

Průmyslové sběrnice

Industrial Bus

Bc. Josef Stašiak

Diplomová práce
2005



Univerzita Tomáše Bati ve Zlíně
Fakulta Technologická

Vložit oficiální zadání diplomové/bakalářské práce

Děkuji mému vedoucímu diplomové práce Ing. Viliamu Dolinayovi za odborné vedení a pomoc v průběhu řešení této diplomové práce.

Prohlašuji, že jsem na celé diplomové práci pracoval samostatně a použitou literaturu jsem citoval.

Ve Zlíně, 06. 06. 2005

.....

jméno diplomanta

RESUMÉ

Cílem této diplomové práce bylo provedení průzkumu a zpracování přehledného popisu průmyslových sběrnic, používaných v praktických hierarchických aplikacích řídicích a monitorovacích systémů.

Bylo provedeno seznámení s technologií rotační vakuové odparky, která bude umístěna v laboratoři mokrých procesů Univerzity T. Bati ve Zlíně.

Realizováno bylo zapojení a oživení snímacích a řídicích členů, včetně propojení s řídicím počítačem, na kterém je nainstalován n řídicí a monitorovací program Control Web 2000 se začleněným protokolem Modicon Modbus.

SUMMARY

The purpose of this master thesis was research and processing of description of industrial buses which are used in practice in hierarchical applications of control and monitoring systems.

The rotating vacuum evaporator which will take place in laboratory of wet processes of UTB Zlin is described in the thesis too.

Finally was realized a connection and activation of sensors and controlers including connection with control computer with control and monitoring software Control Web 2000 (software includes Modicon Modbus protocol) installed.

OBSAH

RESUMÉ.....	4
OBSAH.....	5
ÚVOD.....	7
1 DATOVÉ SÍTĚ	8
1.1 OBECNÉ POJMY DATOVÝCH SÍTÍ.....	8
1.1.1 Základní pojmy.....	8
1.1.2 Topologie sítí.....	9
1.1.3 Kódování a modulace přenosu	10
1.1.4 Synchronizace.....	10
1.1.5 Zabezpečení přenosu dat	11
1.1.6 Potvrzování zpráv	13
1.2 REFERENČNÍ MODEL ISO/OSI.....	13
1.2.1 Mezisíťová spojení	15
1.3 METODY ŘÍZENÍ PŘÍSTUPU K PŘENOSOVÉMU MÉDIU.....	16
2 PRŮMYSLOVÉ SBĚRNICE.....	17
2.1 STANDARD RS-232C, RS-422A A RS-485.....	17
2.1.1 RS 232C	17
2.1.2 RS 422A	17
2.1.3 RS 485	18
2.2 CAN.....	18
2.2.1 Řízení přístupu k médiu a řešení kolizí	21
2.2.2 Zabezpečení přenášených dat a detekce chyb	22
2.2.3 Signalizace chyb.....	23
2.2.4 Datová zpráva (Data Frame).....	24
2.2.5 Žádost o data (Remote Frame)	26
2.2.6 Chybová zpráva (Error Frame).....	26
2.2.7 Zpráva o přetížení (Overload Frame)	27
2.3 DEVICENET.....	27
2.4 MODBUS	28
2.4.1 Aplikační protokol.....	29
2.4.2 Kódování dat	31
2.4.3 ModBus pře sériovou linku	33
2.4.4 MODBUS přes TCP/IP	34
2.4.5 ModBus PLUS (ModBus+).....	36
2.5 ASI.....	36
2.6 FOUNDATION FIELDBUS	40
2.7 HART	41
2.8 LIN (LOCAL INTERCONNECT NETWORK)	44
2.9 M-BUS.....	48
2.9.1 Základní charakteristika	49
2.10 MEASUREMENT BUS	50
2.10.1 Základní charakteristika	50
2.11 PROFIBUS	51
2.12 INTERBUS-S	52
2.13 WI-FI.....	52
3 PRAKTICKÁ ČÁST.....	55
3.1 POPIS A SCHÉMA ROTAČNÍ ODPARKY	55
3.1.1 Popis funkce rotační odparky	55
3.1.2 Schéma rotační odparky	56

3.1.3	Zapojení regulačního obvodu rotační odparky	56
3.2	PROGRAM PACTWARE 2.4.....	58
3.3	NASTAVENÍ BRÁNY (GATEWAY) KSD2-GW-MOD	61
3.4	CONTROL WEB 2000.....	62
3.4.1	Základní informace.....	62
3.4.2	Vývojové prostředí	65
3.4.3	Vytváření aplikace.....	71
3.4.4	Nastavení konstant, ovladače a kanálů	74
3.4.5	Nastavení souborů modbus.dmf a modbus.par	75
3.5	APLIKACE PRO ŘÍZENÍ REÁLNÉHO SYSTÉMU.....	76
ZÁVĚR.....		77
LITERATURA		78
SEZNAM POUŽITÝCH ZKRATEK.....		79
SEZNAM OBRÁZKŮ.....		81
SEZNAM TABULEK		82
SEZNAM PŘÍLOH.....		83

ÚVOD

Automatizace technologických procesů dnes prakticky zasahuje do všech průmyslových odvětví. Současným trendem v automatizaci je zavádění počítačově integrovaných výrobních procesů (CIM), které umožňují efektivnější a spolehlivější výměnu informací mezi jednotlivými fázemi výrobního procesu. Vyhodnocování těchto informací lze výrobní procesy snadněji optimalizovat z různých hledisek, jako jsou efektivita výroby, kvalita, ekonomické náročnosti atd.

Nedílnou součástí automatizace je komunikační technika. Samozřejmostí je dnes propojení PC do sítí a jejich napojení na Internet. Komunikace je dnes důležitá i pro spojení řídicích systémů a jejich periferních prvků. Rychlost a zároveň spolehlivost přenosu informací v řídicích systémech je nutným předpokladem pro správnou funkci regulačních algoritmů. Mezi hlavní trendy v komunikaci, v posledních letech, je zavádění průmyslových číslicových sběrnic, které jsou schopny digitalizovanou informaci přenášet na velké vzdálenosti a to při vysoké míře spolehlivosti a zároveň rychlosti.

V současnosti se do popředí zájmů, v oblasti víceúrovňových hierarchických systémů s podporou SCADA/HMI systémů, dostává možnost přistupovat k jednotlivým aplikacím pracujících v datových sítích, z kteréhokoliv PC jak v rámci Intranetu tak i Internetu. Tuto možnost v dnešní době nabízejí téměř všechny SCADA/HMI systémy. Důležitým krokem při návrhu těchto řídicích a monitorovacích aplikací je rozdělení funkčnosti jednotlivých částí celého systému v rámci LAN na jednotlivé subsystémy.

Důležitou součástí návrhu celého systému je i ekonomické hledisko, které je nutno aplikovat i na nejnižší vrstvu řízení. V této úrovni jsou obsaženy snímače, akční členy, dílčí řídicí podsystémy, přenosové cesty ILAN atd. To vše nesmí být navrženo u úkor spolehlivosti.

Tato diplomová práce si klade za cíl průzkum a přehlednou specifikaci v typech průmyslových sběrnic, které jsou používány v praktických hierarchických aplikacích řídicích a monitorovacích systémů. Pozornost bude zaměřena hlavně na sběrnice MODBUS a CAN, které budou používány pro monitorování a řízení v laboratoři mokrých procesů pro recyklaci koželužského odpadu – odparka.

1 DATOVÉ SÍTĚ

Datová síť umožňuje jednotlivým prvkům řídicích systémů vzájemnou výměnu informací. Může jít například o naměřené hodnoty z procesu, předání hodnot akčním členům, různé chybové hlášení, informace o stavu atd.

1.1 Obecné pojmy datových sítí

V této kapitole jsou shrnuty důležité pojmy z oblasti datových sítí a sběrnic.

1.1.1 Základní pojmy

Stanice – pod tímto pojmem chápeme v průmyslových sítích počítač typu PC nebo IPC, programovatelný automat (PLC), jednočipový mikropočítač (Motorola 68HC11,...)

Spoj – slouží ke komunikaci, mezi jednotlivými stanicemi. Rozlišujeme dvoubodové a vícebodové spojení.

Přenosová cesta – propojuje jednotlivé stanice. Slouží k přenosu signálů (elektronických, optických,...). Realizujeme je metalickými vodiči, koaxiálními kabely nebo optickými vlákny.

Kanál – je cesta pro jednosměrný přenos signálů.

Okruh – jedná se o obousměrný přenos signálů dvěma kanály.

Simplexní přenos – tento typ přenosu umožňuje přenášet data pouze v jednom směru. K realizaci nám postačí pouze jeden kanál.

Poloduplexní přenos - umožňuje vysílání dat oběma směry, ale stanice mohou pouze vysílat nebo přijímat

Duplexní přenos – dovoluje přenášet data obousměrně. Stanice mohou vysílat a zároveň přijímat data. K realizaci potřebujeme 2 kanály (okruh)

Sériový přenos – je takový přenos, při kterém jsou přenášeny informace za sebou například v bitech jedním sdělovacím kanálem.

Paralelní přenos – se přenáší určitý počet signálových prvků několika sdělovacími kanály. Rychlost paralelního spojení je mnohonásobně vyšší než při přenosu sériovém v závislosti na počtu kanálu. Paralelní přenos se využívá k přenosu informací na malé vzdálenosti.

Šířka pásma přenosového kanálu je interval frekvencí, který je schopen kanál přenést. Kanál je přitom souborem technických prostředků, které se na přenosu signálu podílejí (vodiče, multiplexory, modemy, atd.).

Po síti komunikujeme pomocí zpráv. Tyto zprávy se šíří pomocí paketů, což jsou úseky pevně stanovené délky. Dlouhé zprávy jsou rozděleny do několika paketů a kratší zprávy jsou na potřebnou minimální délku doplněny. Rámec je paket před odvysíláním, který je doplněn o některé údaje (např. cílová a zdrojová adresa, CRC, synchronizační posloupnost atd.), aby mohl být správně přenesen.

Virtuálním spojením a datagramem slouží k realizaci obousměrného přenosu dat. Datagram je paket obsahující všechny informace potřebné pro přenos, obsahuje cílovou a zdrojovou adresu. Jeho výhodou je velice snadná realizace a umožňuje posílat pakety více stanicím současně, ale zase může dojít k příjmu v nesprávném pořadí. Virtuální spoj se vytváří pomocí speciálních paketů před vysíláním zpráv. V přepojovacích uzlech sítě se uloží informace o cestě, po které mají pakety putovat. Zaručeno je správné pořadí přijetí paketů.

1.1.2 Topologie sítí

Jedná se o fyzické uspořádání a způsob propojení jednotlivých stanic.

Sběrníková topologie – je nejčastěji využívanou topologií pro průmyslové sběrnice. Vysílá vždy pouze jeden vysílač, od kterého se šíří signál ke všem stanicím. Konce vodičů jsou ukončeny zkratovacími odpory (terminátory). Při poruše spojovacího kabelu nemohou jednotlivé stanice mezi sebou komunikovat. Využívána je u sítí Ethernet.

Hvězdicová topologie – charakteristikou této topologie je připojení více terminálů ke střediskovému počítači. Každý terminál má samostatný kanál. Přenos informací je řízen centrálním počítačem. Výhodou této topologie je vyšší spolehlivost při poruše kabelu, protože se odpojí pouze jeden terminál a ostatní fungují dál. Tato topologie je využívána v síti IBM Token Ring, kde centrální stanice propojuje pracovní stanice do kruhu. Centrální stanice je s každou stanicí spojena 2 kabely.

Kruhová topologie – je založena na dvoubodovém propojení sousedících stanic. Komunikační brána má jeden vstup a jeden výstup. Tuto topologii využívají hlavně sítě založené na optických přenosových médiích. Při přenosu po metalických vodičích se používá pro menší počet stanic, které jsou vždy aktivní. V případě poruchy některé stanice není síť schopna další činnosti.

Stromová topologie – tento typ zapojení sítě vychází z hvězdicové topologie. Koncoví stanici přiřadíme rozbočovací nebo řídicí funkci a připojíme k ní další koncové stanice. Z hlediska přenosových vlastností musíme definovat maximální délku mezi krajními stanicemi a z toho určit maximální zpoždění v síti. Tuto topologii využívá síť Arcnet.

Tyto základní topologické struktury můžeme vhodně kombinovat to tzv. regulárních struktur propojení. Základem pro zlepšení přenosových vlastností sítí jsou pak vícebodové spoje založené na pracovních stanicích s několika Vstupně-výstupními branami.

1.1.3 Kódování a modulace přenosu

Vysílání jednotlivých rámců probíhá sériově. Vysílané bity jsou převedeny na odpovídající signál (elektrický, optický nebo radiový).

Podle způsobu přenosu rozeznáváme:

- Přenos v základním pásmu – logické 0 a 1 jsou vyjádřeny rozdílnými úrovněmi napětí na vodiči, přičemž se ještě používají různé druhy kódování. Teoreticky lze takto přenést signál o rychlosti n vzorku za sekundu v kmitočtovém pásmu $n/2\text{Hz}$
- Přenos v přeloženém pásmu u tohoto způsobu je signál převeden na jiný typ signálu, který se daným přenosovým médiem šíří s co nejmenšími ztrátami. Obvykle to bývá sinusový signál. Způsoby modulací jsou:

- amplitudová (optické a koaxiální kabely)
- frekvenční (telefonní a rádiové spoje)
- fázová (telefonní kabely)

1.1.4 Synchronizace

Při přenosu dat je nutné zabezpečit na přijímací straně identifikaci přenášejících znaků. K tomu je zapotřebí synchronizace vysílací a přijímací stanice. Signály, které přenášejí zprávu rozdělujeme na izochronní a anizochronní. Pro izochronní signál je charakteristické, že časový odstup dvou libovolných charakteristických okamžiků (čas přechodů mezi dvěma charakteristickými stavy) je celočíselným násobkem minimálního intervalu, který nazýváme jednotkový interval. Signály, které nemají tuto vlastnost patří mezi anizochronní.

Z hlediska synchronizace rozlišujeme přenosy:

- *Synchronní* – tento přenos je uskutečňován izochronním signálem. Příjímáč je synchronizován vysílačem pomocí samostatného synchronizačního kanálu. Nevýhodou tohoto přenosu je nutnost použít dalšího přenosového kanálu.
- *Asynchronní* – uskutečňuje se anizochronním signálem. Délka intervalu, při kterém dochází k přenosu bitu (sériový přenos) nebo celého znaku (paralelní přenos) se vymezuje pomocí speciálních řídicích signálů asynchronního přenosu. Těmito signály je např. vysílač potvrzuje platnost dat a přijímač potvrzuje jejich přijetí.
- *Arytmický* – je kombinací synchronního a asynchronního přenosu. Přenos jednoho rámce je synchronní, ale přenos více rámců už je asynchronní. Na počátku rámce se přijímač a vysílač synchronizují a po dobu příjmu se synchronizace provádí pomocí vnitřního generátoru. Jeho kmitočet se však po dobu příjmu nebude shodovat s kmitočtem vysílače. Tento typ přenosu je vhodné využívat pro přenos menšího bloku dat.

Pro přenos dat v průmyslových i počítačových sítích se využívá synchronizace:

- *Bitová* – která synchronizuje každý přenášený bit. K synchronizaci se většinou používá informační signál, který mění svoji polaritu uprostřed informačního intervalu. Tato synchronizace je využita například u fázové modulace (NRZ).
- *Znaková* – která při přenosu rámce, který se při přenosu přenosovým kanálem jeví jako binární posloupnost. Nevýhodou je že musíme identifikovat jednotlivé znaky.
- *Rámcová* – je využívána pro lepší orientaci v přenášeném řetězci a na začátku a konci rámce je vysílána tzv. křídová značka (flag), od které se další informace odvozují. Křídovou značkou může být poslání například posloupnost pěti jedniček.

1.1.5 Zabezpečení přenosu dat

Při přenosu dat mezi vysílačem a přijímačem může dojít ke ztrátě nebo změna přenášených bitů. Aby přijímací stanice poznala, jestli jsou přenesené bity správně přijaté, je využíváno různých druhů zabezpečení. Jejich princip spočívá, že k bitům přenášejících samotnou informaci se přidávají další bity, podle kterých pozná přijímací stanice zda je platnost přijatých dat správná.

Využívá se těchto zabezpečení:

- *Paritním bitem* – který je přidán k přenášené zprávě. Tento bit se získá sečtením bytů (výsledek může nabývat pouze hodnot 0 a 1) a v paritním bytu jsou doplněny tak aby dali součet 0, jedná se pak o sudou paritu nebo 1, jedná se pak o lichou paritu. Nevýhodou je, že toto zabezpečení umí pouze detekovat změnu lichého počtu bitů. Tento typ zabezpečení je využíván pro přenos menšího množství dat převážně u rozhraní RS-232 na počítačích PC.
- *Kontrolním součtem* – který je přidán k přenášené zprávě. Tento kontrolní součet je vytvořen prostým sčítáním jednotlivých přenášených znaků a výsledný součet je připojen na konec zprávy. Příjímač provede součet a porovná jej z kontrolním součtem. Často se místo celého kontrolního součtu přenáší jen jeho posledních n (nejčastěji $n=8$) bitů. Tímto se ale sníží zabezpečovací schopnost.
- *Cyklický kód* – tento způsob zabezpečení přenosu dat patří k nejčastějším způsobům zabezpečování. Jestliže bude počet bitů přenášených dat „ k “, počet bitu pro zabezpečení „ m “, pak celkový počet přenášených bitů bude „ $n=m+k$ “. Zprávu o délce k bitů můžeme zapsat pomocí mnohočlenu proměnné x s k členy stupně $(k-1)$ $G(x) = a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + \dots + a_1x + a_0$,
- kde a_i pro $i=k-1, k-2, \dots, 1, 0$ nabývá hodnot pouze 0 a 1 a vyjadřuje bity zprávy. Dále musí také platit $0 < m < k - 1$.

Zabezpečení zprávy se provede následující operací v aritmetice modulo 2:

$$\frac{G(x)x^{n-k}}{P(x)} = Q(x) + \frac{R(x)}{P(x)}$$

kde $P(x)$ je generační polynom m -tého stupně daného cyklického kódu, $Q(x)$ je podíl a $R(x)$ je zbytek po dělení generačním polynomem.

Vysílaná zpráva pak bude dána mnohočlenem

$$F(x) = G(x)x^{n-k} + R(x).$$

Výsledná zpráva pak bude obsahovat původní zprávu $G(x)$ a na zbývajících $n-k$ místech zbytek $R(x)$ po dělení generačním polynomem.

Z teorie lze odvodit celou řadu generačních polynomů, které mají nejvýhodnější vlastnosti pro určitou maximální délku přenášených dat. Normy pro konkrétní sítě generační polynom předepisují a například průmyslová sběrnice CAN používá generační polynom

$$x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + x.$$

1.1.6 Potvrzování zpráv

Při přenosu rámců mezi stanicemi musíme počítat s poškozením přenášených dat. Mhou nastat tyto případy:

- Rámec je doručen do cílové stanice, ale během přenosu dojde k jeho poškození. Poškození rámce detekuje přijímač pomocí popsanych metod zabezpečení přenosu.
- Rámec k cílové stanici nedorazí.

Pokud nastane jedna z uvedených situací, musí systém zajistit opakované vysílání ztraceného rámce. Pro kontrolu přenosu se používají tyto metody:

- *Pozitivní potvrzování* – pokud přijímací stanice přijme rámec správně, potvrdí jeho přijetí vysílací stanici vysláním speciálního potvrzovacího rámce (ACK). Pokud je rámec přijat chybně nebo vůbec, přijímací stanice ACK neodešle. Vysílací stanice se pak po určité době (time-out) zopakuje vysílání příslušného rámce. Nevýhodou je že při ztrátě rámce dojde ke zdvojenému přenosu rámce.
- *Negativní potvrzování* – vysílací stanice očekává potvrzení až po odeslání N rámců. Přijímací stanice potvrzuje příjem posledního přijatého rámce.
- *Nesamostatné potvrzování* – v rámci je vyhrazeno zvláštní pole pro číslo naposledy přijatého rámce.

1.2 Referenční model ISO/OSI

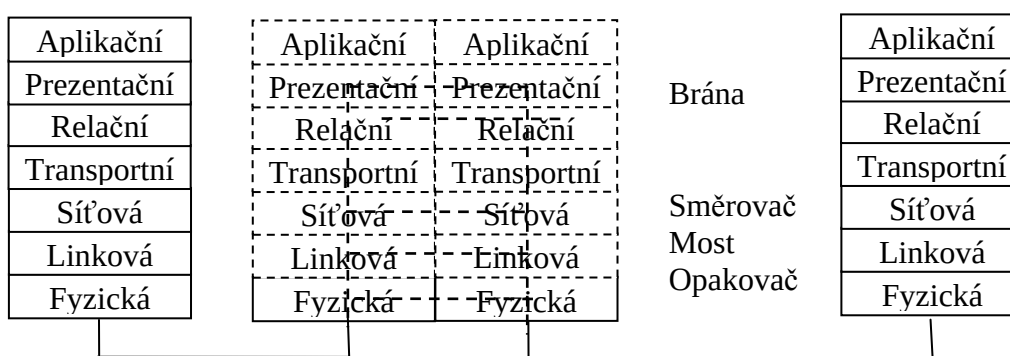
Doporučení pro vzájemnou komunikaci mezi stanicemi až do úrovně specifikace přenosu dat nebo programů vydala organizace Internaional Standard Organization (ISO) a označila ho jako Open Systém Interconnection (ISO/OSI). Dle této normy se komunikační protokoly člení do sedmi vrstev viz obr*****:

- *Fyzická vrstva (Physical Layer)* - hlavním úkolem této vrstvy je příjem a vysílání signálů prostřednictvím přenosového média. Norma přesně definuje fyzikální charakteristiky signálů (rychlost přenosu, způsob kódování,...), druhy propojovacích konektorů, typ přenosového média (kroucená dvoulinka, optické vlákno, koaxiální kabel,..) a topologii sítě (sběrnice, kruh, hvězda,...). Přijaté bity

jsou předávány vyšší vrstvě a vysílání se provádí dle požadavků vyšší vrstvy. Příkladem v síti je opakovač.

