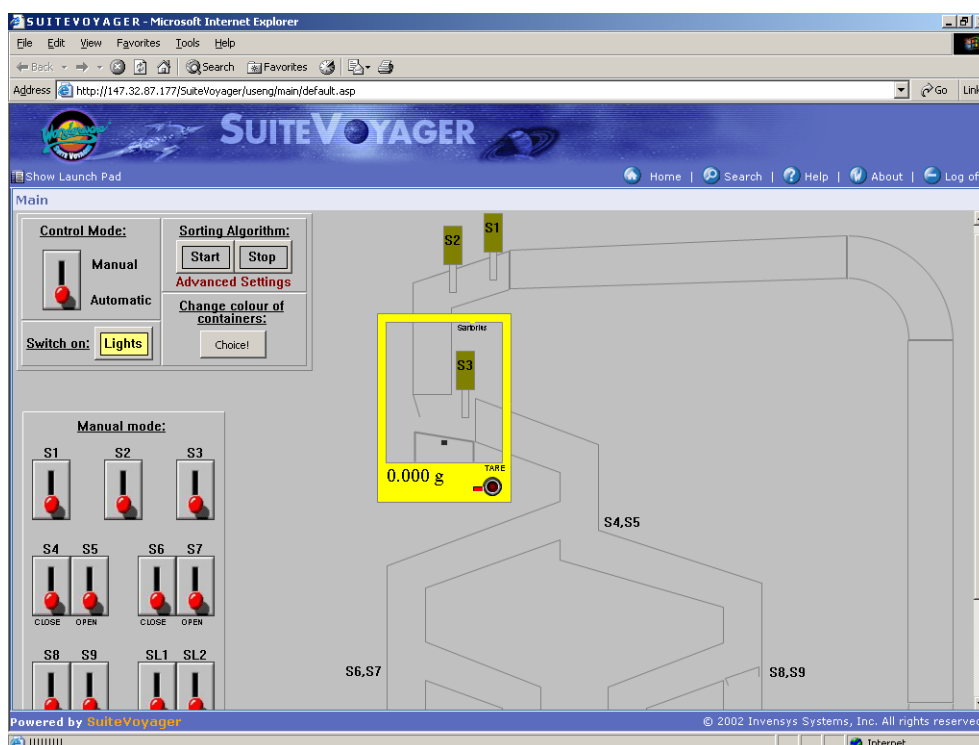
Aplikaci vizualizace třídícího mechanismu je po příslušném přihlášení možné ovládat i z webového prohlížeče MS Internet Explorer (toto je doporučený prohlížeč,

<sup>7</sup>Pokud je obdélníček červený, tak se jedná o stabilní údaj.

<sup>8</sup>Po potvrzení se změna projeví v paměti PLC S7-300.

u jiných není zaručena správná funkce). Uživatel s neomezenými přístupovými právy (ze skupiny Administrators v rámci SUITEVOYAGERU) může plně ovládat (povelovat) vizualizační aplikaci třídícího mechanismu a také provádět změny v systémovém nastavení SUITEVOYAGERU viz obrázek 2.25. Naopak uživatel s omezenými přístupovými právy může jen sledovat průběh vizualizace a nemůže ovládat (povelovat) třídící mechanismus.

Aplikace vizualizace ve SUITEVOYAGERU je zobrazena na obrázku 4.9. Funkce je úplně stejná jako u WINDOWVIEWER s jedním omezením, a to, že nelze měnit barvy u šachet (C1 až C4). Toto omezení je dáno tím, že SUITEVOYAGER nepodporuje všechny skripty<sup>9</sup>, které je možné použít u WINDOWVIEWERU. Dále také správně nefungovala ve WINDOWVIEWERU skupina "radio buttonů", pomocí kterých se volí barva pro danou šachtu.



Obrázek 4.9: Vizualizace ve SUITEVOYAGERU

<sup>9</sup>Zde se jedná konkrétně o cyklicky se opakující skript aktivního okna.

# Kapitola 5

## Statistika – Určení hmotnostních intervalů u míčků

Jak již bylo řečeno v kapitole 2.1.5, vážení míčků je zatíženo jistou chybou, vzniklou třením míčku o stěny a mechanickou závoru. Proto bylo uskutečněno měření, při kterém byl každý míček nezávisle zvážen<sup>1</sup> 16-krát, přičemž doba vážení byla 3 a 15 vteřin od dopadnutí míčku na vážící mechanismus z horního zásobníku.