- *Linková vrstva (Data Link Layer)* - určuje způsob předávání zpráv v síti. Základní funkcí této vrstvy je vytváření a příjem rámců a zajišťuje přenos mezi dvěma uzly sítě. Rozpoznává a určuje význam jednotlivých bitů a proto obstarává potvrzování zpráv, kontrolu správnosti přenosu dat, atd. Příkladem může být síťový most.
- *Síťová vrstva (Network Layer)* – zajišťuje spojení, které nedokáže zajistit linková vrstva. Síťová vrstva musí znát topologii sítě. U mnohobodové sítě musí určit způsob předávání rámce řetězcem uzlů umístěných mezi koncovými stanicemi. Příkladem v síti je směrovač.
- *Transportní vrstva (Transport Layer)* – zajišťuje spolehlivý přenos zpráv mezi koncovými stanicemi. Tato vrstva rozkládá zprávy na jednotlivé pakety, opakuje zprávy v případě chyby, navazuje a ruší spojení.
- *Relační vrstva (Session Layer)* – musí zabezpečit nepřetržitou komunikaci. Jejím hlavním úkolem je vytváření relací mezi koncovými procesy. Vrstva identifikuje uživatele, ověřuje jeho přístupová práva, eviduje provoz a v případě komerčních sítí stanovuje a účtuje poplatky za provoz sítě.
- *Prezentační vrstva (Presentation Layer)* – tato vrstva se stará o správnou výměnu dat – transformuje data do formy vhodné pro přenos, převádí kódy a formáty dat pro nekompatibilní počítače, komprimuje data pro lepší využití komunikačního kanálu a šifruje data podle požadovaného utajení. Vrstva také vytváří neutrální formu dat, která pak může být interpretována různými systémy. Příkladem v síti je brána.
- *Aplikační vrstva (Application Layer)* – má za úkol přímou podporu aplikací – komunikační, systémově orientované a aplikační služby. Patří sem například elektronická pošta, databázové programy, přenos souborů atd.

Jednotlivé vrstvy mohou být interpretovány softwarově nebo hardwarově.



Obrázek 1: Referenční model ISO/ OSI

1.2.1 Mezisíťová spojení

Ke spojování případně rozšiřování sítí se používají opakovače (repeaters), můstky (bridges), směrovače (routers) a brány (gateways). Jejich vztah k jednotlivým vrstvám referenčního modelu OSI/ISO je znázorněn na ***Obrázku ***

- *Opakovač (repeater)* – umožňuje prodloužení sítě. Zesilují signál, který je pro přenos slabý. Je to v podstatě neinteligentní zesilovač s těmito vlastnostmi:
 - obnovuje síťové signály
 - oba propojované segmenty musí mít stejnou přístupovou metodu
 - segmenty propojené opakovačem jsou částí jedné sítě a mají tutéž síťovou adresu
- *Most (bridge)* – spojuje dva podobné nebo různé segmenty sítě. Přesná definice síťového mostu neexistuje, ale zle říct, že můstek umí seřadit pakety podle příslušnosti k segmentům a to bez ohledu na jejich fyzickou podstatu. Neumí nasměrovat rámce, tzv. optimální cestě. Podle adres paketů je třídí a posílá po příslušných segmentů sítě. Most se používá:
 - rozšíření sítě, pokud bylo dosaženo maximální vzdálenosti
 - rozdělení hlavního segmentu na několik menších, což má za následek zmenšení počtu účastníků na jednou segmentu a tím zvýšením propustnosti sítě
 - spojení různých typů sítí

- *Směrovač (router)* – tento síťový prvek se používá pro vzdálené komunikační spojení v rozsáhlých sítích nebo při přechodu mezi jednotlivými sítěmi. Směrovače umí najít nejvhodnější cestu pro doručení paketu. Používá tabulku propojení sítě, aby mohl zajistit optimální nasměrování.
- *Brána (gateway)* – slouží pro spojení LAN, které používají jiné operační systémy. Například spojení mezi LAN a středisko počítačů.

1.3 Metody řízení přístupu k přenosovému médium

V obecném případě neznáme u počítačových sítí ani průmyslových sběrnic pořadí stanic při vysílání. Proto potřebujeme zajistit řízení provozu, které určuje jakým způsobem budou jednotlivé stanice přistupovat k přenosovému médium a tím pádem si předávat data, aniž by došlo k rušení ostatních stanic. Zvolený přístup by měl být zvolen tak, aby nedocházelo k přetížení sítě požadavky na přenos a ani nedostatečným vytížením sítě.

Podle způsobu přístupu k přenosovému médium rozeznáváme přístupy:

- *deterministický* – jedná se o centralizované nebo distribuované řízení. Vysílání je řízeno řídicí stanicí.
- *Stochastický (náhodný)* – vysílání není řízeno žádnou nadřazenou stanicí a jednotlivé stanice přistupují k přenosovému médium náhodně. Výhodou je že není nutná řídicí stanice a vysílací stanice vysílají dle potřeby. Nevýhodou je že dochází ke kolizím (více stanic vysílá současně) a tím se sníží objem přenesených dat.

Toto rozdělení není však plně závazné. Při realizaci konkrétní sítě může být využita vhodná kombinace popsaných metod. Praktickým příkladem je sběrnice ProfiBus, která využívá řízení master/slave s více zařízeními master, které si pak postupně předávají pověření.

2 PRŮMYSLOVÉ SBĚRNICE

Průmyslové sběrnice vycházejí z konstrukce klasických počítačových sítí LAN. Hlavní rozdíly jsou především vyšší spolehlivost přenosu dat a možnost použití v extrémních podmínkách.

Při distribuovaném systému řízení, jsou jednotlivá měřicí místa často vzdálena, jak od řídicího počítače tak i od sebe navzájem a z tohoto důvodu je málo vhodné přenášet od snímačů analogový signál do řídicího počítače. Vhodnější je přenášet data v digitální formě, kde je menší pravděpodobnost odhalení chyby.

2.1 Standard RS-232C, RS-422A a RS-485

Sériové rozhraní RS-232c, RS-422A a RS-485 se dnes běžně používají v distribuovaných systémech řízení, pro monitorování, řízení a komunikaci mezi procesy.

2.1.1 RS 232C

Toto rozhraní bylo vyvinuto a normalizováno u americké asociace pro elektronický průmysl EIA (Electronic Industries Association) v roce 1962. Rozhraní RS 232C, bylo původně určeno pro koncové spojení datového zařízení (např. počítač) s komunikačním datovým zařízením (např. modem), časem RS 232C rozšířilo i mimo obor telekomunikační techniky. K výraznému rozšíření přispělo zejména užití v počítačích standardu PC, kde se pro něj vžilo označení jako sériový port.

RS 232C definuje celkem 20 signálů rozhraní a přiřazuje je konkrétním pozicím na konektoru s 25 kontakty. Typ konektoru není specifikován, ale nejčastěji se využívá konektor Cannon s 25 nebo 9 kontakty. Pro měřicí přístroje a osobní počítače se používá pouze 9 základních signálů.

2.1.2 RS 422A

Tato sběrnice vznikla odstraněním nevýhod rozhraní RS 232C, mezi které patřila zejména nízká komunikační rychlost, spojení pouze 2 zařízení a malou délkou spojení. Přenosové vedení může být až 1200m dlouhé při úměrně snížené přenosové rychlosti.

2.1.3 RS 485

RS 485 bylo vyvinuto a doporučeno EIA v roce 1983. Jde o rozšíření možností sběrnice RS 422A. V dnešní době je sběrnice RS 485 je asi nejrozšířenější průmyslovou sběrnici a tvoří fyzickou vrstvu (podle modelu ISO/OSI) několika průmyslových sběrnic, mezi které patří například Profibus, DIN,-Messbus atd. Podle standardu EIA není definován v RS485 počet vysílačů, ale jen počet přijímačů je omezen na 32 na jedné sběrnici. Signál je přenášen v bitové formě a maximální rychlost je okolo 10 Mbit/s. Přenos je realizován pomocí krouceného dvojdrátu, který patří mezi finančně nejméně nákladný a pro snížení rušení se používá ještě stínění.

Maximální délka vedení je 1200m v závislosti na okolních podmínkách.

tabulka 1: Srovnání parametrů RS 232C, RS 422A a RS 485

Rozhraní	RS 232C	RS 422A	RS 485
Počet vodičů pro jeden směr	1	2-symetrický	2-symetrický
Maximální počet vysílačů/přijímačů	1.I	1.X	32/32
Maximální délka vedení	15 m	1,2 km	1,2 km
Maximální přenosová rychlost	20 kb/s	10 Mb/s	10 Mb/s
Maximální napětí	±25 V	±6 V	±6V
Impedance vedení	3-7 Ω	100 Ω	54 Ω

2.2 CAN

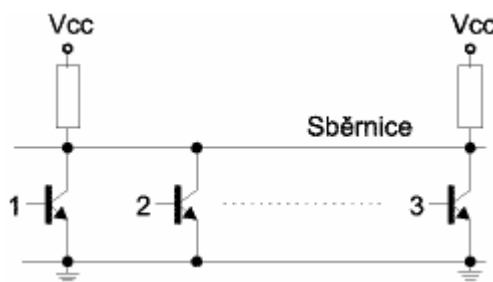
Controller Area Network (CAN) je sériový komunikační protokol, který byl původně vyvinut firmou Bosch pro nasazení v automobilech. Vzhledem k tomu, že přední výrobci integrovaných obvodů implementovali podporu protokolu CAN do svých produktů, dochází k stále častějšímu využívání tohoto protokolu i v různých průmyslových aplikacích. Důvodem je především nízká cena, snadné nasazení, spolehlivost, vysoká přenosová rychlost, snadná rozšiřitelnost a dostupnost potřebné součástkové základny.

V současné době má protokol CAN své pevné místo mezi ostatními fieldbusey a je definován normou ISO 11898. Ta popisuje fyzickou vrstvu protokolu a specifikaci CAN 2.0A. Později byla ještě vytvořena specifikace CAN 2.0B, která zavádí dva pojmy - standardní a rozšířený formát zprávy (lišící se v délce identifikátoru zprávy). Tyto dokumenty definují pouze fyzickou a linkovou vrstvu protokolu podle referenčního modelu ISO/OSI. Aplikační vrstva protokolu CAN je definována několika vzájemně nekompatibilními standardy (CAL/CANopen, DeviceNet, ...).

CAN byl navržen tak, aby umožnil provádět distribuované řízení systémů v reálném čase s přenosovou rychlostí do 1Mbit/s a vysokým stupněm zabezpečení přenosu proti chybám. Jedná se o protokol typu *multi-master*, kde každý uzel sběrnice může být *master* a řídit tak chování jiných uzlů. Není tedy nutné řídit celou síť z jednoho "nadřazeného" uzlu, což přináší zjednodušení řízení a zvyšuje spolehlivost (při poruše jednoho uzlu může zbytek sítě pracovat dál). Pro řízení přístupu k médium je použita sběrnice s náhodným přístupem, která řeší kolize na základě prioritního rozhodování. Po sběrnici probíhá komunikace mezi dvěma uzly pomocí zpráv (datová zpráva a žádost o data), a management sítě (signalizace chyb, pozastavení komunikace) je zajištěn pomocí dvou speciálních zpráv (chybové zprávy a zprávy o přetížení).

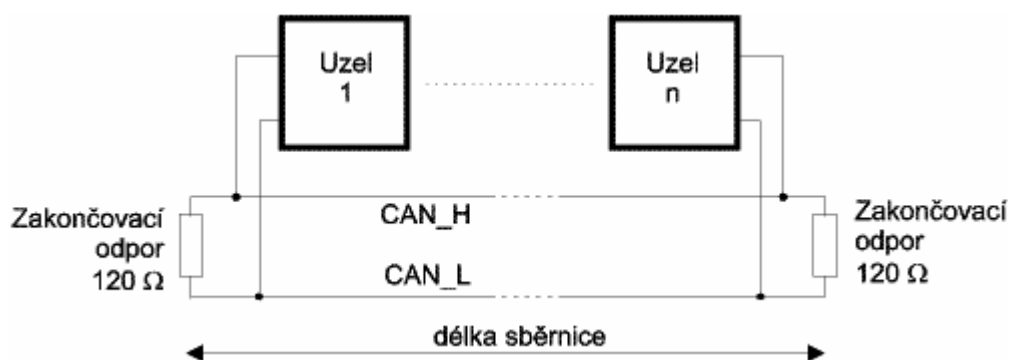
Zprávy vysílané po sběrnici protokolem CAN neobsahují žádnou informaci o cílovém uzlu, kterému jsou určeny, a jsou přijímány všemi ostatními uzly připojenými ke sběrnici. Každá zpráva je uvozena identifikátorem, který udává význam přenášené zprávy a její prioritu, nejvyšší prioritu má zpráva s identifikátorem 0. Protokol CAN zajišťuje, aby zpráva s vyšší prioritou byla v případě kolize dvou zpráv doručena přednostně a dále je možné na základě identifikátoru zajistit, aby uzel přijímal pouze ty zprávy, které se ho týkají (*Acceptance Filtering*).

Základním požadavkem na fyzické přenosové médium protokolu CAN je, aby realizovalo funkci logického součinu. Standard protokolu CAN definuje dvě vzájemně komplementární hodnoty bitů na sběrnici - *dominant* a *recessive*. Jedná se v podstatě o jakýsi zobecněný ekvivalent logických úrovní, jejichž hodnoty nejsou určeny a skutečná reprezentace záleží na konkrétní realizaci fyzické vrstvy. Pravidla pro stav na sběrnici jsou jednoduchá a jednoznačná. Vysílají-li všechny uzly sběrnice *recessive* bit, pak na sběrnici je úroveň *recessive*. Vysílá-li alespoň jeden uzel *dominant* bit, je na sběrnici úroveň *dominant*. Příkladem může být optické vlákno, kde stavu *dominant* bude odpovídat stav "svítí" a *recessive* stav "nesvítí". Dalším příkladem může být sběrnice buzená hradly s otevřeným kolektorem (Obrázek 2), kde stavu *dominant* bude odpovídat logická nula na sběrnici a stavu *recessive* logická jednička. Pak, je-li jeden tranzistor sepnut, je na sběrnici úroveň logické nuly (*dominant*) a nezáleží již na tom, zda je či není sepnutý i nějaký jiný tranzistor. Pokud není sepnut žádný tranzistor, je na sběrnici úroveň logické jedničky (*recessive*).



Obrázek 2: Příklad realizace fyzické vrstvy protokolu CAN

Pro realizaci fyzického přenosového média se nejčastěji používá diferenciální sběrnice definovaná podle normy ISO 11898. Tato norma definuje jednak elektrické vlastnosti vysílacího budiče a přijímače a zároveň principy časování, synchronizace a kódování jednotlivých bitů. Sběrnici tvoří dva vodiče (označované CAN_H a CAN_L), kde *dominant* či *recessive* úroveň na sběrnici je definována rozdílovým napětím těchto dvou vodičů. Dle nominálních úrovní uvedených v normě je pro úroveň *recessive* velikost rozdílového napětí $V_{diff} = 0V$ a pro úroveň *dominant* $V_{diff} = 2V$. Pro eliminaci odrazů na vedení je sběrnice na obou koncích přizpůsobena zakončovacími odpory o velikosti 120 Ohmů. Jednotlivá zařízení jsou na sběrnici připojena pomocí konektorů, nejčastěji jsou používány konektory D-SUB.



Obrázek 3: Principiální struktura sítě CAN podle ISO 11898

Přesné elektrické charakteristiky sběrnice jsou detailně popsány v normě ISO 11898, zde se omezíme na pouhé shrnutí několika nejdůležitějších a pro praktické využití užitečných faktů týkajících se fyzické vrstvy protokolu CAN. Přehled některých parametrů uvádí tabulka 2.

Na sběrnici může být čistě teoreticky připojeno neomezené množství uzlů, avšak s ohledem na zatížení sběrnice a zajištění správných statických i dynamických parametrů sběrnice norma uvádí jako maximum 30 uzlů připojených na sběrnici. Maximální délka sběrnice je pro přenosovou rychlost 1Mbit/s udána normou 40m. Pro jiné přenosové

rychlosti délku sběrnice norma neudává, avšak lze logickým úsudkem dojít k závěru, že pro nižší přenosové frekvence bude maximální délka sběrnice větší. Maximální délky sběrnice pro jinou přenosovou rychlost než 1Mbit/s uvedené v tabulka 2, jsou pouze informativní a závisí na mnoha parametrech (např. typu použitého kabelu).

tabulka 2: Přehled některých parametrů

Přenosová rychlost	125kBit/s - 1MBit/s
Doporučený počet uzlů v síti	max 30
Typická impedance vedení	120 Ohmů
10kBit/s	5200m
20kBit/s	2600m
50kBit/s	1340m
100kBit/s	640m
300kBit/s	200m
500kBit/s	112m
1MBit/s	40m

Linková vrstva protokolu CAN je tvořena dvěma podvrstvami - MAC a LLC:

- První z nich má na starosti přístup k médiu MAC (*Medium Access Control*) a jejím úkolem je provádět kódování dat, vkládat doplňkové bity do komunikace (*Stuffing/Destuffing*), řídit přístup všech uzlů k médiu s rozlišením priorit zpráv, detekce chyb a jejich hlášení a potvrzování správně přijatých zpráv.
- Druhou podvrstvou linkové vrstvy je LLC (*Logical Link Control*), která má za úkol provádět filtrování přijatých zpráv (*Acceptance Filtering*) a hlášení o přetížení (*Overload Notification*).

2.2.1 Řízení přístupu k médiu a řešení kolizí

Je-li sběrnice volná (ve stavu *Bus free*), může libovolný uzel zahájit vysílání. Zahájí-li některý uzel vysílání dříve než ostatní, získává sběrnici pro sebe a ostatní uzly mohou zahájit vysílání až po vyslání kompletní zprávy. Jedinou výjimku zde tvoří chybové rámce, které může vyslat libovolný uzel, detekuje-li chybu v právě přenášené zprávě.

Zahájí-li vysílání současně několik uzlů, pak přístup na sběrnici získá ten, který přenáší zprávu s vyšší prioritou (nižším identifikátorem). Identifikátor je uveden na začátku zprávy. Každý vysílač porovnává hodnotu právě vysílaného bitu s hodnotou na sběrnici a zjistí-li, že na sběrnici je jiná hodnota než vysílá (jedinou možností je, že vysílač vysílá *recessive* bit a na sběrnici je úroveň *dominant*), okamžitě přeruší další vysílání. Tím je

zajištěno, že zpráva s vyšší prioritou bude odeslána přednostně a že nedojde k jejímu poškození, což by mělo za následek opakování zprávy a zbytečné prodloužení doby potřebné k přenosu zprávy. Uzel, který nezískal při kolizi přístup na sběrnici musí vyčkat až bude sběrnice opět ve stavu *Bus free*, a pak zprávu vyslat znovu.

2.2.2 Zabezpečení přenášených dat a detekce chyb

Zprávy přenášené pomocí protokolu CAN jsou zabezpečeny několika mechanismy, které jsou v činnosti současně.

Monitoring - již zmiňovaná metoda, kdy vysílač porovnává hodnotu právě vysílaného bitu s úrovní detekovanou na sběrnici. Jsou-li obě hodnoty stejné, vysílač pokračuje ve vysílání. Pokud je na sběrnici detekována jiná úroveň než odpovídá vysílanému bitu, a probíhá-li právě řízení přístupu na sběrnici (vysílá se *Arbitration Field*), přeruší se vysílání a přístup k médiu získá uzel vysílající zprávu s vyšší prioritou. Pokud je rozdílnost vysílané a detekované úrovně zjištěna jinde než v *Arbitration Field* a v potvrzení přijetí zprávy (*ACK Slot*), je vygenerována chyba bitu.

CRC kód (*Cyclic Redundancy Check*) - na konci každé zprávy je uveden 15-ti bitový CRC kód, který je generován ze všech předcházejících bitů příslušné zprávy podle polynomu: $x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + 1$. Je-li detekována chyba CRC libovolným uzlem na sběrnici, je vygenerována chyba CRC.

Vkládání bitů (*Bit stuffing*) - vysílá-li se na sběrnici pět po sobě jdoucích bitů jedné úrovně, je do zprávy navíc vložen bit opačné úrovně. Toto opatření slouží jednak k detekci chyb ale také ke správnému časovému sesynchronizování přijímačů jednotlivých uzlů. Je-li detekována chyba vkládání bitů, je vygenerována chyba vkládání bitů.

Kontrola zprávy (*Message Frame Check*) - zpráva se kontroluje podle formátu udaného ve specifikaci a pokud je na nějaké pozici bitu zprávy detekována nepovolená hodnota, je vygenerována chyba rámce (formátu zprávy).

Potvrzení přijetí zprávy (*Acknowledge*) - je-li zpráva v pořádku přijata libovolným uzlem, je toto potvrzeno změnou hodnoty jednoho bitu zprávy (*ACK*). Vysílač vždy na tomto bitu vysílá úroveň *recessive* a detekuje-li úroveň *dominant*, pak je vše v pořádku. Potvrzování přijetí zprávy je prováděno všemi uzly připojenými ke sběrnici bez ohledu na zapnuté či vypnuté filtrování zpráv (*Acceptance Filtering*).

2.2.3 Signalizace chyb

Každý uzel má zabudována dvě interní počítadla chyb udávající počet chyb při příjmu a při vysílání. Podle obsahu počítadel může uzel přecházet, co se týká hlášení chyb a jeho aktivity na sběrnici, mezi třemi stavy (aktivní, pasivní, odpojený). Pokud uzel generuje příliš velké množství chyb, je automaticky odpojen (přepnut do stavu *Bus-off*). Z hlediska hlášení chyb tedy rozdělujeme uzly do následujících tří skupin:

Aktivní (*Error Active*) - tyto uzly se mohou aktivně podílet na komunikaci po sběrnici a v případě, že detekují libovolnou chybu v právě přenášené zprávě (chyba bitu, chyba CRC, chyba vkládání bitů, chyba rámce), vysílají na sběrnici aktivní příznak chyby (*Active Error Flag*). Aktivní příznak chyby je tvořen šesti po sobě jdoucími bity dominant, čímž dojde k poškození přenášené zprávy (poruší se pravidlo vkládání bitů).

Pasivní (*Error Passive*) - tyto uzly se také podílejí na komunikaci po sběrnici, ale z hlediska hlášení chyb, vysílají pouze pasivní příznak chyby (*Passive Error Flag*). Ten je tvořen šesti po sobě jdoucími bity *recessive*, čímž nedojde k destrukci právě vysílané zprávy.

Odpojené (*Bus-off*) - tyto uzly nemají žádný vliv na sběrnici, jejich výstupní budiče jsou vypnuty.

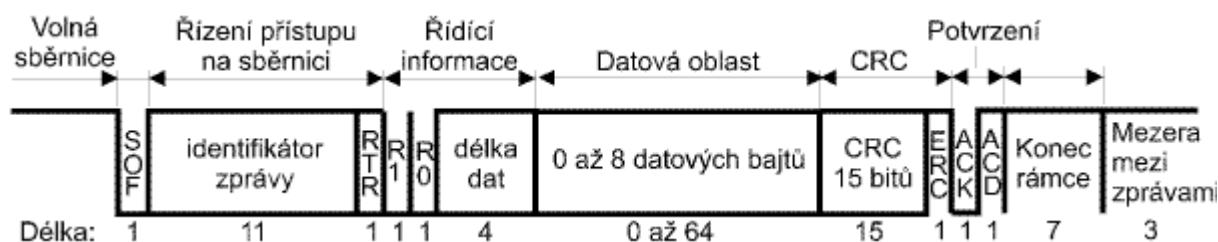
Specifikace protokolu CAN definuje čtyři typy zpráv:

- První dvě se týkají datové komunikace po sběrnici. Je to jednak datová zpráva, která představuje základní prvek komunikace uzlů po sběrnici, a dále pak zpráva na vyžádání dat, kdy uzel žádá ostatní účastníky na sběrnici o zaslání požadovaných dat. Datová zpráva umožňuje vyslat na sběrnici 0 až 8 datových bajtů. Pro jednoduché příkazy uzlům (např. příkazy typu vypni/zapni) není nutné přenášet žádné datové bajty (význam příkazu je dán identifikátorem zprávy), což zkracuje dobu potřebnou k přenosu zprávy a zároveň zvětšuje propustnost sběrnice, zvláště pak při silném zatížení. Zpráva na vyžádání dat je vysílána uzlem, který požaduje zaslání určitých dat. Odpovědí na tento požadavek je odeslání požadovaných dat uzlem, který tato data má k dispozici.
- Poslední dvě zprávy (chybová zpráva a zpráva o přetížení) slouží k managementu komunikace po sběrnici, konkrétně k signalizaci detekovaných chyb, eliminaci chybných zpráv a vyžádání prodlevy v komunikaci.

2.2.4 Datová zpráva (Data Frame)

Protokol CAN používá dva typy datových zpráv. První typ je definován specifikací 2.0A a je v literatuře označován jako standardní formát zprávy (*Standard Frame*), zatímco specifikace 2.0B definuje navíc tzv. rozšířený formát zprávy (*Extended Frame*). Jediný podstatný rozdíl mezi oběma formáty je v délce identifikátoru zprávy, která je 11 bitů pro standardní formát a 29 bitů pro rozšířený formát. Oba dva typy zpráv mohou být používány na jedné sběrnici, pokud je použitým řadičem podporován protokol 2.0B.

Vyslání datové zprávy je možné pouze tehdy, je-li sběrnice volná (stav *Bus Free*). Jakmile uzel, který má připravenou zprávu k vyslání, detekuje volnou sběrnici, začíná vysílat. Zda získá přístup na sběrnici či nikoliv, záleží na již popsaném mechanismu řízení přístupu k médium. Strukturu datové zprávy podle specifikace 2.0A ilustruje Obrázek 4.



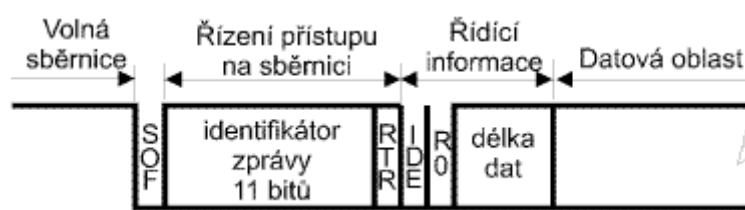
Obrázek 4: Datová zpráva podle specifikace CAN 2.0A

Význam jednotlivých částí datové zprávy podle specifikace 2.0A je následující:

- Začátek zprávy (SOF = *Start Of Frame*)- začátek zprávy, 1 bit *dominant*.
- Řízení přístupu na sběrnici (*Arbitration Field*)
 - Určení priority zprávy
 - Identifikátor zprávy - 11 bitů, udává význam přenášené zprávy
 - RTR bit (*Remote Request*) - 1 bit, příznak udává, zda se jedná o datovou zprávu nebo o žádost o vyslání dat. V datové zprávě musí být tento bit *dominant*, v žádosti o data *recessive*.
- Řídící informace (*Control Field*)
 - R0, R1 - rezervované bity
 - Délka dat - 4 bity, počet přenášených datových bajtů ve zprávě. Povolené hodnoty jsou 0 až 8.
- Datová oblast (*Data Field*) - datové bajty zprávy. Maximálně 8 bajtů je vysláno od MSB
- CRC (*CRC Field*):
 - 16 bitů, zabezpečovací CRC kód
 - CRC kód - 15 bitů
 - ERC (*CRC oddělovač*) - 1 bit *recessive*
- Potvrzení (*ACK Field*) - 2 bity
- ACK (bit potvrzení) - 1 bit
- ACD (oddělovač potvrzení) - 1 bit *recessive*
- Konec zprávy (*End Of Frame*) - 7 bitů *recessive*

- Mezera mezi zprávami (*Interframe Space*) - 3 bity *recessive*, odděluje dvě zprávy

Specifikace CAN 2.0B definuje dva formáty datové zprávy standardní a rozšířený. Standardní zpráva (*Standard Frame*) je převzat ze specifikace 2.0A, má délku identifikátoru zprávy 11 bitů. Jediným rozdílem je zde využití bitu R1 na indikaci, zda se jedná o rámec standardní nebo rozšířený. Zde se podle CAN 2.0B tento bit označuje IDE (*Identifier Extended*) a je *dominant* pro standardní formát a *recessive* pro rozšířený formát zprávy. Z obr. 4, který zobrazuje začátek rámce je vidět, že řízení přístupu na sběrnici (priorita zprávy) je dána opět 11-ti bity identifikátoru a hodnotou bitu RTR (*Remote Request*).



Obrázek 5: Začátek datové zprávy (standardní formát) podle specifikace 2.0B

Rozšířený rámec (*Extended Frame*) používá celkem 29 bitový identifikátor zprávy. Ten je rozdělen do dvou částí o délkách 11 (stejný identifikátor je použit ve standardním formátu) a 18 bitů (Obrázek 6). Bit RTR (*Remote Request*) je zde nahrazen bitem SRR (*Substitute Remote Request*), který má v rozšířeném formátu vždy hodnotu *recessive*. To zajišťuje, aby při vzájemné kolizi standardního a rozšířeného formátu zprávy na jedné sběrnici se stejným 11-ti bitovým identifikátorem, získal přednost standardní rámec. Bit IDE (*Identifier Extended*) má vždy *recessive* hodnotu. Bit (RTR) udávající, zda se jedná o datovou zprávu nebo žádost o data je přesunut za konec druhé části identifikátoru. Pro řízení přístupu k médiu jsou použity ID (11 bit), SRR, IDE, ID (18 bit), RTR. V tomto pořadí je určena priorita datové zprávy.



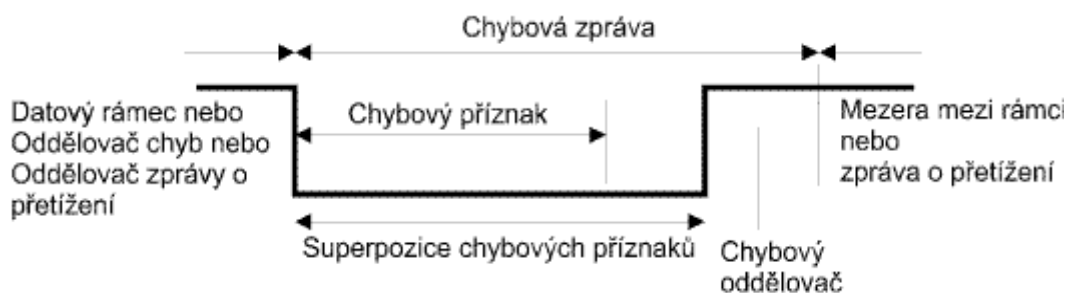
Obrázek 6: Začátek datové zprávy (rozšířený formát) podle specifikace 2.0B

2.2.5 Žádost o data (Remote Frame)

Formát žádosti o data je podobný jako formát datové zprávy. Pouze je zde RTR bit (pole řízení přístupu na sběrnici) nastaven do úrovně *recessive* a chybí datová oblast. Pokud nějaký uzel žádá o zaslání dat, nastaví takový identifikátor zprávy, jako má datová zpráva, jejíž zaslání požaduje. Tím je zajištěno, že pokud ve stejném okamžiku jeden uzel žádá o zaslání dat a jiný data se stejným identifikátorem vysílá, přednost v přístupu na sběrnici získá uzel vysílající datovou zprávu, neboť úroveň RTR bitu datové zprávy je *dominant* a tudíž má tato zpráva vyšší prioritu.

2.2.6 Chybová zpráva (Error Frame)

Chybová zpráva slouží k signalizaci chyb na sběrnici CAN. Jakmile libovolný uzel na sběrnici detekuje v přenášené zprávě chybu (chyba bitu, chyba CRC, chyba vkládání bitů, chyba rámce), vygeneruje ihned na sběrnici chybový rámec. Podle toho, v jakém stavu pro hlášení chyb se uzel, který zjistil chybu, právě nachází, generuje na sběrnici buď aktivní (šest bitů *dominant*) nebo pasivní (šest bitů *recessive*) příznak chyby. Při generování aktivního příznaku chyby je přenášená zpráva poškozena (vzhledem k porušení pravidla na vkládání bitů), a tedy i ostatní uzly začnou vysílat chybové zprávy. Hlášení chyb je pak indikováno superpozicí všech chybových příznaků, které vysílají jednotlivé uzly. Délka tohoto úseku může být minimálně 6 a maximálně 12 bitů.

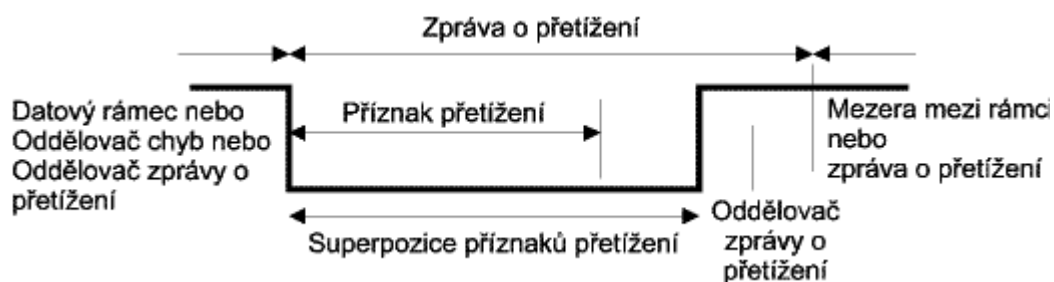


Obrázek 7: Chybová zpráva protokolu CAN

Po vyslání chybového příznaku vysílá každá stanice na sběrnici bity *recessive*. Zároveň detekuje stav sběrnice a jakmile najde první bit na sběrnici ve stavu *recessive*, vysílá se dalších sedm bitů *recessive*, které plní funkci oddělovače chyb (ukončení chybové zprávy).

2.2.7 Zpráva o přetížení (Overload Frame)

Zpráva o přetížení slouží k oddálení vyslání další datové zprávy nebo žádosti o data. Zpravidla tento způsob využívají zařízení, která nejsou schopna kvůli svému vytížení přijímat a zpracovávat další zprávy. Struktura zprávy je podobná zprávě o chybě, ale její vysílání může být zahájeno po konci zprávy (*End of Frame*), oddělovače chyb nebo předcházejícího oddělovače zpráv přetížení.



Obrázek 8: Zpráva o přetížení

Zpráva o přetížení je složena z příznaku přetížení (šest bitů *dominant*) a případné superpozice všech příznaků přetížení, pokud jsou generovány více uzly současně. Za příznaky přetížení následuje dalších sedm bitů *recessive*, které tvoří oddělovač zprávy o přetížení.

2.3 DeviceNet

Standard DeviceNet je otevřený síťový standard pro přenos dat v průmyslových systémech. Mezi výhody DeviceNetu patří, že umožňuje připojovat a odpojovat zařízení za chodu. Sdružuje rozvod datových signálů a napájení. Počet zařízení je maximálně 64. Přenosovou rychlost je možné volit v rozsahu 125/250/500kB.

Základem DeviceNetu je technologie CAN, tento standard obsahuje prostředky pro komunikaci typu žádost-odpověď, umožňuje fragmentovat dlouhé bloky dat a detekovat duplicitní zařízení.

DeviceNet standardizuje pouze protokol aplikační a fyzické vrstvy, zbytek je pak definován specifikací CAN. CAN obsahuje algoritmus řízení přístupu, k přenosovému médium a signalizaci, neobsahuje však definici protokolu aplikační vrstvy, způsob připojení k přenosovému médium a taky definici přenosového média.

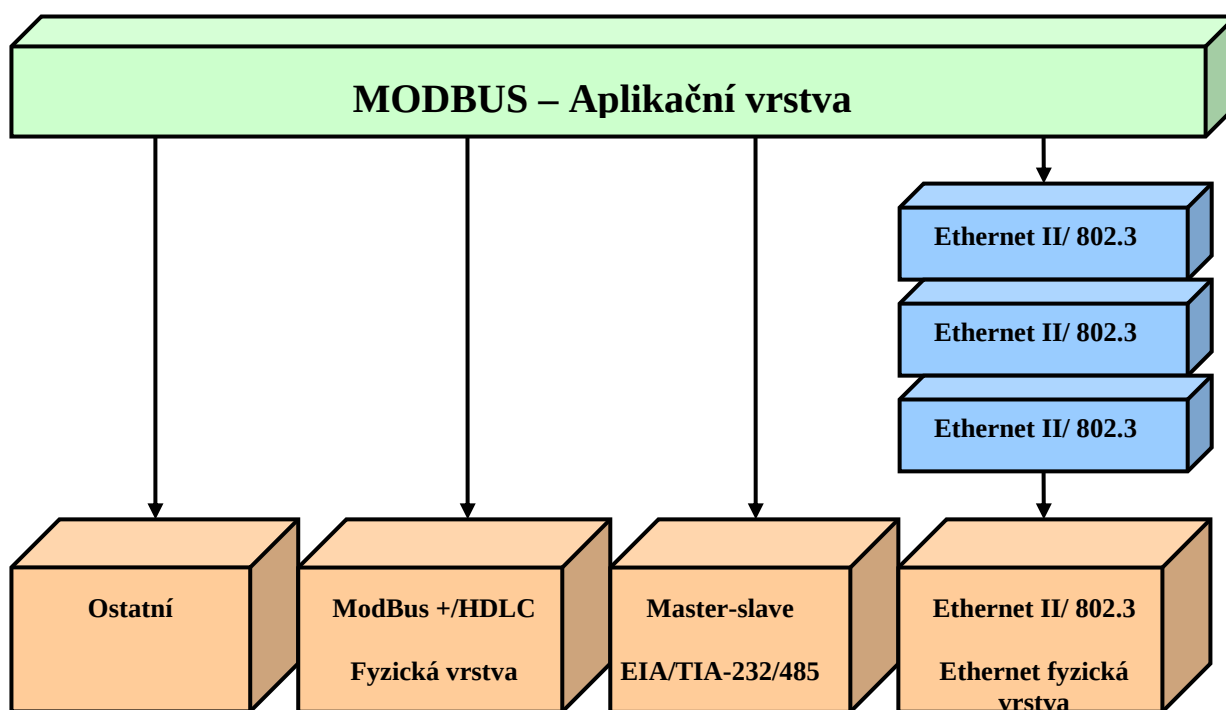
2.4 ModBus

MODBUS je otevřený protokol pro vzájemnou komunikaci různých zařízení (PLC, dotykové displeje, I/O rozhraní apod.), který umožňuje přenášet data po různých sítích a sběrnicích (RS-232, RS-485, Ethernet TCP/IP, MODBUS+ atd.). Komunikace funguje na principu předávání datových zpráv mezi klientem a serverem resp. masterem a slavem.

MODBUS je protokol pro přenášení zpráv pracující na aplikační hladině OSI modelu a poskytuje komunikaci na principu master-slave mezi zařízeními komunikujícími na různých typech sběrnic a sítích. Struktura standard byla vyvinuta firmou firmou Modicon v roce 1979 a postupně se z něho stal průmyslový standard. MODBUS je protokol typu žádost-odpověď a poskytuje přenos služeb specifikované tzv. kódem funkce. Protože je MODBUS plně otevřená architektura, umožňuje komunikaci po mnoha různých typech architektur sítí.

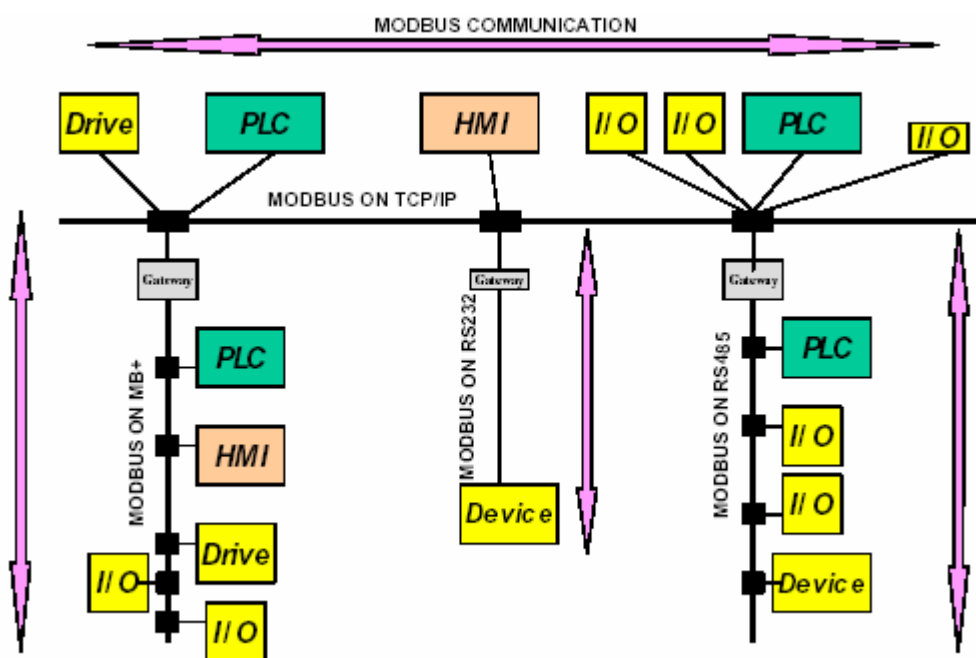
V současnosti je sběrnice Modbus začleněna do komunikace po sítích:

- TCP/IP Internet
- Asynchronní sériový přenos (RS 232, RS 422, RS 485, vlákno a rádiový přenos)
- Modbus PLUS vysokorychlostní síť



Obrázek 9: Struktura a rozsah současné verze protokolu MODBUS

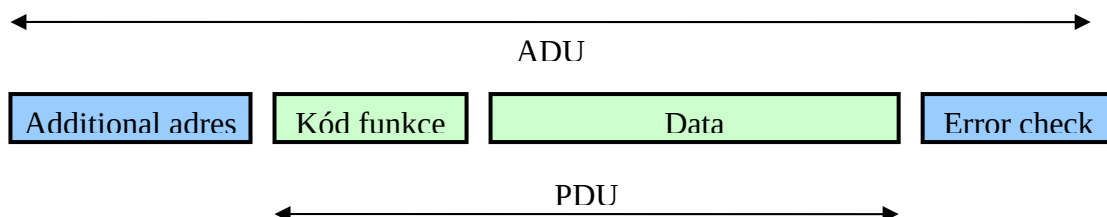
MODBUS na úrovni aplikační vrstvy OSI modelu poskytuje inicializaci a řízení pro velké množství průmyslových zařízení jako jsou například PLC, HMI (Human Machine Interface), Motion controllers, I/O zařízení atd. Při propojování různých typů sítí, po kterých MODBUS poskytuje komunikaci, je nutné použít brány (Gateway) na každém rozhraní dvou sítí.



Obrázek 10: Příklad struktury propojení ModBus s různými rozhraními

2.4.1 Aplikační protokol

ModBus protokol je definovaný jednoduchým PDU (Protocol Data Unit) na úrovni aplikační vrstvy OSI modelu a je tedy nezávislý na komunikaci spodních vrstev OSI modelu. Mapování MODBUS protokolu na specifickou sběrnici nebo síť pak může přidat další data do tzv. ADU (Application Data Unit). Zatímco PDU je stejný pro komunikaci na všech typech sítí, ADU již se může lišit v závislosti na typu sítě.