Naměřené hmotnosti pro jednotlivé míčky znázorňuje graf na obrázku 5.1, kde v levé části jsou měření po 3 a v pravé části 15 vteřinách. Jednotlivé barvy průběhů odpovídají barvám míčků kromě černé barvy, která zastupuje žluté míčky kvůli lepší čitelnosti. Z průběhů je patrné, že měření po 3 vteřinách mají mnohem menší rozptyl než měření po 15 vteřinách. To je způsobeno tím, že se váha neustále snaží vynulovat odchylku vážícího mechanismu (aby dosáhla stabilního údaje), což ale vede ke skokovým změnám hmotnosti<sup>2</sup> vlivem tření míčku o stěny. Proto pro další statistické zpracování jsou použita jen data změřená 3 vteřiny po dopadnutí míčku na vážící mechanismus. V praxi se ukázalo, že ideální doba pro zvážení míčku je někde mezi třemi až pěti vteřinami.

K určení mezí (viz tabulka 3.3), podle kterých se rozpoznává o jakou barvu míčku se jedná je nutné znát, v jakém intervalu se bude nacházet většina<sup>3</sup> měření pro daný míček. Za předpokladu chyby měření popsané v předchozím odstavci a vyloučení ostatních<sup>4</sup> chyb měření, lze rozdělení měřených dat aproximovat normálním rozdělením.

Normální rozdělení je určeno střední hodnotou  $\mu$  a rozptylem  $\sigma^2$ , u kterého se používá označení s kvadrátem, aby se zdůraznilo, že nabývá pouze kladných hodnot. Je známo, že v intervalu  $\langle \mu - 3\sigma, \mu + 3\sigma \rangle$  leží 99.7% všech naměřených hodnot. Je tedy nutné ze vzorku naměřených dat (tzv. výběrový soubor) určit parametry  $\mu$  a  $\sigma^2$  pro všechna měření, které nastanou v budoucnu (tzv. populace).

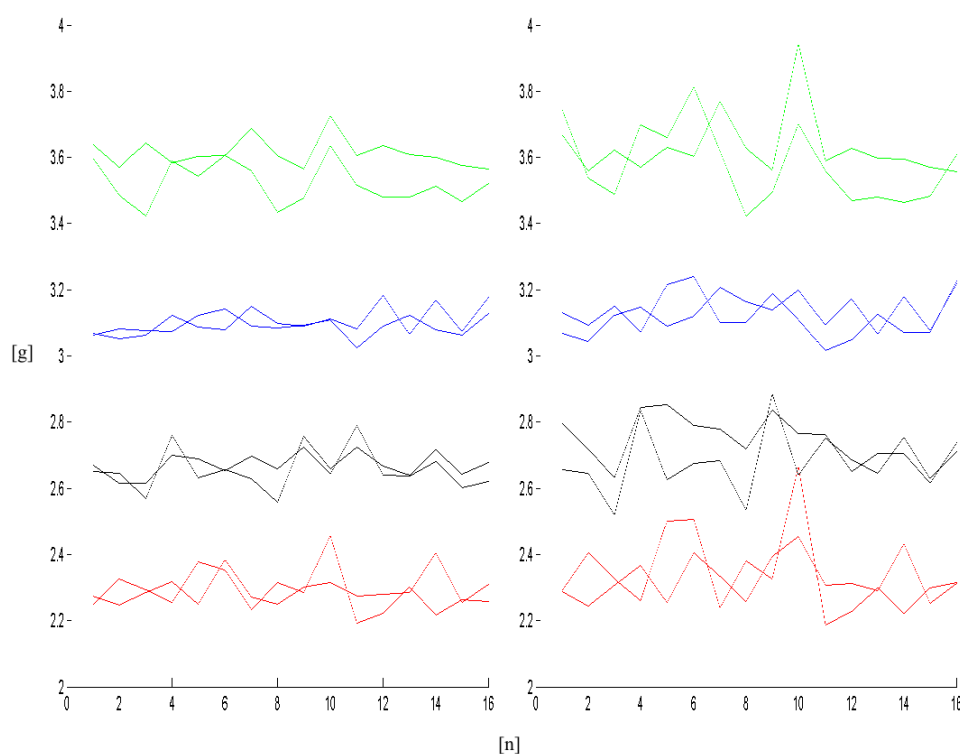
---

<sup>1</sup>Byla použita funkce ve STEP7, která vždy v daném okamžiku (okamžik, kdy se z hmotnosti míčku rozhoduje o jakou barvu se jedná a nastavují se příslušné zarážky) zaznamenala hmotnost do datového bloku.

<sup>2</sup>Tyto skokové změny způsobují, že váha nedosáhne stabilního údaje ani po pěti minutách vážení.

<sup>3</sup>Dejme tomu 99%.

<sup>4</sup>Např. zamezení větších vibrací při vážení nebo vniknutí cizího tělesa na vážící mechanismus atd.



Obrázek 5.1: Graf naměřených hmotností míčků – vlevo 3 s, vpravo 15 s

**Pozn.:** Tato statistika nebere v úvahu změnu hmotnosti míčků v závislosti na čase (vlivem tření míčků o stěny a špiněním míčků) a též nutnost cejchovat váhu po určité době provozu.

## 5.1 Určení $\mu$ a $\sigma^2$

Vzhledem k tomu, že se jedná o aproximaci normálním rozdělením, a že není k dispozici velký počet (řádově stovky až tisíce) naměřených dat, nelze určit  $\mu$  jako aritmetický průměr a  $\sigma^2$  jako výběrový rozptyl (viz dále). Lze pouze s nějakou pravděpodobností tvrdit (odhadnout), pokud má být odhad věrohodný, že  $\mu$  a  $\sigma^2$  leží v nějakém intervalu nebo jsou shora resp. zdola omezeny - tj. statistika. Jedná se o tzv. intervalové odhady parametrů. U parametru  $\mu$  nás zajímá, v jakém konkrétním intervalu leží (tj. oboustranný interval) a u  $\sigma^2$  je důležitá její maximální hodnota (tj. jednostranný interval).

Jestliže máme  $n$  měření  $X_1, X_2, \dots, X_n$ , pak intuitivní odhad střední hodnoty výběrového souboru je

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i. \quad (5.1)$$

Výběrový rozptyl pro výběrový soubor dat  $X_1, X_2, \dots, X_n$  je

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2. \quad (5.2)$$

Střední hodnotu  $\mu$  lze tedy určit jako  $100 \cdot (1 - \alpha)\%$  interval spolehlivosti<sup>5</sup> (oboustranný)

$$P \left[ \bar{X} - t_{1-\frac{\alpha}{2}}(n-1) \frac{S}{\sqrt{n}} < \mu < \bar{X} + t_{1-\frac{\alpha}{2}}(n-1) \frac{S}{\sqrt{n}} \right] = 1 - \alpha, \quad (5.3)$$

kde  $n$  je počet naměřených dat (velikost výběrového souboru) a  $t_\beta(\nu)$  je  $\beta$ -kvantil  $t$ -rozdělení<sup>6</sup> s  $\nu$  stupni volnosti, kde  $\nu = n - 1$ . Pro větší výběrový soubor dat ( $n > 30$ ) lze  $t_\beta(\nu)$  nahradit  $u_\beta$  ( $\beta$ -kvantil normálního rozdělení).

Horní mez odhadu rozptylu  $\sigma^2$  lze určit jako jednostranný  $100 \cdot (1 - \alpha)\%$  interval spolehlivosti

$$P \left[ 0 < \sigma^2 < \frac{(n-1)S^2}{\chi_\alpha^2(n-1)} \right] = 1 - \alpha, \quad (5.4)$$

kde  $\chi_\beta^2(n-1)$  je  $\beta$ -kvantil  $\chi^2$  rozdělení s  $n - 1$  stupni volnosti.

## 5.2 Určení intervalů z naměřených dat

Z naměřených dat bylo pomocí vztahů 5.3 a 5.4 vypočteno pro každý míček  $\mu$  a  $\sigma^2$  viz tabulka 5.1, přičemž byl uvažován 99% interval spolehlivosti ( $\alpha = 0,01$ ).