Obrázek 11: Struktura ModBus rámce

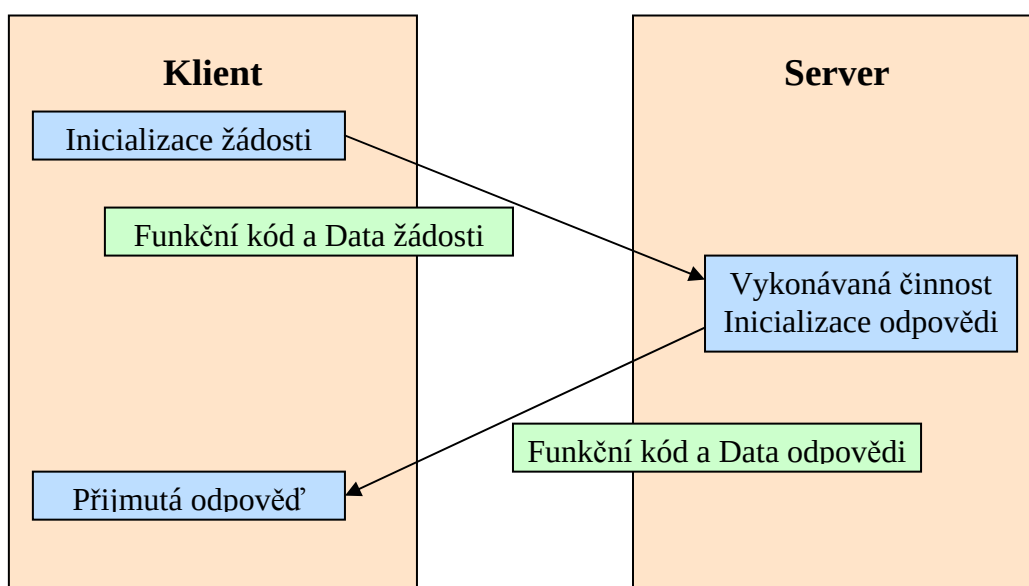
ModBus ADU se nastavuje klientem, který inicializuje datovou komunikaci. Kód funkce určuje serveru akci, kterou má s daty vykonat. Kód funkce má délku 1 bajtu, čili může obsahovat 1-255 různých funkcí. Jsou definovány i subfunkční kódy, které se předávají k funkčním kódům pro určení vícenásobných funkcí. Délka ADU je limitována pouze maximální délkou dat pro přenos po dané fyzické síti.

Komunikace probíhá na základě žádosti klienta na server, při kterém v PDU přenáší kód funkce a potřebná data. V případě, že v průběhu komunikace nenastane chyba, tak server po provedené akci vyšle odpověď zpět klientovi. Odpověď obsahuje původní kód funkce a data výsledku akce.

V případě že nastane chyba na úrovni serveru, tak odpověď obsahuje místo kódu funkce tzv. kód výjimky pro identifikaci chyby.

Protokol ModBus definuje tři typy PDU:

- ModBus Request PDU - mb_req_pdu - definuje funkční kód i data v žádosti o akci
- ModBus Response PDU - mb_rsp_pdu - definuje funkční kód akce a výsledná data v odpovědi
- ModBus Exception Response PDU - mb_excep_rsp_pdu - definuje kód vyjímky a data vyjímky v odpovědi o chybě



Obrázek 12: Ukázka bezchybné transakce dat mezi klientem a serverem

2.4.2 Kódování dat

Je využíváno tzv. Big-Endian reprezentace adresy dat. Ta umožňuje přenášet čísla o větším počtu bitů než odpovídá vysílanému jednomu bajtu. Prvním vysílaným bajtem je tzv. MSB (Most Significant Byte – nejvýznamější bajt).

Příklad dělení čísel podle Big-Endian reprezentace:

- 16 - bits: 0x1234 se rozdělí na: 0x12 a 0x34
- 32 - bits: 0x12345678 se rozdělí na: 0x12, 0x34, 0x56, 0x78

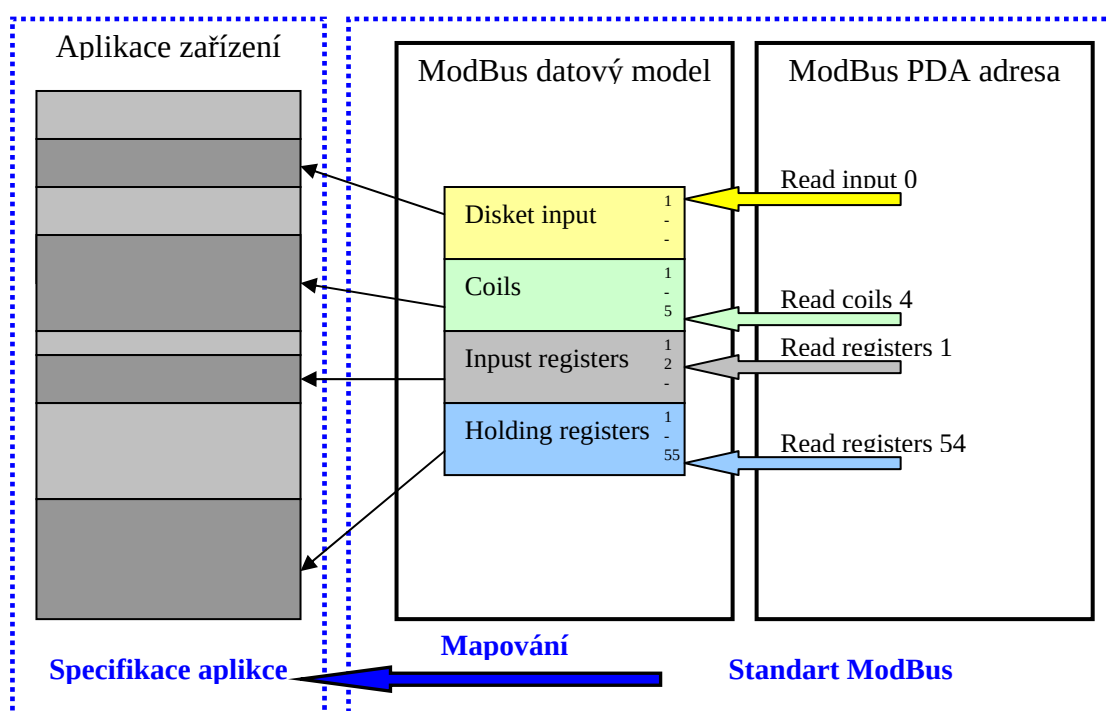
Datový model

Datový model protokolu může být rozdělen na sérii oblastí, kde každá má určité vlastnosti. Následující tabulka 3 druhy definovaných oblastí.

tabulka 3: Rozdělení oblastí datového modelu

Primární oblast	Typ objektu	Typ přístupu	Popis
Input Discrete	jeden bit	čtení	data dodaná IO zařízeními
Coils	jeden bit	čtení i zápis	data mohou být měněna aplikačním programem
Input Registers	16-bitové slovo	čtení	data dodaná IO zařízeními
Holding Registers	16-bitové slovo	čtení	data mohou být měněna aplikačním programem

Rozlišování mezi vstupy a výstupy a mezi bitově adresovanými a slovem adresovanými položkami nezahrnuje chování aplikací. Všechny manipulovaná data však musí být v aplikační paměti komunikujícího zařízení. Každou primární oblast lze rozdělit až na 65536 položek (2^{16}), s nimiž se pracuje podle dodaného funkčního kódu. Na absolutních adresách v paměti zařízení nezáleží, protože se pracuje pouze s relativními odkazy. Organizovat data v aplikační paměti komunikujícího zařízení lze v rámci oddělených bloků nebo jednoho bloku.



Obrázek 13: Datový model ModBus s oddělenými bloky a jeho adresování

Adresovací model dat

ModBus protokol definuje přesná adresovací pravidla v PDU:

- každý element v oblasti datového modelu je číslován od 1 do n
- mapování mezi MODBUS datovým modelem a zařízením je plně v kompetenci zařízení
- každý PDU může adresovat až 65535 elementů

Kódy funkcí

Kódy funkcí, které udávají typ akce k provedení dotazovanou stanicí, lze rozdělit na dvě skupiny:

- Veřejné kódy funkcí (Public Function Codes) - kódy definované standardem a organizacemi
- Uživatelsky definované kódy funkcí (User-defined Function Codes) - ty může zvolit a implementovat uživatel

Veřejné kódy funkcí obsahují čísla 1 až 65 a 111 až 127 a mimo speciálních funkcí udávají i akce čtení stavů jednotlivých vstupů, nastavení výstupů, čtení stavu

zařízení atd. Počet využívaných funkcí závisí na potřebách komunikujícího zařízení a i síti, po které se komunikuje.

2.4.3 ModBus pře sériovou linku

Pro komunikaci na sériové sběrnici, byl vytvořen protokol ModBus Seriál Line Protocol, který definuje pouze 2 vrstvu (linková) modelu ISO/OSI . Na první vrstvě (fyzické) je využito standartu rozhraní RS-232 nebo RS-485.

tabulka 4: Porovnáním ModBusu, přes sériovou linku s modelem ISO/OSI

Vrstva	Model ISO/OSI	Modbus aplikační protokol
7	Aplikační vrstva	
6	Prezentační vrstva	
5	Relační vrstva	
4	Transportní vrstva	
3	Síťová vrstva	
2	Linková vrstva	ModBus Serial Line Protocol
1	Fyzická vrstva	EIA/TIA – 485 (232)

Linková vrstva – Modbus Seriál Line Protocol

ModBus Serial Line Protocol umožňuje komunikaci mezi zařízeními typu Master-Slave. Na tuto linku může být připojena jen jedna řídicí stanice typu Master a až 247 stanic typu Slave. Komunikace je inicializována pouze stanicí Master. Stanice Slave mohou mezi sebou komunikovat jen za pomoci stanice Master.

Stanice Master může využívat dvou typů žádosti na stanici Slave:

- unicast mód – Stanice Master navazuje komunikaci pouze s jednou stanicí Slave, která má svou unikátní adresu (1-247)
- broadcast mód – Stanice Master vyšle žádost všem jednotkám Slave najednou (všechny Slave musí akceptovat adresu 0 určenou pro broadcast)

Struktura MODBUS rámce pro komunikaci po sériové lince je shodný s obecným schématem MODBUS rámce ADU. Adresa udává komunikující Slave stanici a přenáší se v obou směrech komunikace a to z důvodu, která jednotka právě odpovídá. Jako Error Check se využívá metody CRC nebo LRC.

- Linková vrstva popisuje mimo komunikace Master/Slave i tzv. přenosový mód:

- RTU (Remote Terminal Unit) mód - každý 8 - bitový bajt ve zprávě obsahuje 2 hexadecimální znaky
- ASCII mód - každý 8 - bitový bajt je vyslán jako 2 znaky ASCII

RTU mód

Na následujícím obrázku je zobrazen formát RTU módu s paritou. Každá zpráva obsahuje 11 bitů z toho je 8 bitů datových, 1bit je start, stop a parita. Komunikace může probíhat s paritou i bez parity. V druhém případě je pak paritní bit nahrazen dalším stop bitem. V tomto módu se využívá metody kontroly chybovosti CRC.

Star	1	2	3	4	5	6	7	8	Par	Ston
------	---	---	---	---	---	---	---	---	-----	------

Obrázek 14: Bitová reprezentace RTU Módu

Výhodou tohoto řešení je lepší rozložení desetiny znaků a tím větší datová propustnost při stejném boud rate v porovnání s ASCII módem. Každá zpráva však musí být vysílána v kontinuálním režimu.

ASCII mód

Na obrázku 8 je zobrazen formát ASCII módu s paritu. Každá zpráva zde obsahuje 8 datových bitů. Opět je možný i přenos bez parity. Pak je opět paritní bit nahrazen stop bitem. V tomto módu se pak využívá metody kontroly chybovosti LRC.

Star	1	2	3	4	5	6	7	Par	Ston
------	---	---	---	---	---	---	---	-----	------

Obrázek 15: Bitová reprezentace ASCII Módu

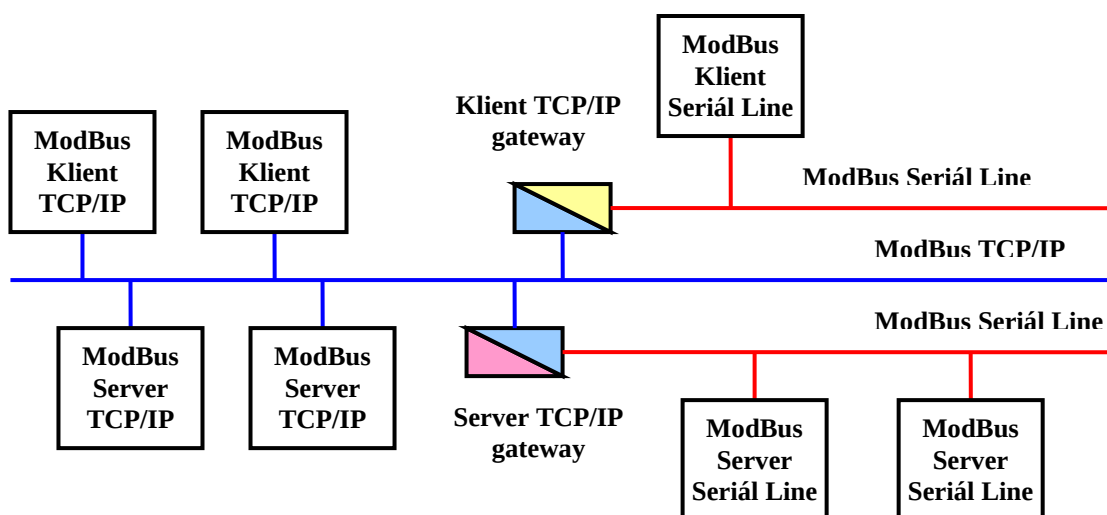
Výhodou tohoto módu je možnost použít až sekundového časového intervalu mezi vysláním jednotlivých znaků aniž by došlo k chybě přenosu.

2.4.4 MODBUS přes TCP/IP

Komunikační systém MODBUS přes TCP/IP zahrnuje dva rozdílné typy zařízení:

MODBUS Klient a Server napojené na síť TCP/IP. Propojovací zařízení jako jsou mosty (bridges), brány (gateways), směrovače (routers) pro propojení sítě TCP/IP s sítí tvořenou sériovou linkou a protokolem MODBUS po sériové lince.

Na obrázku Obrázek 16 je zobrazena hierarchie a architektura komunikačního systému ModBus přes TCP/IP. Při této komunikaci se MODBUS uplatňuje pouze na úrovni aplikační vrstvy.



Obrázek 16: ModBus TCP/IP architektura komunikace

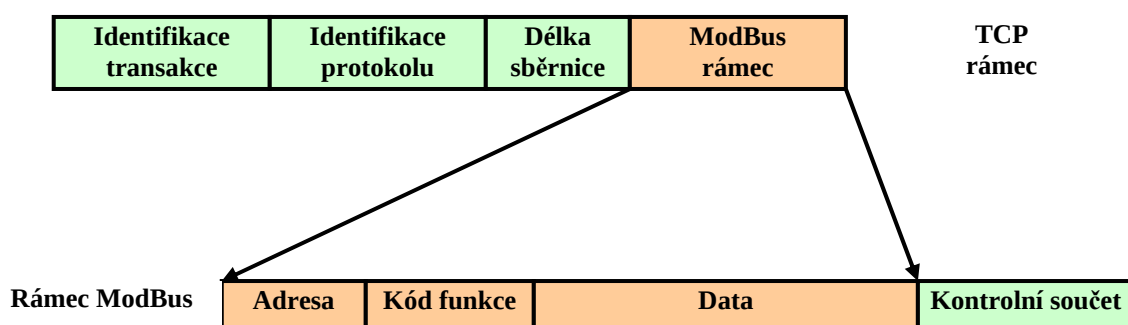
MODBUS on TCP

Základem komunikace protokolu MODBUS TCP/IP je ADU, která proti obecné definici obsahuje MBAP Header (MODBUS Application Protocol Header) místo bloku Additional address. MBAP označuje v TCP/IP protokolu přenos protokolu MODBUS.

MBAP Header obsahuje:

- Identifikátor transakce - identifikace žádostí a odpovědí transakce, využívá se pro párování transakcí
- Identifikátor protokolu - pro hodnotu 0 = MODBUS, využívá se pro multiplexování v síti
- Délka - číslo následujícího bajtu
- Identifikátor jednotky - identifikuje vzdálený Slave napojený na sériovou linku, využívá se pro vnitřní směrování mezi sítí TCP/IP a MODBUS sériovou linkou.

MODBUS rámec, tedy ADU, je dále zahrnut do TCP rámce dle obrázku 9. Všechny MODBUS TCP/IP ADU komunikují přes TCP přidělený port 502. Na nižších hladinách se již s rámci pracuje dle jednotlivých protokolů pro komunikaci v síti Ethernet, tedy nazávisle na MODBUS.



Obrázek 17: Princip zahrnutí rámce MODBUS do rámce TCP

2.4.5 ModBus PLUS (ModBus+)

Protokol pro linkovou vrstvu modelu OSI je nazýván jako ModBus PLUS. Byl vybit firmou Schneider Automation a to pro výměnu informací mezi firemními produkty. ModBus PLUS je otevřený protokol, který definuje komunikaci typu peer-to-peer s výměnu tokenů. Jedná se tedy o strukturu sítě Token Ring s fyzickým přístupem založeným na přenosové rychlosti 1MB/s.

Na síti může být až 64 adresovaných zařízení. Délka sítě lze pomocí opakovačů natáhnout až na 1800m. Minimální délka mezi jednotlivými uzly je 3m. Jako přenosové médium je využito stíněné kroucené dvojlinky. Síť může být strukturovaná a to pomocí mostů.

2.5 ASI

Standard ASI (Actuator Sensor Interface) vznikl počátkem devadesátých let. V současnosti standard udržován konsorciem výrobců, je však přihlášen ke standardizaci organizací IEC. Paralelně existuje (od roku 1991) i organizace uživatelů. Pro další rozvoj standardu je důležitá jeho podpora firmou Siemens, která má také nejširší nabídku produktů s tímto rozhraním.

Cílem standardu je podpora binárních akčních členů a senzorů na nejnižších úrovních procesní automatizace. Jde o sběrnici s Master-Slave řízením, současně lze připojit až 31 účastnických jednotek. Důležitým rysem standardu je možnost jejich napájení po sběrnici, která tak slouží pro napájení i pro přenos dat. Velkou výhodou je také prakticky libovolná topologie (liniová, hvězda, strom či jejich kombinace) a neexistence terminátorů na koncích vedení.

Charakteristické rysy sběrnice ASI:

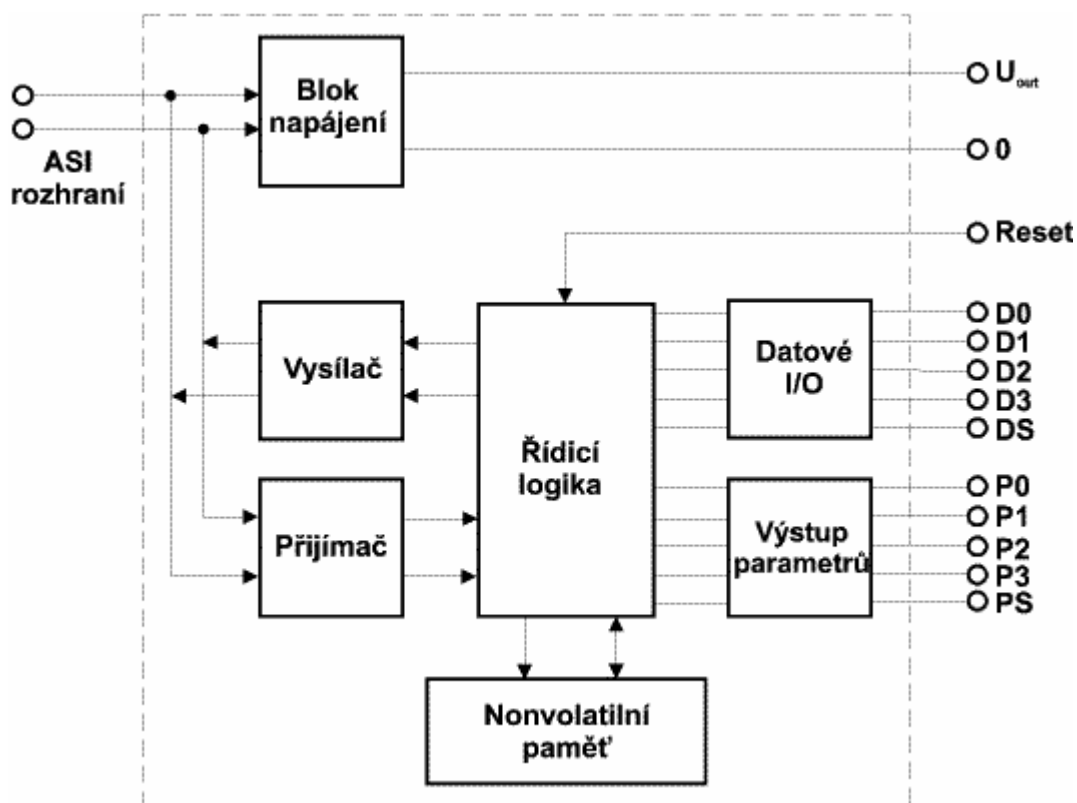
- Nestíněná dvoudrátová sběrnice s libovolnou topologií
- Přenos dat a napájení po jediném vedení
- Délka vedení je maximálně 100 m, s použitím opakovačů maximálně 300
- Řízení komunikace na principu Master - Slave
- Maximálně 31 účastnických stanic
- Maximálně 124 senzorů a 124 akčních členů
- Kódování dat kódem Manchester
- Zabezpečení telegramu paritou
- Vysoká rychlost komunikace (cykl sběrnice kratší než 5 ms)
- Velmi jednoduchá instalace

Vzhledem k využití jediného vedení pro přenos napájení i datových signálů je definice fyzické vrstvy zcela specifická. Nominální hodnota napájecího napětí je 24 V. Součástí napájecího zdroje je i standardem definovaná indukčnost, která se podílí na přenosu dat. Při vysílání mění jednotlivé stanice odběr proudu, což vyvolá vznik napěťových impulsů (kladných a záporných) na vedení. Tyto napěťové impulsy jsou superponovány na napájecím napětí, jejich výskyt a polarita jsou sledovány a vyhodnocovány.

Sběrníkové propojení všech stanic může mít prakticky libovolnou topologii (sběrnici, odbočná vedení, hvězdu, strom nebo jejich libovolnou kombinaci). Jediným požadavkem je, aby součet délek všech větví vedení nepřesáhl 100 m. Je-li požadována vyšší délka vedení, je nutno použít opakovač. Lze použít maximálně dva opakovače v jednom systému, celková délka vedení je tedy omezena na 300 m. Jak již bylo uvedeno výše, standard nevyžaduje použití terminátorů na koncích vedení, což dále zjednodušuje instalaci. Vzhledem ke zcela libovolné topologii sběrnice by použití klasických pasivních terminátorů stejně nebylo možné.

Pro realizaci ASI rozhraní, vysílač-přijímač, jsou k dispozici speciální obvody, realizující veškeré jeho funkce. Struktura obvodu je zobrazena na obrázku 1. Obvod poskytuje napájecí napětí pro senzory a akční členy, avšak při vyšších požadavcích na

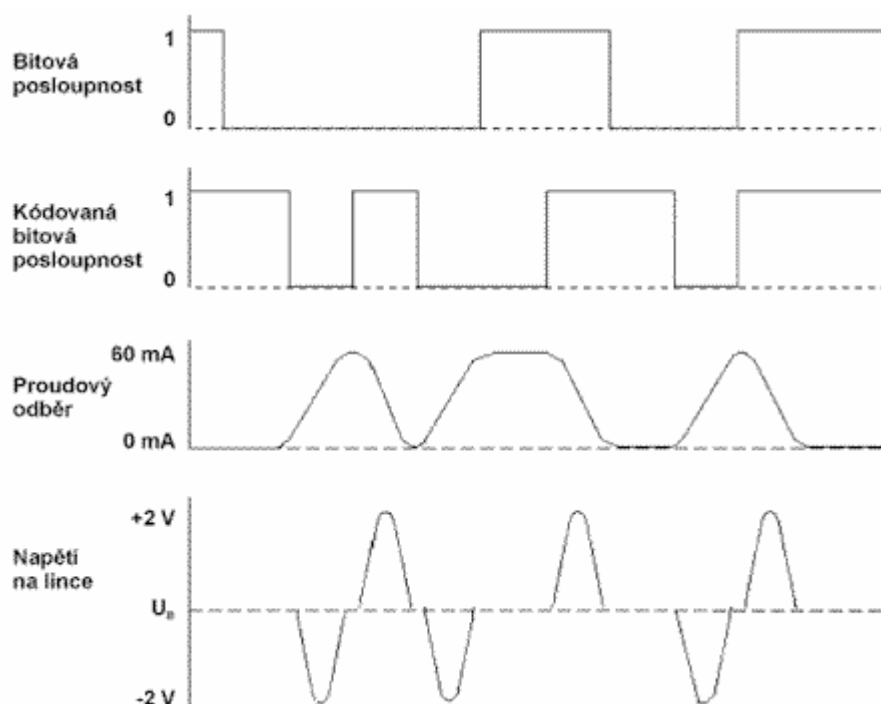
odebíraný proud musí být k dispozici zvláštní napájení. To se týká především účastnických stanic s akčními členy. Dále jsou k dispozici čtyři vstupně výstupní datové linky pro připojení senzorů nebo akčních členů a čtyři výstupy tzv. parametrů. Obě tyto čtveřice jsou doplněny vývody s funkcí "Strobe".



Obrázek 18: Struktura stanice napojené na sběrnici ASI

Na propojení sběrnice ASI se používá speciální plochý kabel s výstupkem na jedné straně, tím je zajištěna správné zapojení. Připojení další účastnické stanice se provádí tzv. krepováním. V takovém případě odpadá odizolování spojů a taky nejsou zapotřebí žádné konektory.

Kódování je definováno standardem a je zobrazeno na obrázku 2. Binární posloupnost je nejprve zakódována kódem Manchester (ve středu bitového intervalu bitu s úrovní log. 0 je v zakódovaném signálu přechod log. 1 -> log. 0, ve středu bitového intervalu bitu s úrovní log. 1 je v zakódovaném signálu přechod log. 0 > log. 1). Kódovaná bitová posloupnost je vysílačem převedena na změny odběru proudu, které se pak na lince projeví výše zobrazenými napěťovými impulsy. Délka bitového intervalu je cca 6 ms, což odpovídá přenosové rychlosti 166 Kb/s. Skutečný přenosový výkon je vzhledem k prodlevám v komunikaci poněkud nižší.



Obrázek 19: Způsob kódování signálu

Zabezpečení přenosu je prováděno následujícími prvky:

- Start bit - Prvním impulsem, který se vyskytne na sběrnici při přenosu telegramu musí být negativní impuls.
- Alternace impulsů - Vzhledem ke zvolenému typu modulace musí mít dva následující impulsy opačnou polaritu.
- Mezera mezi impulsy - mezi dvěma impulsy uvnitř telegramu smí být mezera o délce maximálně délky jednoho impulsu.
- Informační obsah - ve druhé polovině bitového intervalu musí být vždy impuls.
- Kontrola parity-v kódovém slově telegramu musí být sudý počet bitů hodnoty log.1.
- Stop bit - Posledním impulsem, kterým končí přenos telegramu po sběrnici, musí být kladný impuls.
- Délka telegramu - po ukončení telegramu stop bitem již na sběrnici nesmí následovat žádné další impulsy.

Linková vrstva používá řízení přístupu Master-Slave a specifický formát telegramů (zpráv). Jednotlivé účastnické stanice jsou cyklicky oslovovány stanicí řídící, celý cyklus trvá podle počtu účastnických stanic maximálně 5 ms pro 31 účastnických stanic. Při menším počtu dotazovaných účastnických stanic se doba trvání cyklu úměrně zkracuje.

2.6 Foundation Fieldbus

Standard průmyslové sběrnice označovaný jako Foundation Fieldbus, vyvinula nezisková organizace Fieldbus Foundation, sdružující významné výrobce z oblasti regulační a automatizační techniky a programového vybavení, ve snaze vnést stabilizaci do nepřehledné oblasti systémů pro průmyslovou komunikaci.

Foundation Fieldbus je číslicová sběrnice s poloduplexním seriovým přenosem, která je určena ke komunikaci mezi prvky regulačních obvodů. Architektura je navržena tak, aby byli zachovány základní výhody dosavadních analogových systémů s proudovou smyčkou 4-20mA a instalována může být nastávající kabeláž, která má výhody:

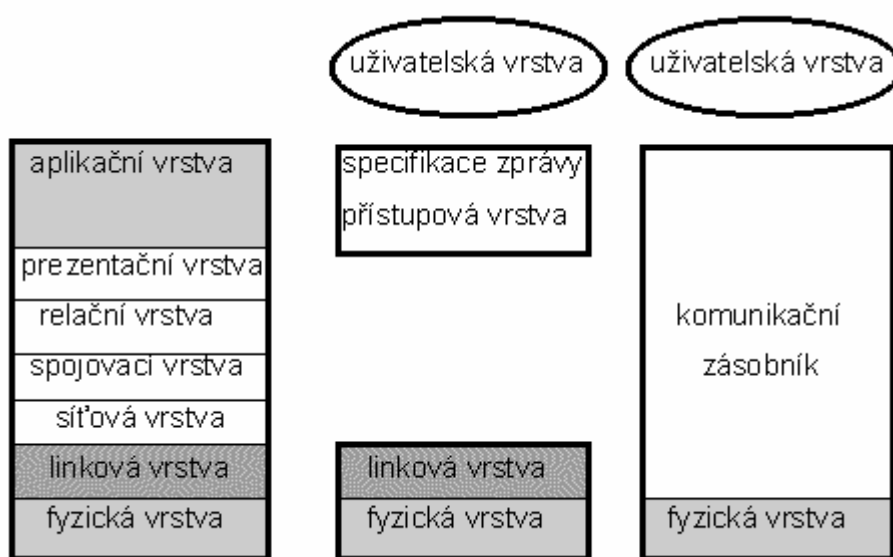
- standardizované rozhraní
- napájení přístrojů po komunikačním vedení
- použitelnost v prostředí s nebezpečím výbuchu

Tyto dosavadní výhody Foundation Fieldbus rozšiřuje o další:

- obousměrný přenos více parametrů po páru vodičů
- úspora rozvodů, na jeden pár může být napojeno více jednotek
- rychlá informace o výjimečných stavech
- implementace principů distribuovaného řízení snižuje nároky na výkonnost a počet řídících terminálů
- schopnost funkční diagnostiky jednotek a jejich zapojení

Architektura standardu vychází ze standardu modelu ISO/OSI a je dělena do tří částí:

- fyzická vrstva
- komunikační zásobník
- uživatelská vrstva



Obrázek 20: Definice vrstev standardu FF a vztah vůči modelu OSI

Standard FF definuje i uživatelskou vrstvu, kterou OSI již nezahrnuje, čímž je maximálně podporována záměnnost a kompatibilita produktů různých výrobců (přístrojů i nadstavbového software).

2.7 HART

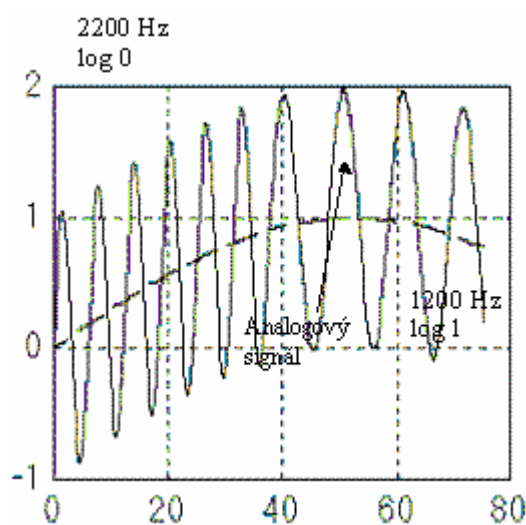
HART (Highway Addressable Remote Transducer) je rozšířeným a průmyslem akceptovaným standardním protokolem, umožňujícím oboustrannou číslicovou komunikaci se zařízeními propojenými dvou vodičovou proudovou smyčkou s analogovým přenosem signálů proudovými úrovněmi 4 až 20 mA.

Po původním vedení, směrem k řídicímu centru, je možné současně přenášet analogový údaj, který přenáší hodnotu měřené veličiny ve formě proudu v rozmezí 4 až 20 mA, tak i číslicovou informaci např. identifikační údaje senzoru, kalibrační a diagnostické údaje atd. Dále je umožněn přenos měřených údajů v analogové, tak i číslicové formě a tím vyloučit vznik možných chyb převodem D/A a A/D signálu.

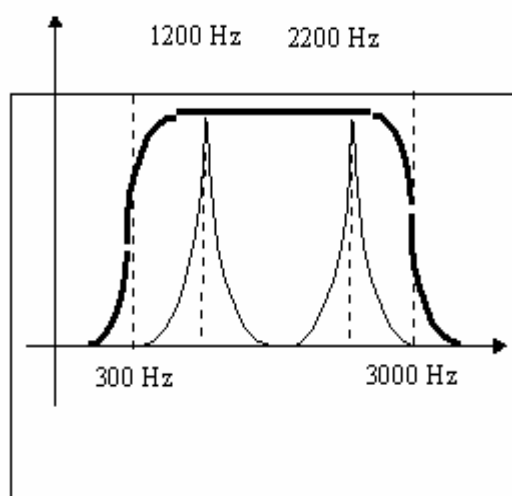
Z ekonomického hlediska je protokol velmi výhodný, protože počáteční investice je minimální. Zavedení protokolu komunikace a rozhraní HART může uživatel ještě během stávajícího provozu stávajícího analogového senzoru vyzkoušet jeho chování jako inteligentního senzoru. Navíc je protokol HART plně kompatibilní s protokolem ProfiBus.

Přenos číslicové informace po vedení proudové smyčky se realizuje superpozicí (např. transformátorovou vazbou) frekvenčně klíčováných signálů (FSK - Frequency Shift

Keying) na původní analogový proudový signál 4 až 20 mA. Parametry frekvenčního klíčování jsou dány standardem Bell 202 - logické 1 (H) odpovídá úsek signálu o kmitočtu 1200 Hz, logické 0 (L) pak signál o kmitočtu 2200 Hz. Amplituda superponovaného proudu je volena dostatečně nízká - 0,5 mA. Proces FSK probíhá bez skoků fáze takže za každé situace je střední hodnota superponovaného proudu nulová (obr.1). Střední hodnota analogového signálu je tedy nezávislá na přenosu digitální informace. Zařízení *master* vysílá signály ve formě napěťových úrovní, zatímco zprávy ze zařízení *slave* se vysílají proudovými signály. Proud smyčky se snadno převede na napěťovou úroveň takže *přijímací obvody* na straně *master* i *slave* pracují v napěťovém režimu.



Obrázek 21: Kódování logických úrovní (FSK Bell 202)



Obrázek 22 : Poloha spektra signálu HARTv telefonním hlasovém kanálu

tabulka 5: Úrovně komunikačních signálů

Vlastnost	Hodnota
Vysílání "master"	min 400 mV
	max 600 mV
Vysílání "slave"	min 0,8 mA
	max 1,2 mA
Minimální úroveň signálu slave (zátěž 230 ohmů)	184 mV
Maximální úroveň signálu master (zátěž 1100 ohmů)	1320 mV
Citlivost přijímače (příjem zaručen)	120 mV - 2,0 V
Práh přijímače (signál ignorován)	0 mV - 80 mV

K potlačení případných rušivých účinků číslicových komunikačních signálů na analogově vyhodnocovanou měřicí jednotkou je použito dolnofrekvenční propusti.

Přenos číslicových informací je organizován systémem master-slave. Zařízení ve funkci master obdrží nejméně 2x za sekundu informaci o stavu zařízení typu slave. Běžná doma odezvy je 500ms. S systému je umožněna možnost dvou zařízení master a to primárního a sekundárního.

Komunikace se děje v hvězdicové nebo sběrníkové topologii. Při hvězdicové topologii lze rychlost odezvy zvýšit dvojnásobně na 250 ms. Při sběrníkové topologii může probíhat pouze číslicová komunikace a smyčkový proud je udržován na minimální hodnotě 4mA a to z důvodu napájení senzorů.

Signály protokolu HART je snadno možné přenášet ve spektru telefonního hlasového signálu, jak plyne z jejich spektrálního rozložení na obrázku 2. Vznikají tím nové možnosti dálkového přenosu dat ze senzorů (např. monitorování plynovodů) v protokolu HART veřejnou telefonní sítí.

Zprávy jsou kódovány jako posloupnost 8-mi bitových bytů převedených do sériového tvaru obvodu HART. Je použita lichá parita.

tabulka 6: Kódování 1 bytu zprávy v protokolu HART

0	D00	D01	D02	D03	D04	D05	D06	D07	P	1
Start bit									bit parity	Stop bit

tabulka 7: Struktura zprávy

preambul e	znak start	adresa	příkaz	počet bytu	STATUS	DATA	kon.součet
------------	------------	--------	--------	------------	--------	------	------------

Preamble obsahuje 3 nebo více znaků FF a slouží pro synchronizaci frekvenčního diskriminátoru přijímacího modemu.

Adresa - obsahuje jednobytovou nebo 5-ti bytovou adresu právě komunikujících zařízení master a slave. Bit MSB adresového pole odlišuje primárního (MSB=1) a sekundárního mastera (nejčastěji ruční komunikátor).

Příkaz - obsahuje čísla od 0 do FF representující příkazy HART. Kód 254 je určeno pro expanzi: je-li následován dalším bytem, může být počet příkazů rozšířen na více než 256. Ve zprávě zařízení slave je přijatý příkaz zopakován.

Počet bytů - (ve zprávě) obsahuje také celé číslo, indikující počet bytů v částech STATUS a DATA, kontrolní součet není zahrnut. Slouží přijímajícímu zařízení ke kontrole úplnosti zprávy.

STATUS - se vyskytuje pouze v odpovědi zařízení slave a je tvořen dvěma byty: první indikuje případnou chybu komunikace, druhý stav zařízení (např. zařízení obsazeno, příkaz nelze interpretovat)

Krokem vpřed v programovém vybavení protokolu HART je zavedení programovacího jazyka vyšší úrovně - DDL (Device Description Language), s funkcí obdobnou jak ji známe z různých budičů - driverů periferních zařízení např. pro osobní počítače.

2.8 LIN (Local Interconnect Network)

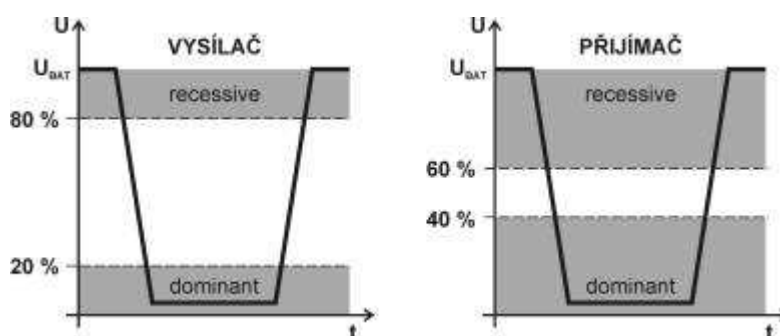
Sběrnice LIN (Local Interconnect Network) byla navržena pro použití v automobilové technice a to s ohledem na minimální cenové náklady spojené s její aplikací. Jedná se i sériovou asynchronní sběrnici používající jednovodičové spojení. Sběrnice LIN nemá nahradit sběrnici CAN, ale má uplatnění v oblastech, kde sběrnici CAN nevyužijeme v plném rozsahu nebo by nebyli z cenových důvodů napojeny na elektronický řídicí systém automobilu. Cena zřízení LINu je 2-3 levnější než CAN. Rozšíření standartu LIN napomáhá i že se jedná o otevřený standard sériové automobilové sběrnice třídy A.

První specifikace sběrnice LIN byla vytvořena v roce 1999 LIN Konsorciem, které v sobě sdružuje výrobce z automobilového průmyslu a výrobce elektroniky. První nasazení bylo uskutečněno v roce 2001.

Jedná se o sběrnici typu single-master/multiple-slave, kde jedno řídicí zařízení kontroluje komunikaci s jedním nebo více podřízenými zařízeními. Jednotlivá napojení na

jednovodičovou sběrnici tvoří drátový AND a komunikace probíhá maximální přenosovou rychlostí až 20 kbit/s. Ke generování komunikace lze použít hardwarových a softwarových prostředků běžného UART/SCI interface, přičemž podřízené jednotky (slave) nepotřebují k činnosti přesný krystalový generátor hodin, ale vystačí např. s RC oscilátorem. Synchronizaci pro komunikaci totiž provádí řídicí zařízení (master) na začátku každé komunikace. Koncepce budičů sběrnice vychází ze standardu ISO 9141 s vylepšeními v oblasti EMC. Data jsou zabezpečena kontrolním součtem. Hlavička je zabezpečena dvojicí paritních bitů.

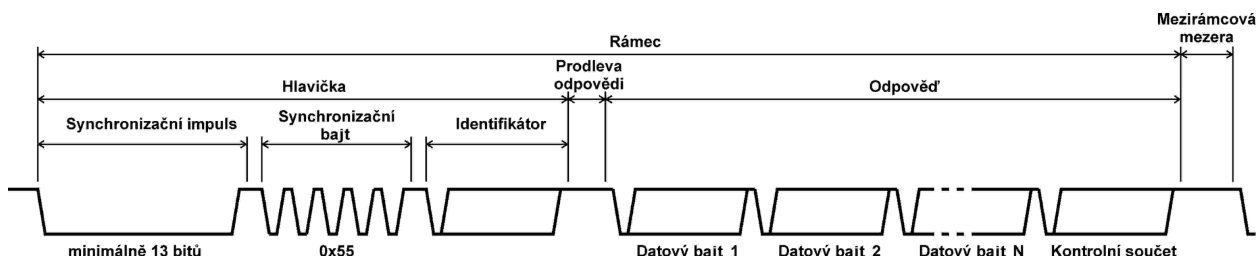
Fyzická vrstva byla odvozena od standardu ISO 9141. Princip sběrnice LIN je v použití jednoho vodiče pro obousměrnou komunikaci pomocí realizace funkce logického součinu prostřednictvím spínačů a rezistorů zapojených na LIN sběrnici v každém připojeném zařízení. Jsou definovány dvě vzájemně komplementární hodnoty stavů na sběrnici a to dominant a recessive. Hodnoty úrovní jednotlivých napětí jsou ukázány na obrázku 1. Spínače při sepnutí spojují sběrnici se zemí, a stačí aby byl sepnut alespoň jeden z nich a sběrnice přejde do stavu dominant, což představuje stav logické nuly. Rezistory zapojené mezi napájecí napětí a sběrnici pak na ní udržují, pokud není žádný spínač sepnutý, stav recessive, tedy logickou jedničku. Maximální délka sběrnice je 40m.



Obrázek 23: Rozhodovací úrovně vysílače a přijímače

LIN používá jednotný formát rámce zprávy, který slouží k synchronizaci, adresaci uzlů a k výměně dat mezi nimi. Formát rámce zprávy je na obrázek 2. Řídicí jednotka (master) začíná komunikaci, určuje přenosovou rychlost a vysílá hlavičku rámce zprávy. Ostatní jednotky, ale i jednotka master mohou vysílat odpověď složenou z datových bajtů a kontrolního součtu. Hlavička začíná synchronizačním impulsem a následným synchronizačním polem. Toto pole slouží k zesynchronizování podřízených jednotek (slave) na bitovou rychlost jednotky master. Tyto jednotky tak vystačí s jednoduchým

zdrojem časové základny v podobě RC oscilátoru, což má kladný vliv na cenu jednotek slave.



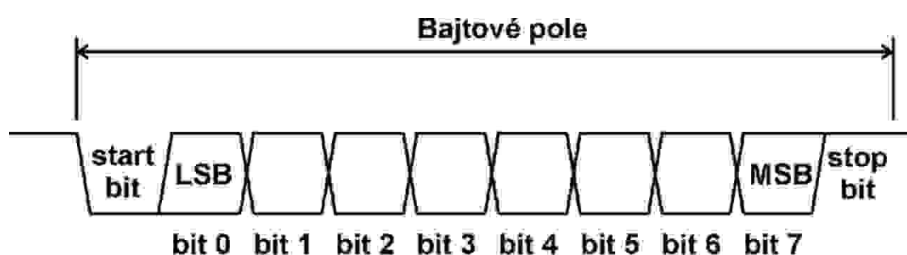
Obrázek 24: Formát rámce zprávy

Pro komunikaci lze použít maximální přenosovou rychlost 20 kbit/s, která představuje horní limit při použití jednodrátového přenosového vedení vzhledem k EMI. Minimální přenosová rychlost je stanovena na 1 kbit/s z důvodu předcházení problémům s praktickou implementací time-out period. Aby se usnadnila implementace do levných LIN zařízení, jsou k použití doporučeny 3 přenosové rychlosti viz tabulka 1.

tabulka 8: Doporučené přenosové rychlosti

Slow	2 400 bit/s
Medium	9 600 bit/s
Fast	19 600 bit/s

Jako zdroj signálu pro komunikaci lze použít standardní UART interface s bajtovým polem, viz obrázek 3. Jediná výjimka z 8N1 módu komunikace je synchronizační impuls, který vysílá pouze master. Doba trvání tohoto impulsu odpovídá vyslání minimálně 13 bitů a je většinou generován softwarově. Tento vzor je jednoznačně identifikovatelný protože jeho délka je větší než jakákoliv standardní sekvence na sériovém kanále. Začátek zprávy je tak bezpečně rozpoznatelný a poskytuje podřízeným jednotkám dostatek času pro zachycení komunikace na sběrnici v libovolném stavu podřízené jednotky.