Míček	$T = 3 \text{ s}$		$T = 15 \text{ s}$	
	$\mu$ [g]	$\sigma$ [g]	$\mu$ [g]	$\sigma$ [g]
červený 1	$\langle 2.24, 2.34 \rangle$	0.12	$\langle 2.25, 2.43 \rangle$	0.21
červený 2	$\langle 2.26, 2.33 \rangle$	0.08	$\langle 2.28, 2.37 \rangle$	0.11
žlutý 1	$\langle 2.63, 2.70 \rangle$	0.08	$\langle 2.69, 2.80 \rangle$	0.12
žlutý 2	$\langle 2.62, 2.70 \rangle$	0.10	$\langle 2.61, 2.75 \rangle$	0.16
modrý 1	$\langle 3.07, 3.11 \rangle$	0.05	$\langle 3.07, 3.17 \rangle$	0.11
modrý 2	$\langle 3.07, 3.14 \rangle$	0.07	$\langle 3.09, 3.17 \rangle$	0.09
zelený 1	$\langle 3.47, 3.57 \rangle$	0.11	$\langle 3.49, 3.66 \rangle$	0.20
zelený 2	$\langle 3.58, 3.65 \rangle$	0.08	$\langle 3.56, 3.70 \rangle$	0.17

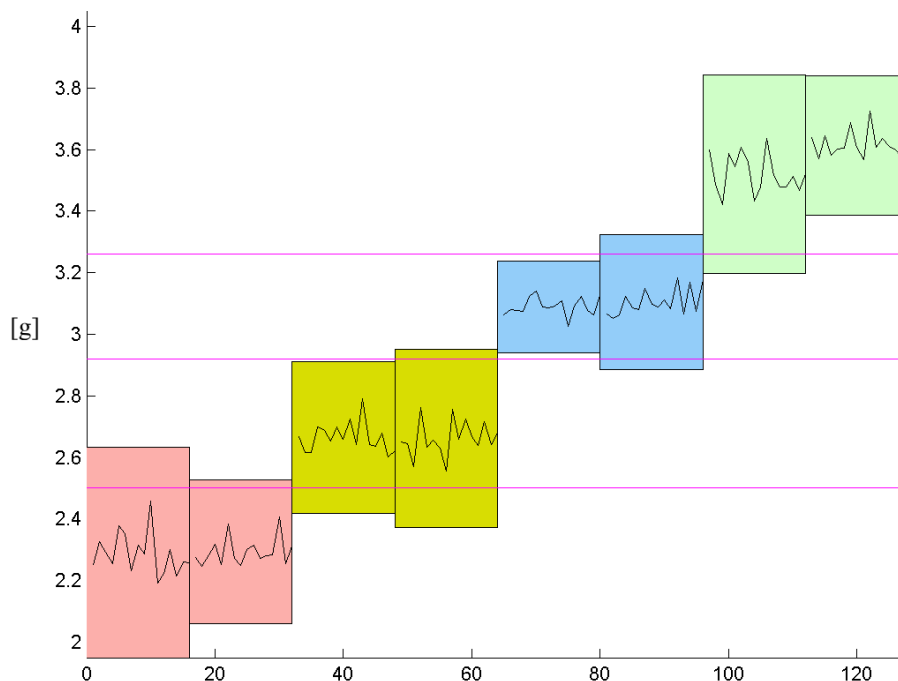
Tabulka 5.1: Vypočtený interval  $\mu$  a horní odhad  $\sigma$  pro všechny míčky

Z vypočtených  $\mu$  a  $\sigma^2$  byly určeny intervaly, ve kterých by se mělo nacházet 99% všech měření pro daný míček, jako  $\langle \mu_d - 2.58\sigma, \mu_h + 2.58\sigma \rangle$ , kde  $\mu_d$  je dolní mez odhadu

<sup>5</sup>Tzn., že v daném intervalu leží střední hodnota  $\mu$  s pravděpodobností  $1 - \alpha$ , kde  $1 - \alpha$  se nazývá koeficient spolehlivosti.

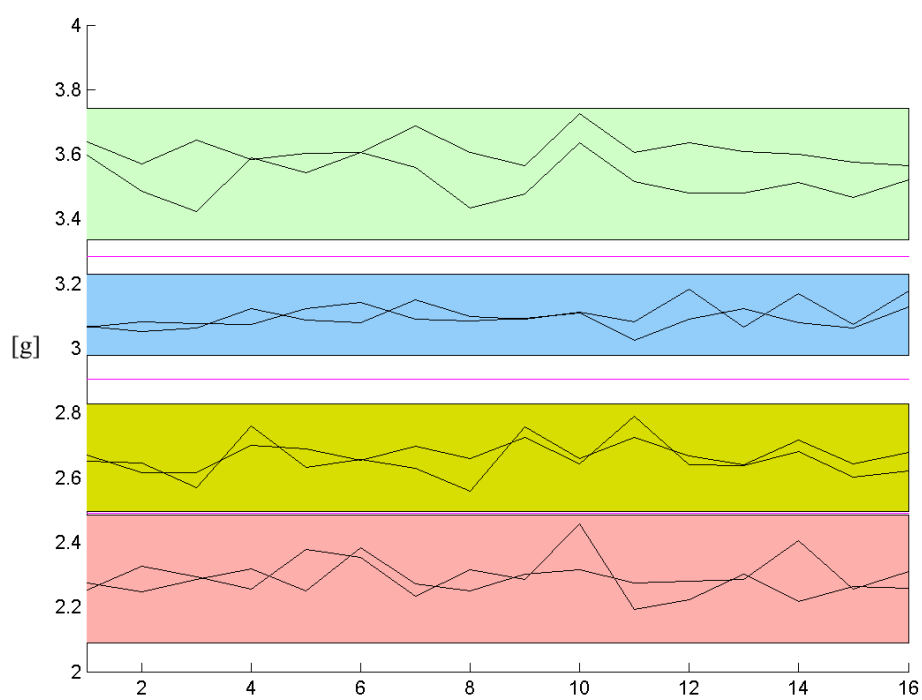
<sup>6</sup>Nazývané též Studentovo rozdělení.

$\mu$  a  $\mu_h$  horní mez. Tyto intervaly jsou na obrázku 5.2 pro každý míček zvlášť, přičemž jejich barva odpovídá barvě míčku. Z obrázku je patrné, že se intervaly jednotlivých barev míčků navzájem překrývají, a tudíž není zaručena 99% spolehlivost roztrídění míčků. K dosažení 99% spolehlivosti (podle této statistiky) by míčky musely být od sebe více hmotnostně vzdáleny, aby nedocházelo k překrývání intervalů.



Obrázek 5.2: Hmotnostní intervaly a meze pro 99% spolehlivost

Spolehlivost správného roztrídění stávajících míčků je přibližně 90%, což odpovídá intervalu  $\langle \mu_d - 1.65\sigma, \mu_h + 1.65\sigma \rangle$ , kde parametry  $\mu$  a  $\sigma^2$  byly určeny na 95% intervalu spolehlivosti. Tento interval pro jednotlivé míčky je na obrázku 5.3, přičemž výsledný interval dané barvy je sjednocením intervalů obou míčků stejné barvy. Na obrázcích 5.2 a 5.3 jsou fialovou vodorovnou úsečkou znázorněny hranice, pomocí kterých se rozhoduje, o jakou barvu míčku se jedná. Tyto hranice leží přesně ve středu mezi intervaly sousedních míčků a číselně odpovídají mezím (z obrázku 5.3) v tabulce 3.3, přičemž dolní mez žlutého míčku a horní mez zeleného míčku jsou určeny s ohledem na velikost intervalu.



Obrázek 5.3: Hmotnostní intervaly a meze pro 90% spolehlivost



# Závěr

Vlastní diplomovou práci lze rozdělit na hardwarovou a softwarovou sekci. V hardwarové sekci byl navržen a realizován model třídícího mechanismu včetně řídicího systému. V softwarové sekci byl nakonfigurován Terminal Server pro vzdálené řízení a naprogramovány veškeré aplikace pro řídicí systém a vizualizaci průběhu řízení.

Ve spojení operačního systému Windows 2000 Server a vývojového prostředí Step7 (obojí běží na Terminal Serveru) je možné vzdáleně, pomocí sítě Internet, programovat programovatelný automat CPU 315-2 DP. Také je možné vzdáleně vytvořit v prostředí WindowMaker novou vizualizační aplikaci. Při vzdáleném přihlášení k Terminal Serveru je k dispozici klasická pracovní plocha Windows jako u lokálního přihlášení s omezením počtu barev na 256. Terminal Server bude zabezpečen, proti napadení hackery, začleněním do projektu Lablink pomocí externího firewallu, který spolupracuje s rezervním systémem a umožní přístup jen registrovaným uživatelům.