Obrázek 25: Základní bajtové pole

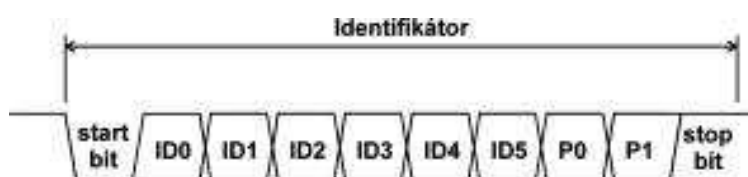
Posledním blokem hlavičky je identifikační pole obrázek 4. Tento identifikátor v sobě nese informaci o odesílateli, příjemci či příjemcích a délku datové části. Je

zabezpečen pomocí dvou paritních bitů a to bitu 6 a bitu 7. Jejich hodnoty jsou získány jako XOR kombinace významových bitů identifikátoru podle rovnic (1) a (2)

$$P_0 = ID_0 + ID_1 + ID_2 + ID_4 \quad (1)$$

$$P_1 = ID_1 + ID_3 + ID_4 + ID_5 \quad (2)$$

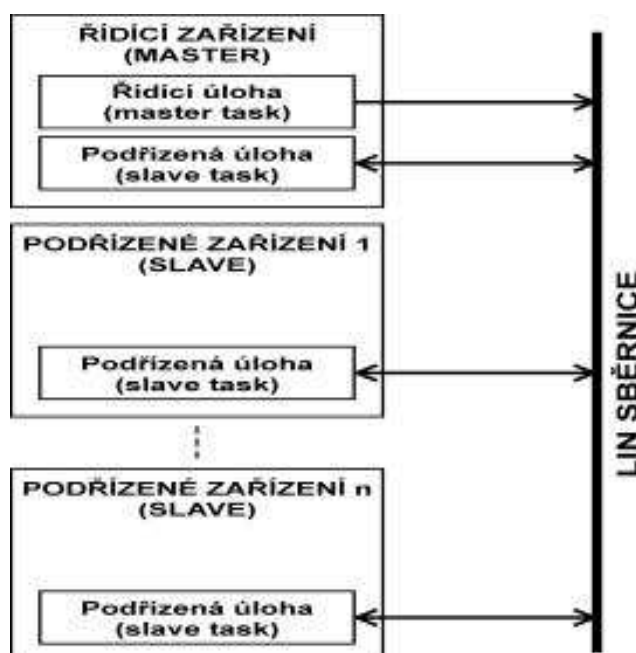
Vlastní datová část přenosu je pak zabezpečena invertovanou hodnotou součtu s přenosem přes tyto bajty. V nové verzi LIN protokolu je pak možnost započítat do tohoto součtu ještě i bajt identifikátoru a tím dále zvětšit zabezpečení přenosu proti chybám.



Obrázek 26: Rozmístnění jednotlivých bitů identifikátoru

Z uvedeného vyplývá, že je možné rozlišit kombinace 6-ti bitů tedy 64 různých identifikátorů.

Uspořádání struktury sběrnice z pohledu komunikace je zobrazeno na Obrázek 27. Řídící úloha v řídicím zařízení určuje veškeré dění na sběrnici prostřednictvím vysílání hlavičky. V té je obsažena veškerá informace o tom jaký druh komunikace bude následovat. Ostatní úlohy v podřízených zařízeních i v řídicím zařízení tuto hlavičku sledují a podle jejího obsahu případně přijímají nebo vysílají datovou část rámce.



Obrázek 27: Komunikační struktura sběrnice

Ve skutečnosti mohou podřízené úlohy vykonávat i další akce. Úplný výčet činností, jak pro řídicí zařízení tak pro podřízené zařízení, je uveden v následující části:

LIN Řídicí zařízení (Master) - řídí veškerou komunikaci na sběrnici:

- Definuje přenosovou rychlost
- Vysílá synchronizační impuls, synchronizační pole, identifikátor
- Monitoruje a potvrzuje data pomocí kontrolního součtu
- Přepíná podřízené jednotky do sleep módu a znovu je probouzí
- Reaguje na signál probuzení od podřízených jednotek.

LIN Podřízené zařízení (Slave) - Na sběrnici může být připojeno až 16 podřízených zařízení. Řídicí jednotka též obsahuje podřízenou úlohu (slave task) jako ostatní podřízená zařízení (slaves). Podřízené zařízení má na starosti:

- Čekání na synchronizační impuls
- Synchronizuje se podle synchronizačního pole
- Podle identifikátoru dělá jednu z následujících činností:
 - Nereaguje
 - Přijímá data
 - Vysílá data
 - Kontroluje nebo posílá kontrolní součet.

2.9 M-Bus

Standard M– Bus (z anglického Meter– Bus) je určen pro aplikace sběru dat z měřičů odběru nejrůznějších médií. Je vyvíjen ve spolupráci University Paderborn a firmy Texas Instruments Německo. Standard je zatím ve vývoji, nicméně je natolik ustálen, že na jeho základě lze navrhovat jednotlivé prvky a budovat celé systémy. Zde uvedený popis vychází z verze 4.8 standardu M– Bus, která definuje všechny použité protokolové vrstvy včetně aplikačních protokolů.

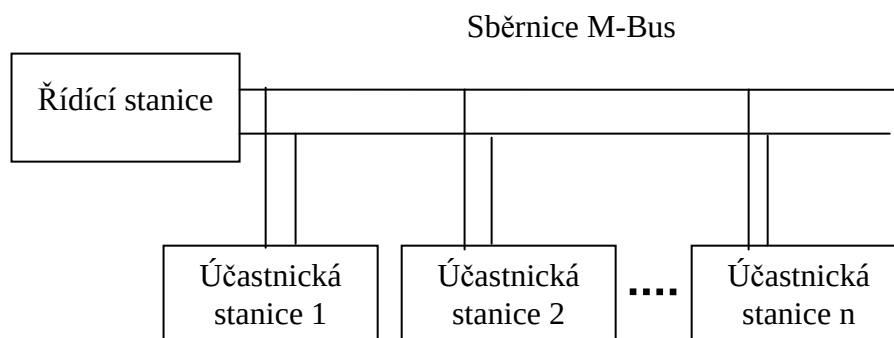
2.9.1 Základní charakteristika

Vzhledem k relativně úzké a poměrně specializované aplikační oblasti jsou na M-Bus kladeny specifické požadavky:

- Musí zajistit propojení velkého množství zařízení (řádově několika set) na vzdálenost až několika kilometrů.
- Přenos dat musí být kvalitně zabezpečen proti chybám.

Na druhé straně je typickou vlastností aplikace nepříliš časté odečítání naměřených hodnot s nízkými nároky na odezvy v reálném čase. Všechny protokolové vrstvy lze řešit softwarově a to díky malé přenosové rychlosti 9600Bd a nízkým požadavkům měřičů. Charakteristické rysy standardní konfigurace lze shrnout do následujících bodů:

- Speciální implementace fyzické vrstvy
- Galvanicky oddělené rozhraní
- Možnost napájení účastníků po sběrnici
- Dvoudrátové vedení s délkou až několik kilometrů
- Řízení komunikace na principu Master - Slave
- Bez implementace síťové vrstvy maximálně 250 účastníků
- Asynchronní přenos znaků, 8 bitů dat, sudá parita
- Přenosová rychlost 300 až 9600 Bd
- Zabezpečení datového bloku pomocí kontrolního součtu



Obrázek 28: Struktura sběrnice M-Bus

2.10 Measurement Bus

Measurement Bus vznikl v roce 1986 a v praxi je používán od roku 1990. Odpovídá standardu DIN 66348, část 2, který definuje fyzickou a linkovou vrstvu ISO-OSI modelu. Jde o sběrnici s Master-Slave řízením, definice její fyzické vrstvy odpovídá standardu RS-485. Na sběrnici je tedy možno připojit až 32 jednotek s typickou přenosovou rychlostí 9600 Bd. Norma však umožňuje použití přenosových rychlostí až do 1 MBd. Pro vyšší vrstvy (aplikační) ISO-OSI modelu je doporučen standard DIN 66348, část 3. Typickou aplikací je potom v závěru uvedený standard EPSI pro automatizaci čerpacích stanic.

2.10.1 Základní charakteristika

Tato norma byla vytvořena především pro datové přenosy v oblasti jednoduché automatizace a sběru dat. Značnou výhodou tohoto standardu je jeho snadná implementovatelnost v libovolném procesoru, vybaveném pouhým asynchronním sériovým rozhraním (UARTem), vše ostatní lze realizovat programově. Jelikož je toto rozhraní běžnou součástí prakticky všech mikrokontrolerů, další nezanedbatelnou výhodou standardu je jeho levná implementace. Charakteristické rysy lze shrnout do několika následujících bodů:

- Sběrnice dle EIA RS-485
- Délka datového bloku maximálně 128 znaků
- Řízení komunikace na principu Master - Slave
- Galvanicky oddělené plně duplexní rozhraní
- Čtyřvodičová sběrnice se stíněným kabelem s kroucenými páry
- Délku vedení je doporučeno omezit asi na 500m, v praxi je závislá na přenosové rychlosti
- Maximálně 32 účastnických stanic včetně stanice řídící
- Asynchronní přenos znaků, 7 bitů dat, sudá parita
- Přenosová rychlost do 1 Mbit/s, typicky do 9600 Bd
- Zabezpečení datového bloku pomocí příčné parity

2.11 PROFIBUS

Sběrnice PROFIBUS (zkratka PROcess FIeld BUS), byla vyvinuta v roce 1990 v SRN a je definována v normě DIN 19 245. Sběrnice využívá sběrnicovou topologii a fyzická vrstva je definována normou pro sběrnici RS-485. V současnosti je využíván jako přenosové médium i optických kabelů. Přístupová metoda může být:

- Token Passing – sběrnice si postupně předávají pověření v logickém kruhu
- Master-Slave - jedna stanice master řídí činnost celé sběrnice. Komunikace dvou zařízení slave, probíhá pomocí řídicího zařízení master.
- Hybridní – této metody se využívá, pokud potřebujeme více řídicích zařízení master. Tato metoda je kombinací předchozích dvou metod.

V profibusu jsou definována vrstvy 1,2 a 7 z referenčního modelu ISO/OSI a základní charakteristikou první vrstvy je:

- Délka sběrnice 1200 m bez opakovací s opakovací max. 4800m.
- Přenosová rychlost sběrnice je v rozmezí 9,6-500kb/s
- Počet účastníků je 32 a s použitím opakovací max. 127
- Přenosovým médiem je kroucený dvoudrát s galvanickým oddělením nebo optický kabel.

Linková vrstva se nazývá Fieldbus Data Link (FDL) a obsahuje prostředek Medium Access Kontrol (MAC), který určuje která stanice smí přenášet data. Jeho úkolem je také zajistit, aby vysílala pouze jedna stanice. Linková vrstva také zajišťuje vysílání dat jako službu 7.vrstvy a pracuje bez přímého spojení a to buď se zpráva vysílá všem stanicím nebo pouze určité skupině stanic.

Aplikační vrstva poskytuje jednotlivé služby nezbytné pro realizaci komunikace z hlediska uživatele.

V současné době existují tři, navzájem kompatibilní, varianty komunikačního standardu:

- Profibus-DP – je určen pro rychlý přenos na úrovni snímačů a akčních členů. Používá pouze dvě vrstvy a to fyzickou a linkovou.

- Profibus-FMS – Je univerzální sběrnice pro komunikaci vyšších inteligentních řídicích prvků. Je kladen větší důraz na funkčnost a poskytované služby než na rychlost komunikace a používá všechny vrstvy – fyzickou, linkovou a aplikační.
- Profibus-ISP – je určen pro řízení procesů, kde je požadována velká spolehlivost a bezpečnost.

Všechny uvedené modifikace mohou pracovat společně na přenosovém médiu. Navíc v jednom zařízení mohou pracovat dva protokoly najednou.

2.12 Interbus-s

Interbus-S je otevřená sběrnice pro komunikaci a řízení v reálném čase mezi snímači a akčními členy. Svůj název dostala pro rychlost řízení akčních členů.

Topologie je modifikovaná kruhová sběrnice, protože schéma připomíná stromovou topologii a Interbus-S je kompromisem mezi kruhovou a stromovou topologií. Přenosový cyklus je řešen jako v kruhové síti, zatímco fyzické připojení je v podstatě zdvojená přenosová cesta stromové topologie. Celým systémem je svisle vedena páteřová sběrnice, od které odbočují lokální nebo rozsáhlé boční sběrnice. Sběrnice Interbusu-S tvoří dvojce vodičů procházející všemi stanicemi.

Velkou výhodou kruhového přenosu je jednoduchá diagnostika funkčnosti systému, která je prováděna řídicí stanicí. Nevýhodou však je, že při poruše jedné stanice je znemožněna komunikace celého systému.

Technické parametry jsou:

- Přenosovým médiem je dvou drát nebo optický kabel.
- Páteřní sběrnice může mít až 13 km s opakovací každých maximálně 400m.
- Maximálně 256 stanic v logickém kruhu.
- Přístupová metoda je master-slave

2.13 Wi-fi

Bezdrátové sítě existují přibližně od roku 1992, ale v té době chyběl jakýkoliv standard. V roce 1997, byl přijat standard bezdrátových sítí IEEE 802.11, jimiž jsou moderní WLAN sítě definovány a standardizovány.

Wi-fi jsou definovány normami IEEE 802.11a/b/g. Omezení je, že zařízení pracující se standardem 802.11a nelez propojit s 802.11b/g. V Evropě není povolen standard 802.11a.

Wi-fi sítě pracují v rozsahu frekvenčního pásma 2,4-2,4835GHz, tedy v bezlicenčním pásmu a je někdy označované jako ISM. Kromě tohoto pásma je pro Wi-fi sítě vyhrazeno pásmo 5GHz.

tabulka 9: Povolené rádiové frekvence pro Wi-fi v pásmu 2,4GHz

Region	Frekvenční rozsah	Počet kanálů
USA	2,4000-2,4835	79
Evropa	2,4000-2,4835	79
Francie	2,4465-2,4835	27
Španělsko	2,445-2,475	35
Japonsko	2,471-2,497	23

Popis jednotlivých standardů IEEE 802.11:

- 802.11a – WLAN v pásmu 5GHz s přenosovou rychlostí až 54MB/s
- 802.11b – WLAN v pásmu 2,4GHz s přenosovou rychlostí až 11MB/s
- 802.11c – Dfinice procedur pro síťové mosty. Ve skutečnosti to s WLAN má jen málo společného, jde ale o užitečný standard pro přístupové body.
- 802.11d – Mezinárodní harmonizace. Se vznikem standardu 802.11 se ukázalo, že je potřeba mezinárodní kooperace a harmonizace. Zejména pásmo 5 GHz se používá v mnoha státech různě a bylo třeba tomu standardizaci přizpůsobit tak, aby nevycházela vstříc pouze potřebám USA a Japonska.
- 802.11e – Rozšíření MAC pro QoS. Zkratka QoS označuje službu Quality of Service zajišťující vyrovnanou kvalitu služby důležitou například pro multimédia. Zjednodušeně řečeno je potřeba, aby (když někdo v bezdrátové síti telefonuje nebo pořádá videokonferenci) trvalý tok jeho dat měl přednost před lidmi, kteří například jen stahují poštu a chvilkový výpadek naprosto nepoznají, zatímco v hlasu nebo videu by byl hodně poznat.

- 802.11f – Inter Access Point Protocol (IAPP) - Stávající specifikace 802.11 nezahrnují standardizaci komunikace mezi jednotlivými přístupovými body pro zajištění bezproblémového roamingu, tedy přechodu uživatele od jednoho přístupového bodu k druhému
- 802.11g – Zvýšení rychlosti v pásmu 2,4 GHz na 20 Mbit/s se zpětnou kompatibilitou s 802.11b.
- 802.11h – Změny v řízení přístupu k spektru 5GHz, které by měly reflektovat připomínky regulátorů evropských zemí tak, aby bylo možno sítě v pásmu 5 GHz využívat i mimo budovy.
- 802.11i – Zlepšení bezpečnosti v 802.11 bezdrátových sítích vylepšením autentifikačního a šifrovacího algoritmu.
- 802.11j – Práce na alokaci nových frekvenčních rozsahů pro multimediální služby bezdrátových sítí.
- 802.11k – Tento projekt má definovat měření a správu radiových zdrojů tak, aby vyhovovaly novým vysokofrekvenčním radiovým sítím.
- 802.11s – Mesh standard pro samoorganizující se WiFi sítě.

Zabezpečení Wi-fi:

- SSID (Service Set ID), kterým se označují přístupové body (Access Point, AP), představuje nejnižší stupeň bezpečnosti pro komunikaci ve WLAN.
- Protokol WEP (Wired Equivalent Privacy) – jedná se o volitelný doplněk k 802.11b pro řízení přístupu a zabezpečení dat. WEP pracuje na symetrickém principu kde se pro šifrování a dešifrování používá stejný algoritmus i totožný statický klíč.

3 PRAKTICKÁ ČÁST

V této části diplomové práce bude popsána rotační vakuová odparka, její snímání a řízení pomocí řídicího systému. Dále program Control Web 2000, ve kterém bude následně vytvořena jednoduchá aplikace pro monitorování, vizualizaci a řízení rotační vakuové odparky a také nastavení konfiguračních souboru Control Webu 2000, pro komunikaci přes sběrnici ModBus.

3.1 Popis a schéma rotační odparky

Obrázek 29 znázorňuje schéma rotační vakuové odparky, která bude umístěna v laboratoři mokrých procesů na Univerzitě Tomáše Bati ve Zlíně.

3.1.1 Popis funkce rotační odparky

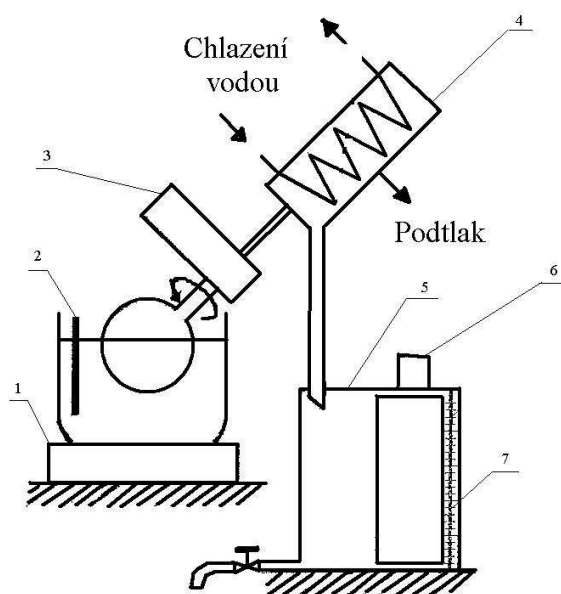
Jedná se o zařízení, které zajišťuje odpařování a zpětnou kondenzaci odpařených par. Odparka může pracovat periodicky nebo nepřetržitě. Při periodickém provozu se odparka naplní zředěným roztokem a pak se opaří určité množství rozpouštědla, aby se dosáhlo požadované koncentrace roztoku. Zároveň poklesne hladina roztoku v odparce a během odpařování se mění teplota a koncentrace roztoku, což má za následek změnu hybné síly a součinitele prostupu tepla.

V našem případě se jedná o rotační odparku, kterou by jsme mohli zařadit do skupiny jednostupňových periodicky pracujících odparek. Rotační odparka použitá pro zahuštění enzymatického hydrolyzátu rovnoměrně otáčí baňkou s odpařovaným roztokem a tím ho rovnoměrně promíchává. Promíchávání roztoku má za následek lepší odpařování vody z roztoku v baňce. Nevýhodou je, že se také odpařuje i vody z vodní lázně odparky na povrchu baňky. Dalším zlepšením odpařování vody z roztoku je zavedení podtlaku do celého systému, tím dojde ke snížení bodu varu kapaliny a tím jsou zajištěny nižší náklady na energii, při ohřívání. Hodnota podtlaku má velmi zásadní vliv na průběh odpařování a proto je důležitým požadavkem co největší podtlak systému. Nevýhodou ale je že se tím zvyšují náklady na vytvoření podtlaku a proudění chladicí kapaliny.