Průběh řízení je možné ovlivnit pomocí aplikace WindowViewer (nutnost přihlášení k Terminal Serveru) nebo přímo z webového prohlížeče pomocí SuiteVoyageru, který běží pod webovým serverem IIS a vyžaduje režim přístupu na základě ověřování. V případě použití SuiteVoyageru není možné provozovat webový server IIS také v režimu anonymního přístupu (oba režimy lze kombinovat) pro webové stránky třídícího mechanismu (aby bylo možné webové stránky prohlížet bez přihlašování). Pokud je IIS provozován také v režimu anonymního přístupu, tak SuiteVoyager neumožní zpětný zásah do řídicího systému a jen pasivně zobrazuje průběh řízení. Proto webové stránky budou umístěny na katedrální webový server DCE a webový server IIS (na Terminal Serveru) bude provozován jen pro účely SuiteVoyageru v režimu přístupu na základě ověřování.

V kapitole 5 bylo zjištěno, že model třídícího mechanismu ve stávající podobě dosahuje úspěšnosti třídění přibližně 90%. Spolehlivost třídění by šlo zvýšit větším odstupňováním hmotností vážených míčků (omezením je výkon fukaru) nebo jiným způsobem vážení. Tím je myšleno nahradit současnou šikmou plošinku, na vážícím mechanismu váhy, rovnou plošinkou tak, aby byl míček v průběhu vážení v kontaktu jen s vážícím mechanismem. Dále by pak bylo nutné mechanicky zajistit opuštění míčku z vážícího mechanismu.

Detekce míčku pomocí snímače na vážícím mechanismu se ukázala jako nedostatečná, protože v některých případech došlo k tomu, že snímač míček nezaregistroval. Pak stačilo lehce s míčkem pohnout a snímač již míček detekoval. Proto je vhodnější míček detekovat pomocí změny údaje hmotnosti (nenulová hmotnost signalizuje míček) a informaci ze snímače považovat za redundantní.

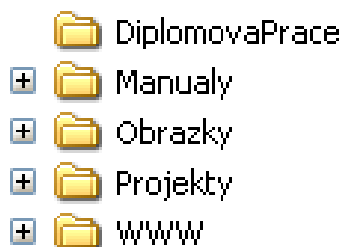
# Literatura

- [1] Bílek J., Burget P. *Průmyslové sítě a sběrnice, přednášky*. ČVUT FEL KŘT, Praha, 2001.
- [2] Richter J. *Průmyslová sběrnice PROFIBUS DP, diplomová práce*. ČVUT FEL KŘT, Praha, 1999.
- [3] PROFIBUS *Techology and Application, System Description*. PROFIBUS Nutzenorganisation, Germany, 2002.
- [4] Žoldák M. *Real-time komunikační protokol nad protokolem UDP, semestrální práce (v prog. jazyce C++) z předmětu "Návrhy automatizačních zařízení"*. ČVUT FEL KŘT, Praha, 2003.
- [5] W. Richard Stevens *The Protocols (TCP/IP Illustrated, Volume 1)*. Addison-Wesley Pub Co, 1994.
- [6] Demlová M. *Matematika 6D, přednášky*. ČVUT FEL KM, Praha, 2001.
- [7] Rogalewicz V. *Pravděpodobnost a statistika pro inženýry, skriptum FEL*. Ediční středisko ČVUT, Praha, 2000.
- [8] SIMATIC NET *DP/RS 232C Link, User Description*. SIEMENS AG, Germany, 1999.
- [9] SIMATIC *PLC S7-300, CPU Specifications CPU 312 IFM to CPU 318-2 DP, Reference Manual*. SIEMENS AG, Germany, 2001.
- [10] SIMATIC *S7-300 Programmable Controller Module Specifications, Reference Manual*. SIEMENS AG, Germany, 2002.
- [11] SIMATIC *Programovatelný automat S7-200, Systémový manuál*. SIEMENS s.r.o., Brno, 2002.
- [12] SIEMENS *SIMATIC S7-200 EM 221 Digital Input 16 x 24 VDC, EM 222 Digital Output 4 x 24 VDC-5A, EM 222 Digital Output 4 x Relays-10A, Data*. SIEMENS AG, Germany, 2002.
- [13] Sartorius *MP8-1/MP8-2/Data Input, Interface Description*. Sartorius GmbH, Germany, 1983.

- 
- [14] Sartorius *Electronic Analytical Balances 1601MP8, 1602MP8 and 1612MP8, Installation and Operating Instructions*. Sartorius GmbH, Germany, 1983.
- [15] WONDERWARE *Wonderware Siemens Simatic NET S7 I/O Server, User's Guide*. Wonderware Corporation, U.S.A., 2000.
- [16] WONDERWARE *Setting Up an MPI Connection with Siemens SIMATIC NET 6.0 Software Using a CP5611 Profibus Card and an S7-400 PLC (CPU 413-1), Tech Note 303*. Wonderware Corporation, U.S.A., 2003.
- [17] AXIS Communications *AXIS 2100 Network Camera User's Guide v.1.1*.
- [18] Rybička J. *L<sup>A</sup>T<sub>E</sub>X pro začátečníky*. Konvoj, Brno, 1999.
- [19] <http://www.ad.siemens.de>
- [20] <http://www.axis.com>
- [21] <http://www.wonderware.com>

# Příloha A

## Struktura přiloženého CD ROM



Přiložený CD ROM v jednotlivých složkách obsahuje:

**DiplomovaPrace** – Tato diplomová práce ve formátu PDF (tři varianty: odsazení pro jednostranný tisk, odsazení pro obustranný tisk, bez odsazení).

**Manualy** – Dokumentace k používanému hardwaru a softwaru ve formátu PDF.

**Obrazky** – Veškeré použité obrázky ve formátech PNG, JPG, PDF a CDR (Corel Draw). Dále jsou zde uloženy výkresy mechanické části třídícího mechanismu ve formátu PDF.

**Projekty** – Kompletní projekt pro S7-200 v prostředí MICROWIN. Projekt pro S7-300 včetně třídícího algoritmu + základní projekt bez třídícího algoritmu umožňující manuálně ovládat akční členy (obojí ve STEP7). Projekt umožňující konfiguraci PCStation pro SIMATIC NET (STEP7). Projekt vizualizace třídícího mechanismu v prostředí INTOUCH + upravený projekt vizualizace pro SUITEVOYAGER.

**WWW** – Webové stránky projektu vzdáleného řízení třídícího mechanismu.

# Příloha B

## Konfigurace služby Terminal Services

Tuto službu je nutné nejdříve nainstalovat, protože není implicitně součástí instalace Windows. Lze ji nainstalovat v *Přidat nebo odebrat programy* → *Přidat nebo odebrat součásti systému*. Pro správnou funkci Terminal Services je nutné provést následující nastavení<sup>7</sup>:

1. Nastavení, aby se tato služba spouštěla automaticky při každém startu systému: *Nástroje pro správu* → *Služby* → *Terminal Services* – *Vlastnosti* – *Typ spuštění*: *Automaticky*.
2. Je nutné povolit vzdálené přihlášení k systému: *Nástroje pro správu* → *Konfigurace služby Terminal Services* – *Připojení* – *RDP-Tcp přenos typ Microsoft RDP 5.0* – *Vlastnosti* zde je možné provést kompletní konfiguraci a potřebné nastavení (max. povolenou dobu vzdálené relace; doba, po které se ukončí vzdálená relace po odpojení; doba, po které se ukončí nečinná relace atd.).
3. Nakonec je možné u každého uživatele zvlášť povolit resp. zakázat vzdálené připojení: *Nástroje pro správu* → *Správa počítače* – *Místní uživatelé a skupiny*.

**Pozn.:** Terminal Server je provozován v **režimu aplikačního serveru**. Přepnutí do **režimu vzdálené administrace** lze provést v *Přidat nebo odebrat programy* → *Přidat nebo odebrat součásti systému* – *Další*, a zde je možné zvolit jeden z obou režimů.

Sledování aktivních vzdálených relací a jejich případné odpojení lze provést v: *Nástroje pro správu* → *Správce služby Terminal Services*.

---

<sup>7</sup>*Nástroje pro správu* lze nalézt: *Start* → *Nastavení* → *Ovládací panely*.