3.1.2 Schéma rotační odparky

Celý regulační systém rotační odparky se skládá z viz Obrázek 29:

- 1) Topného tělesa
- 2) Snímače teploty
- 3) Rotační odparky
- 4) Chladiče
- 5) Sběrné nádoby kondenzátu
- 6) Snímače podtlaku
- 7) Snímače výšky hladiny



Obrázek 29: Schéma rotační odparky

3.1.3 Zapojení regulačního obvodu rotační odparky

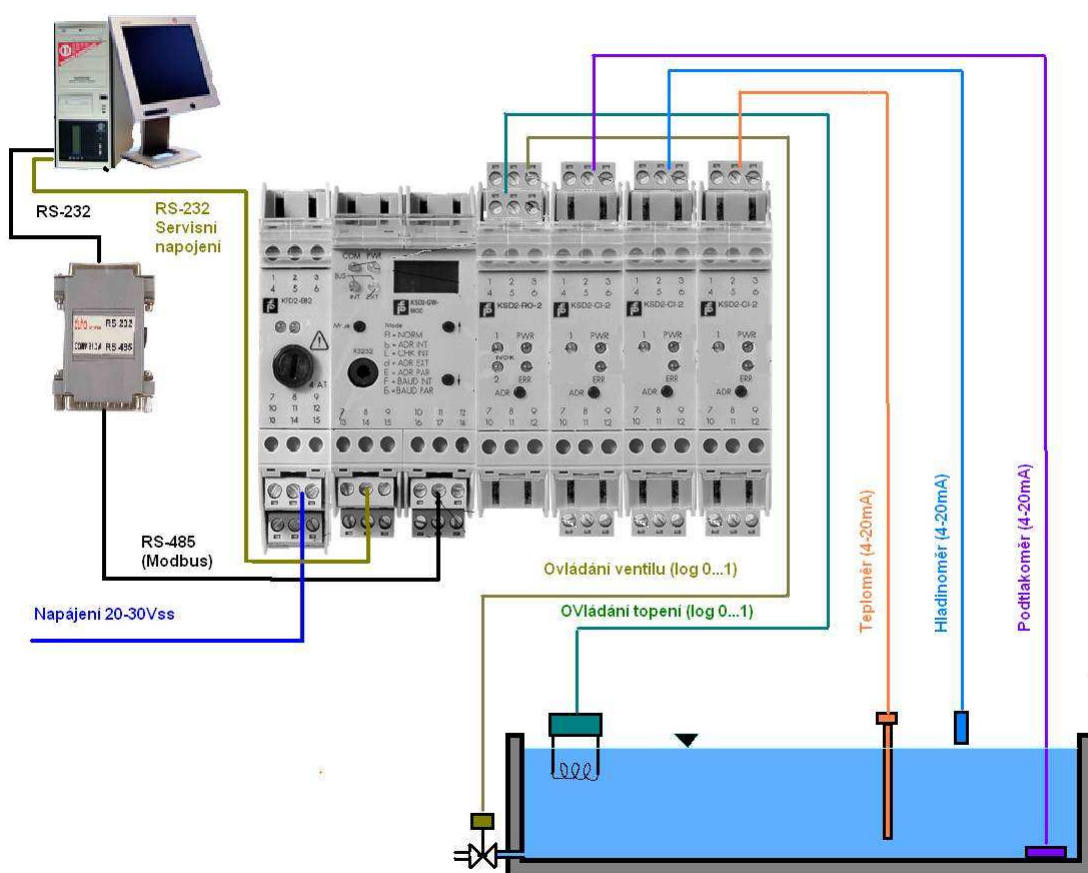
Celý systém, viz Obrázek 30, bude řízeno osobním počítačem standardu IBM nebo malým průmyslovým PDA počítačem Adventech. Na řídicím počítači poběží RUN TIME verze programu Control Web 2000 s nainstalovanou podporou sběrnice ModBus. Podřízené součásti (KSD2-GW-MOD) bude připojena k řídicímu počítači pomocí rozhraní

RS 232, na kterém bude hardwarový převodník, který bude převádět standard RS232 na RS485.

Zařízení od firmy Proper-Fuchs KSD2-GW-MOD slouží jako brána mezi řídicím počítačem a koncovými měřícími (vstupními) a řídicími (výstupními) členy. Použity jsou KSD2-CI-S.H, které je analogový jednobaný snímač senzorů s výstupem 0 (4) – 20mA a KSD2-RO-2, které je dvoustavový (zapnuto, vypnuto) řídicí člen. Na tento koncový člen mohou být napojeny 2 akční členy dvoustavových hodnot.

Analogovými senzory s výstupem 0(4)-20mA budeme měřit tyto hodnoty rotační vakuové odparky:

- teplotní
- podtlakové
- hladinové



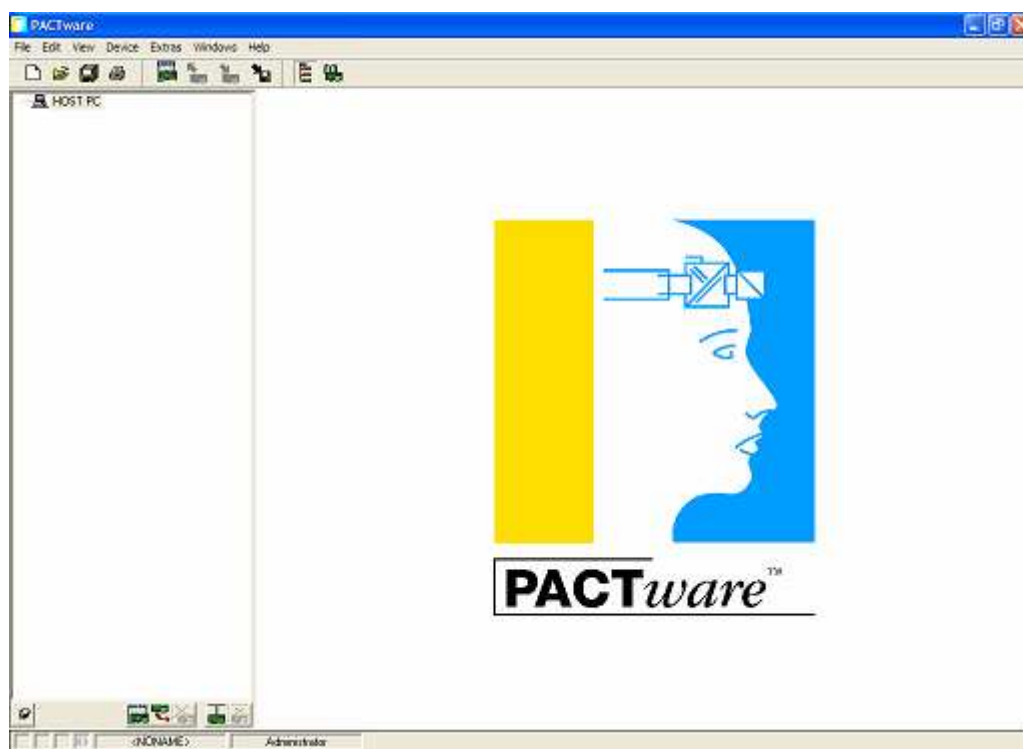
Obrázek 30: Schéma rotační odparky z hlediska snímání a regulace

Podřízený řídicí systém je umístěn na napájecí a zároveň komunikační liště (interní sběrnice) s označením PR-05 (Power Rail with grid, 5 leads) na které bude taky umístěn, kromě již zmíněných zařízení, také napájecí člen s pojistkou KFD2-EB2. Napájení je realizováno stejnosměrným napětím a to 20 – 30V optimálně je to 24Vss.

Po interní sběrnici zařízení mezi sebou komunikují pomocí standardu CAN, který byl popsán v kapitole 2.2.

3.2 Program PACTware 2.4

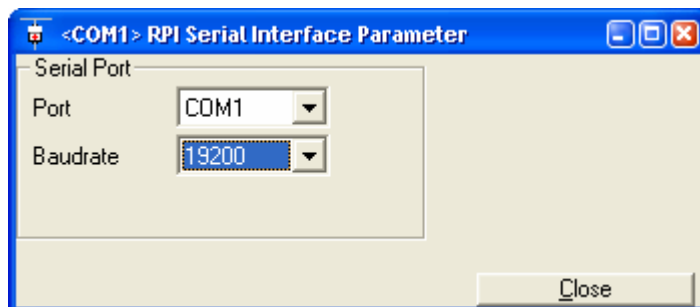
Program PACTware 2.4, je dodávám jako programové vybavení k produktům firmy Proper-Fusch a slouží k servisnímu nastavování jednotlivých částí automatizačních komponentů již zmíněné firmy.



Obrázek 31: Úvodní okno programu PACTware

Program je rozdělen na 2 okna a to na okno připojených zařízení, které jsou zobrazeny, tak jak jsou fyzicky zapojeny a okno dialogové, ve kterém je možno číst diagnostické zprávy a taky konfigurovat daná zařízení.

Do levé části přidáme nejprve ovladač sériového rozhraní (RPI Serial Interface) ve kterém nastavíme port na kterém bude zařízení fyzicky připojeno (COM 1 nebo COM 2) a přenosovou rychlost viz Obrázek 32.

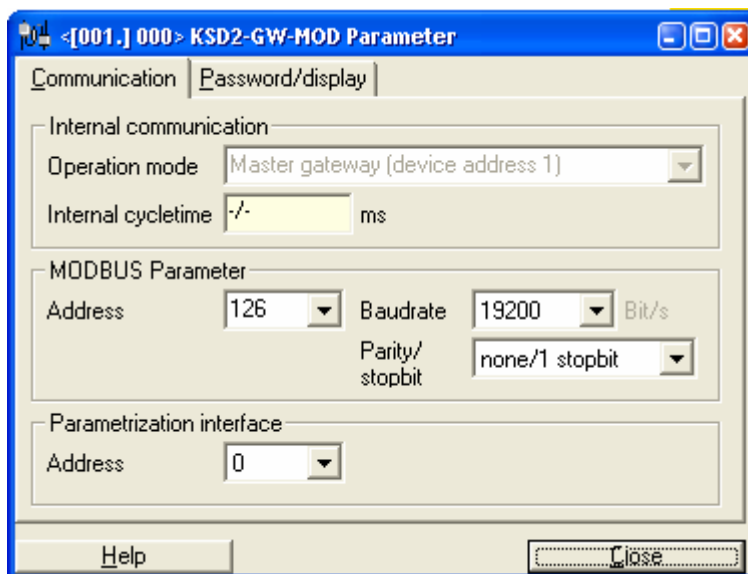


Obrázek 32: Nastavení ovladače sériového rozhraní

Dále přidáme zařízení které bude sloužit jako brána (Gateway), v našem případě to bude KSD2-GW-MOD, na kterém je možno nastavovat parametry ModBus sběrnice a to:

- adresu která může být v rozsahu 0-127 (2^7)
- přenosovou rychlost sběrnice 1200 – 57600Bit/s (standardně nastvaní 19200 Bit/s)
- typ parity/počtem stop bitu

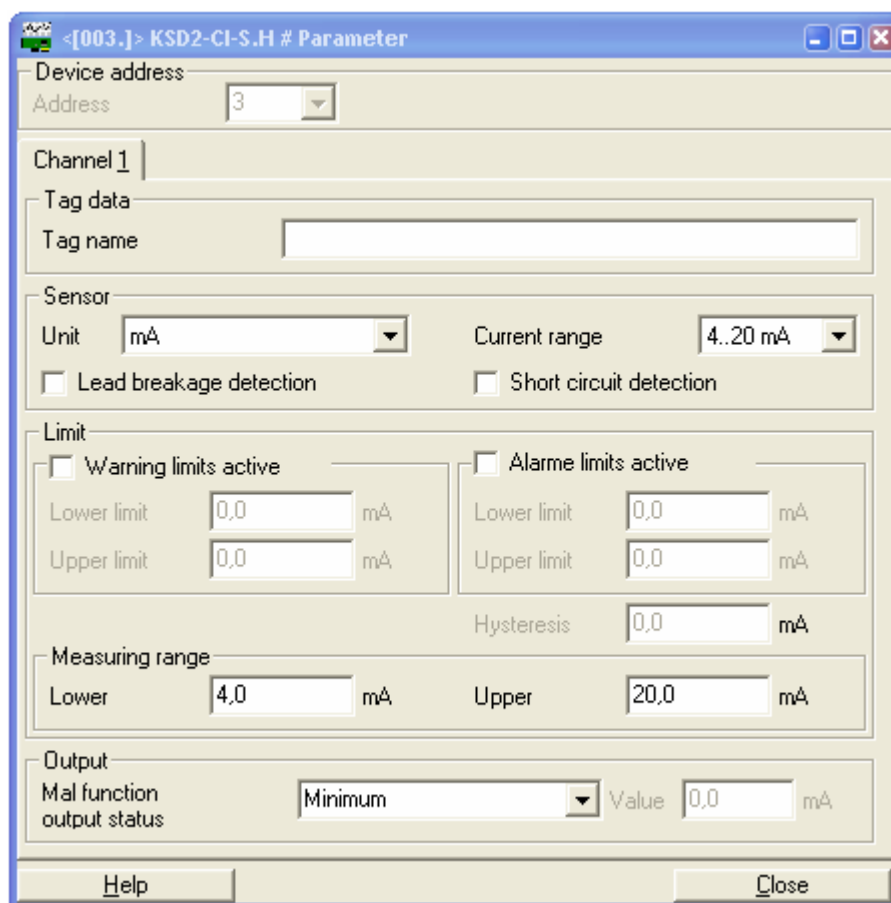
Dále je zde umožněna simulace poruch a diagnostika zařízení.



Obrázek 33: Nastavení parametrů zařízení KSD2-GW-MOD

Následně můžeme přidávat měřící koncová vstupní nebo výstupní zařízení, podle toho které máme fyzicky připojeny na Gateway. U všech připojených zařízení je nutno nakonfigurovat RPI adresu 3 – 127 a adresy se nesmí opakovat. Dále je zde také možnost nastavení, které záleží přímo na daném typu zařízení, např KSD2-CI-S.H (spojitý vstup) umožňuje nastavení typ měřené veličiny, typ spojitého signálu 0-20mA nebo 4-20mA, jména (popisky) dané veličiny a taky nastavení minima a maxima, které odpovídají 0(4) –

20 mA. Každé měřící zařízení přímo převádí spojitý signál na digitální formu v závislosti na nastavení.



Obrázek 34: Nastavení parametrů zařízení KSD2-CI-S.H

Velkou výhodou je fakt, že měřící koncové zařízení si pamatují nastavení do nich uložené a po výpadku proudu se nemusí vše nastavovat znova. I tyto koncové zařízení umožňují prohlížení měřených hodnot, diagnostiku a simulaci stavů.



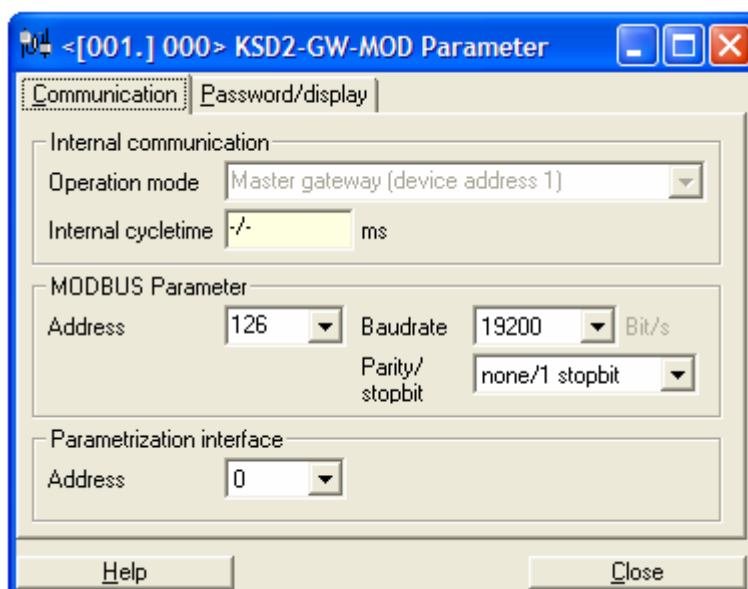
Obrázek 35: Příklad prohlížení aktuální měřené hodnoty, pomocí servisního konektoru

3.3 Nastavení brány (gateway) KSD2-GW-MOD

Brána (gateway) KSD2-GW-MOD, lze nastavit buď přímo pomocí tlačítek a jednoduchého displeje nebo pomocí servisního rozhraní a programu PACTware 2.4, který umožňuje pouze nastavení parametrů sběrnice ModBus viz Obrázek 36. Nastavují se parametry:

- Adresa na sběrnici (v našem případě 1)
- Rychlost komunikace (19200 Bit/s)
- Typ parity a stop bitu (even – sudá / 1 stop bit)

Veškeré nastavení pak nahrajeme do zařízení pomocí servisního propojení.



Obrázek 36: Nastavení parametrů pomocí servisního programu

Dále je tu možnost ručního nastavení přímo na zařízení, jeho nevýhodou je horší přehlednost nastavených parametrů. Ale nastavují (přiřazují) se zde adresy koncových zařízení. Vše se nastavuje pomocí 3 tlačítek (Mode, tlačítka nahoru a dolů).

Menu má tyto možnosti nastavení:

- A – Normální provoz a čísla chybových hlášení
- B – Nastavení interní adresy ostatních zařízení
- C – Kontrola interní adresy ostatních zařízení
- D – adresa zařízení

- E – Adresa brány pro servisní propojení
- F – Rychlost komunikace po vnitřní sběrnici (CAN)
- G – Rychlost komunikace přes servisní propojení
- H – rychlost komunikace přes ModBus
- I – Typ parity a počet stop bitů

Nastavení (přiřazení) adresy se provádí následovně:

- 1) Najedeme do menu „B“ a pomocí posunovacích tlačítek navolíme adresu, pokud je již adresa použita na jiné zařízení je automaticky přeskočena.
- 2) Na koncovém zařízení podržíme tlačítko ADR 3s stlačené, přestane blikat LED dioda (červená – Error).
- 3) Na gateway nejedeme do menu „A“ a podržíme tlačítko MODE 3s, displej přestane ukazovat chybovou hlášku.
- 4) Kontrolu správnosti zadání interní adresy můžeme provést v menu „C“. Po najetí do tohoto menu, stlačíme tlačítko ADR na koncovém zařízení a na displeji se nám zobrazí číslo koncového zařízení.

3.4 Control Web 2000

Na našem trhu je v současné době dostupná celá řada programových systému SCADA/MMI, které podporují vizualizaci a řízení technologických procesů. Příkladem je Control Web 2000 (nejnovější verze 5), InTouch, Promotic a mnoho dalších. Pro náš případ bylo zvoleno programové vybavení Control Web 2000.

3.4.1 Základní informace

Control Web 2000 je univerzální nástroj pro vývoj a nasazování vizualizačních a řídicích aplikací, aplikací sběru, ukládání a vyhodnocování dat, aplikaci rozhraní člověk-stroj. Objektově orientovaná struktura zajišťuje aplikacím systému nejširší rozsah nasazení od nenáročných aplikací vizualizace až po řízení v reálném čase.

Control Web 2000 koncepčně vychází z osvědčené architektury svých předchozích verzí a to Control Panel a Control Web.w Nasazením těchto systémů do jaderných

elektráren a celopodnikových vizualizačních systémů a až po přímé řízení strojů a jednoduché vizualizace dokazuje velmi široké možnosti této architektury.

Hlavním cílem návrhu systému Control Web 2000 je učinit realizaci běžných úkolů snadnou a neobvyklých, komplikovaných úkolů možnou. Samozřejmostí je respektování všech existujících standardů pro běh programů a jejich uživatelské rohraní, výměnu dat a přístup k databázím, komunikaci po počítačových sítích a spolupráci s hardware pro sběr dat a zařízení.

Control Web 2000 pracuje v prostředí operačních systémů implementující aplikační programové rozhraní WIN32 a podporuje řadu průmyslových standardů.

Control Web 2000 má vlastnosti:

- je obdobný všem ostatním SCADA/MMI systémům používaných v průmyslu. K dispozici jsou všechny komponenty nutné k tvorbě vizualizačních aplikací, dále poskytuje skutečnou programovatelnou a otevřenou komponentovou architekturu. Zabudovaná není množina vizualizačních přístrojů. Každý přístroj je dynamicky linkovaná knihovna detekovaná při startu systému. Tuto množinu lze dle potřeb libovolně rozšiřovat.
- Umožňuje sekvenční řízení procesů. Virtuální přístroje nemusí být aktivovány, ale v přesně definovaném čase a v přesně definované sekvenci.
- Jednotlivé komponenty systému jsou volně programovatelné. Pokud potřeby uživatelů sahají za možnosti řady panelů s vizualizačními a ovládacími prvky, každá komponenta má k dispozici mocné programátorské nástroje, jako jsou lokální proměnné a libovolně definovatelné procedury reagující na události.
- Umožňuje práci v reálném čase. Nespolehá na tzv. databázi reálného času, která je doplňována maximální možnou rychlostí, v praxi to může být i několik sekund. Každý kanál je čten přesně v době, kdy jej nějaký virtuální přístroj požaduje. Real-time časování je přesně monitorováno a řízeno.
- Umožňuje tvorbu distribuovaných řešení. Tím se nemyslí pouze replikace tagů přes NEJDDE v NEJBIOS sítích, ale skutečné zpřístupnění libovolného datového elementu všem spojeným aplikacím pro libovolné TCP/IP sítě včetně Internetu. Virtuální přístroje mohou být aktivovány po síti stejně jako metody dynamického

rozhraní virtuálních přístrojů mohou být volány po síti. Síťová komunikace může být samozřejmě precizně časována a řízena k dosažení optimálního výkonu.

- Umožněna je také vizualizace technologií prostřednictvím internetových standardů HTTP a HTML pomocí libovolného WWW klienta, ať pracuje v prostředí MS Windows, Apple Macintosh nebo UNIX. Přitom se nejedná o pouhý plug-in do nějakého existujícího HTTP server, protože Control Web obsahuje plnohodnotný HTTP server dynamicky tvořící stránky podle stavu technologie pracující i na obyčejných Windows 95. Navíc umožňuje pomocí HTTP a HTML technologii i řídit.
- Kontejner pro komponenty Active X představuje most mezi systémem Control Web 2000 a vlastnostmi i metodami (OLE Automation) Active X komponent. To znamená, že do aplikace lze zabudovat Active X komponenty a programově je ovládat (nastavovat jim vlastnosti a volat metody z procedur jakýchkoliv přístrojů).
- Control Web 2000 je nezávislý na použitém hardware. DDE je samozřejmě podporováno spolu s OPC (OLE for Process Control) a řadou dalších standardů pro průmyslové automaty, samostatné moduly a měřicí karty. Rozhraní ovladačů je plně dokumentováno a otevřeno, takže každý si může ovladač napsat podle svých potřeb.
- Vzestupná kompatibilita aplikací mezi systémy Control Pane, Control Web a Control Web 2000 rozšiřuje jedinou architekturu na nejširší škálu hardware, od bezdiskových stanic až po multiprocesorové servery.
- Snadnost používání může redukovat programování na několik pohybů myši. Průvodci budují kostru aplikace a navigují uživatele přes počáteční stadia návrhu aplikace. Integrované vývojové prostředí umožňuje kdykoliv předcházet mezi textovým módem a grafickým módem návrhu. Aplikace je vždy uchována v podobě textového souboru, nikoliv v podobě nečitelných binárních dat.
- Control Web 2000 podporuje zásadně dva odlišné módy činnosti. Real-time mód je identický se současně podporovanou architekturou. Nový vizualizační mód vyžaduje výrazně menší snahu od tvůrce aplikace. Přístroje nejsou časovány a jsou aktivovány po změně dat. Menší úsilí nutné k vytvoření aplikace je vykoupeno

ztrátou real-time vlastností a možností programovat řídicí algoritmy. Bez možnosti napsat jakýkoliv program není ani možno napsat špatný program.

3.4.2 Vývojové prostředí

Je rozděleno na několik částí. Tyto části jsou však společně provázány. Pracovní plocha tohoto vývojového programu je rozčleněna do několika sekcí, mezi kterými se lze přepínat pomocí záložek umístěných v dolní části obrazovky.

Moduly

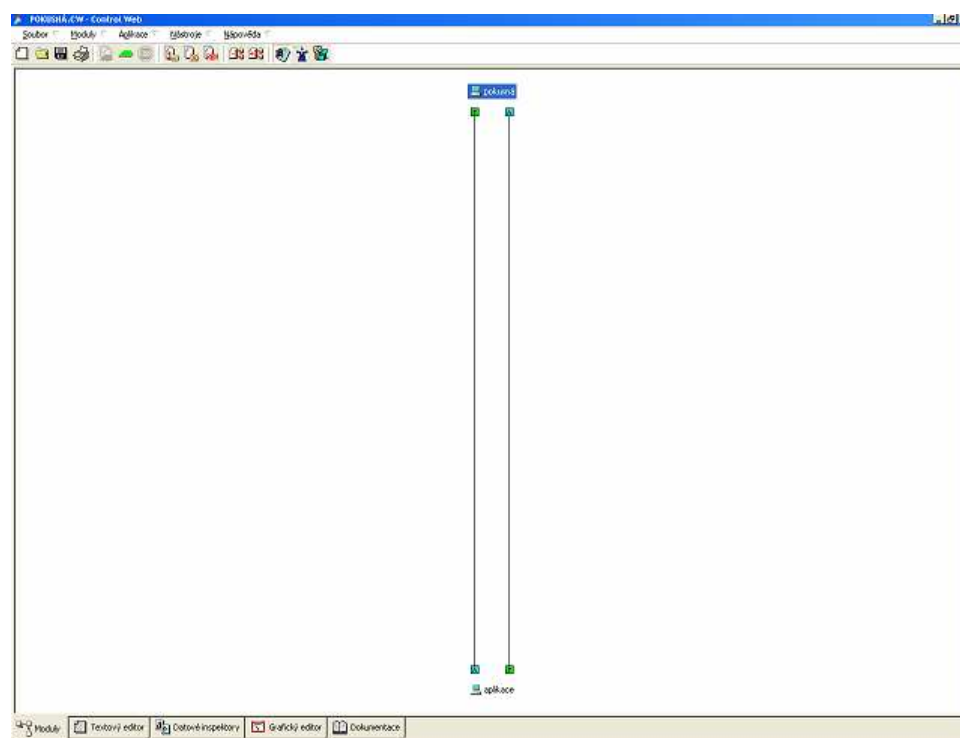
Využívá se u vícemodulárních aplikací k definici vazeb mezi jednotlivými samostatnými programy (celá aplikace se nakonec tváří jako aplikace jediná), které mohou být umístěny buď na stejném počítači nebo přístupné kdekoli na síti.

Souhrnně se aplikace, které Control Web provozuje, označují jako:

- modulární - sestává se z více modulů (souborů). Je-li modul jen jeden, aplikace je jednomodulární (případ většiny jednoduchých aplikací),
- distribuované - čili vícemodulární aplikace, jejíž jednotlivé moduly běží na různých počítačích. Jednomodulární aplikace nemůže být distribuovanou.

Moduly, ze kterých se vícemodulární aplikace skládá, jsou:

- místní - běží na počítači, kde byl spuštěn první modul
- vzdálený - běží na jiném počítači.



Obrázek 37: Záložka moduly

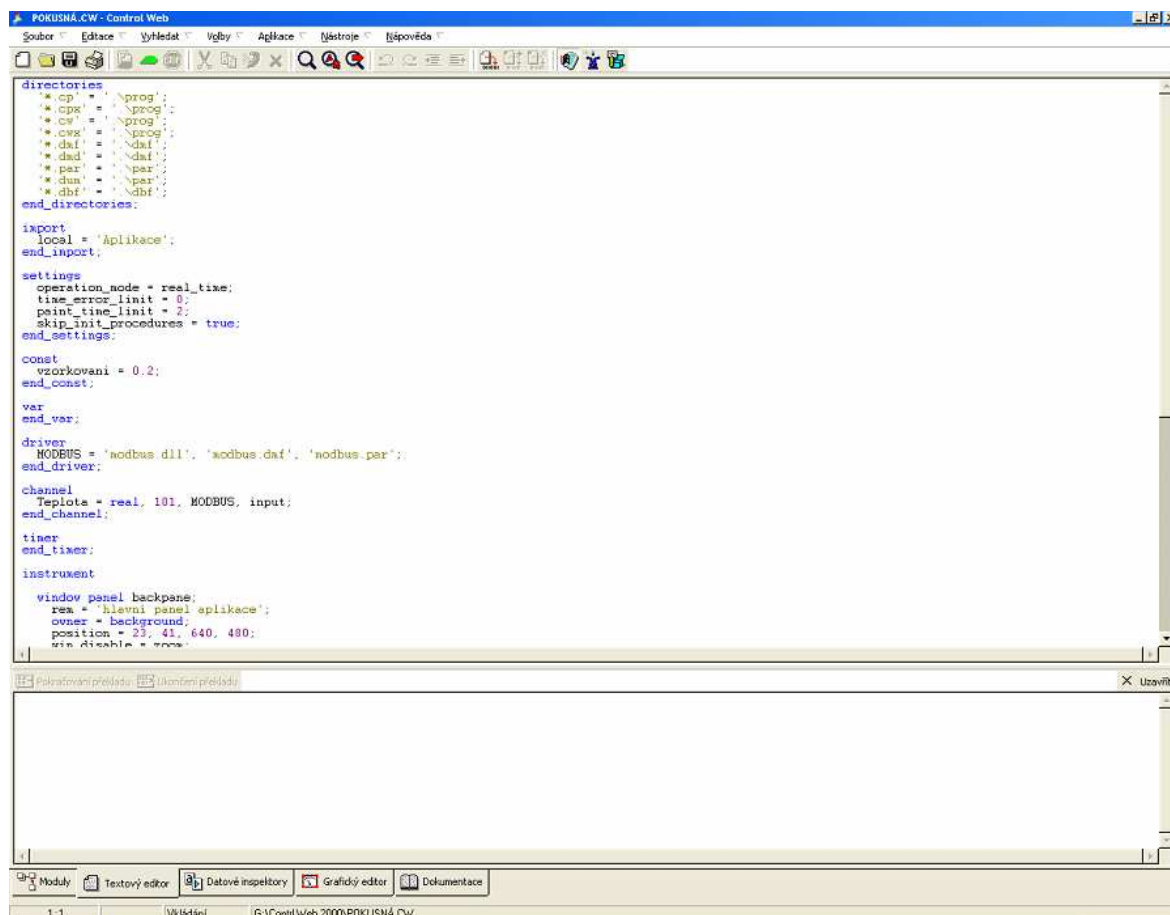
Textový editor

Z počátku se u zkušených uživatelů a to jen vyjíměčně začíná programovat přímo ve zdrojovém kódu. Mnohem jednodušší je začínat v grafickém editoru, kde vše probíhá v režimu WYSISYG a zadávání parametrů pomocí dialogových oken. Později při ladění aplikace je využití textového editoru mnohem výhodnější, pokud tedy nechcete přidávat např. nový přístroj. S postupem času se pro uživatele stane syntaxe zápisu zdrojového kódu čitelnější a začnete se v něm orientovat.

Editor je rozdělen na dvě části:

- horní část – probíhá editace zdrojového kódu aplikace a velkou výhodou je barevné zvýraznění klíčových příkazů, textových řetězců (v uvozovkách) a čísel. Ve zdrojovém kódu větší přehlednost.
- Dolní části – zde se objevují zprávy z kompilátoru.

Důležitým faktem je že jsou rozlišovány velké a malé písmena.



Obrázek 38: Záložka Textový editor

Datové inspektory

Pokud neovládáte dostatečně zápis v textovém editoru, budete všechny globální konstanty a proměnné, ovladače, kanály a moduly definovat zde. Dále tato sekce umožňuje měnit nastavení aplikace.

Datové elementy můžeme rozdělit na:

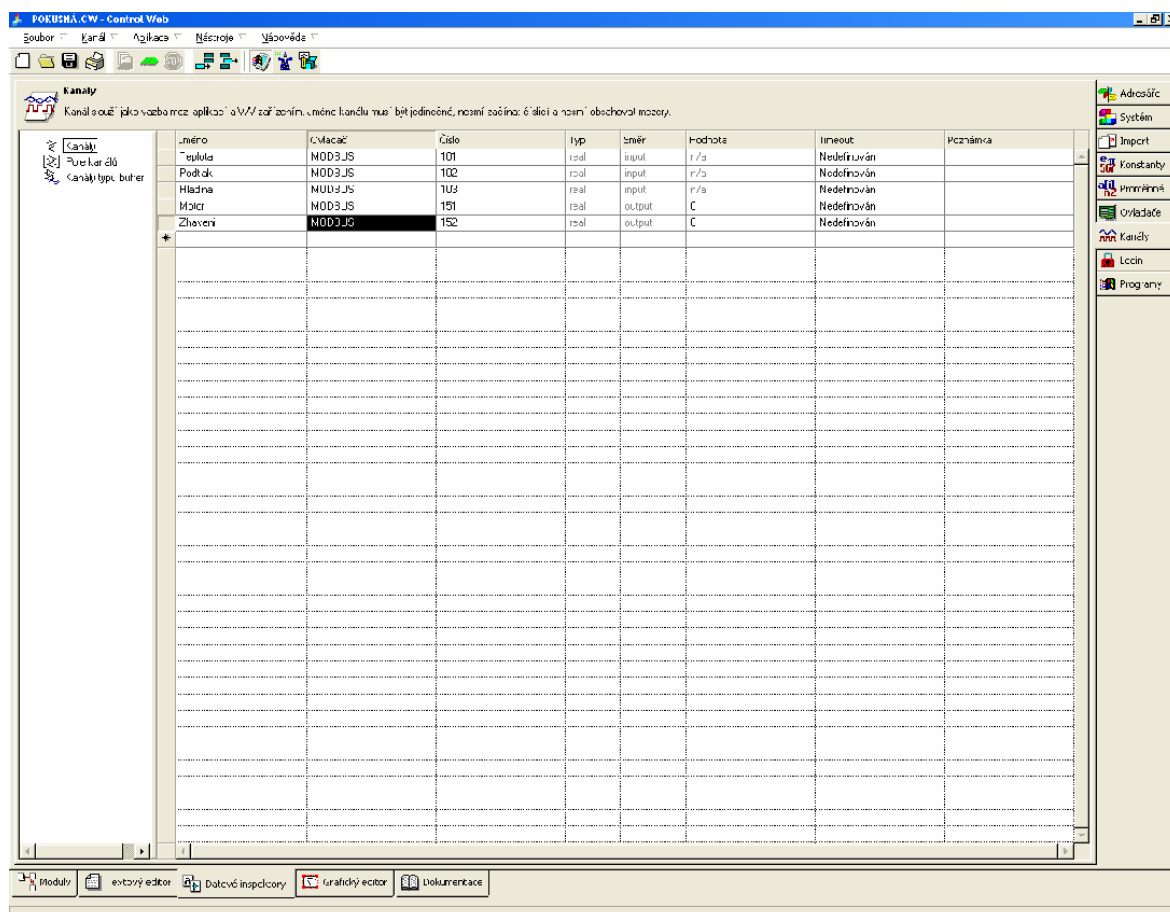
- globální - společné pro celý program a mohou je používat veškeré virtuální přístroje, definují se právě v této záložce
- lokální – využívají se v rámci jednoho virtuálního přístroje. V případě shody názvů s globálním elementem má lokální přednost.

Podle směru toku dat dělíme datové elementy na:

- vstupní – lze z nich pouze číst
- výstupní – lze na ně pouze zapisovat
- obousměrné – umožňují zápis a čtení

Z hlediska povahy datových elementů je můžeme rozdělit na:

- číselné
- logické
- textové



Obrázek 39: Záložka Datové inspektory

Adresáře

Ekvivalentní se sekci *directories* ve zdrojovém zápisu. Při běžné práci vůbec nevyužijete. V případě použití externích souborů zde uvedete cestu a jejich příponu. Po vygenerování nového projektu je nastavena standardní cesta.

Systém

Ekvivalentní se sekci *settings* ve zdrojovém zápisu. Chod aplikace je nastaven automaticky a ke správné funkci není potřeba z těchto parametrů nic měnit. Mezi další možnosti patří zobrazení okna s informacemi o časování, nastavení důležitých časů pro kanály, moduly a nastavení síťových parametrů.

Import

Ekvivalentní se sekcí *import* ve zdrojovém kódu. Zde se definují moduly distribuované aplikace, ať už lokální či vzdálené a jejich vzájemná vazba. V případě lokálního modulu se uvádí pouze cesta ke zdrojovému souboru, v případě vzdáleného nebo připojovaného modulu je nutné zadat IP adresu nebo doménu počítače a opět cestu ke zdrojovému souboru. V obou posledních případech se objeví několik parametrů (se standardními hodnotami) nastavujících komunikaci s moduly. Vzdálený modul je se základním modulem (projektem) spojen pevně.

Konstanty

Ekvivalentní se sekcí *const* ve zdrojovém kódu. Pod tímto pojmem se skrývá seznam všech globálních konstant.

Proměnné

Ekvivalentní se sekcí *var* ve zdrojovém kódu. Zde se definují globální proměnné, jejich typ a počáteční hodnoty.

Ovladače

Ekvivalentní se sekcí *driver* ve zdrojovém kódu. Pro komunikaci se vstupně/výstupními zařízeními slouží ovladače, které jsou pro každý typ zařízení specifické. Control Web obsahuje šest ovladačů, ostatní jsou samostatným produktem a je nutné je objednat. V našem případě je to ovladač modbus.

Kanály

Ekvivalentní se sekcí *channel* ve zdrojovém kódu. Po zapsání ovladače je k vlastní komunikaci zapotřebí vytvořit kanály. Ty mohou být z hlediska směru toku dat vstupní nebo výstupní a jejich název je přímo globální proměnnou, která obsahuje hodnoty s nimiž lze pracovat. K jejich vytvoření je nutné znát nastavení ovladače (čísla vstupních a výstupních kanálů, pod kterým v systému Control Web vystupují, nikoli čísla fyzických vstupů a výstupů na konektoru laboratorní karty).

Login

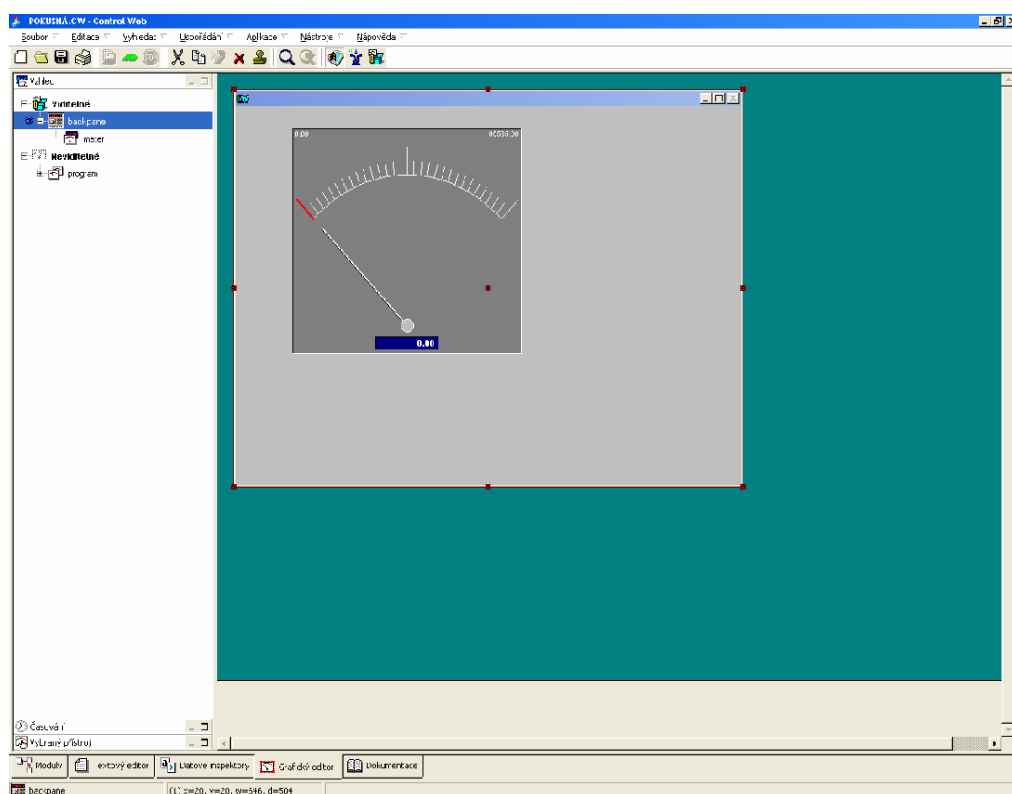
Zde se definují přístupová práva uživatelů.

Programy

Programy, uvedené v seznamu budou spuštěny při startu aplikace.

Grafický editor

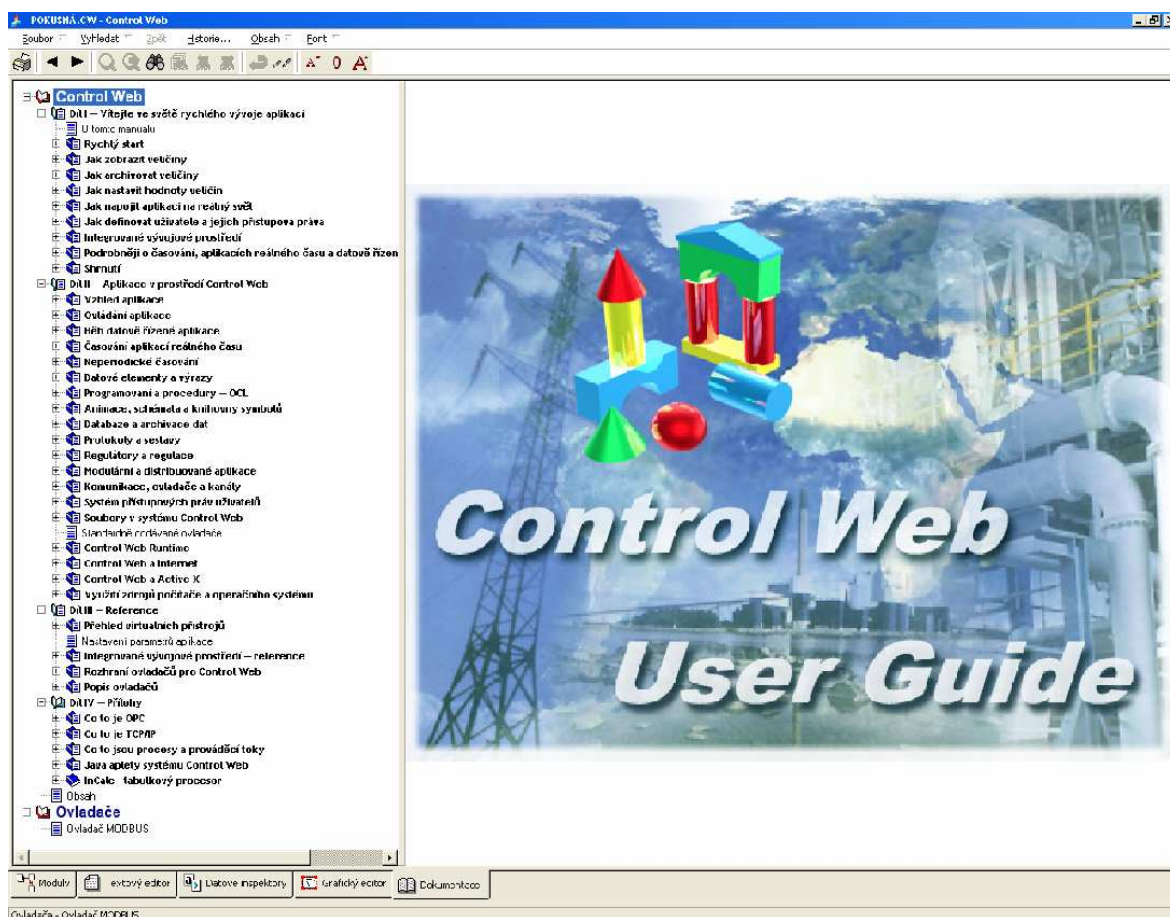
Vždy, protože jde o WYSIWYG (What You See Is What You Get) editaci, která je velmi snadná a rychlá a s pomocí několika formulářů, kde vyplníte požadované hodnoty, za vás vygeneruje zdrojový kód sama. Z hlediska rozmísťování a prvnímu nastavování přístrojů je to nejlepší možnost. Pohodlnost je ale na úkor času. Přeci jen je v textovém editoru opravení jedné položky dílem okamžiku, kdežto zde je nutné se k nějaké akci 'proklikat' přes nejedno menu. Veškeré změny provedené v grafické, resp. textové podobě se při přepnutí automaticky změní i v textové, resp. grafické části editoru. V grafickém editoru je v levé části obrazovky je *hierarchický strom*, ve kterém je zobrazena struktura vytvářené aplikace, včetně hladinového uspořádání a přivlastnění přístrojů příslušným panelům (pokud jich aplikace má více).



Obrázek 40: Záložka grafický editor

Dokumentace

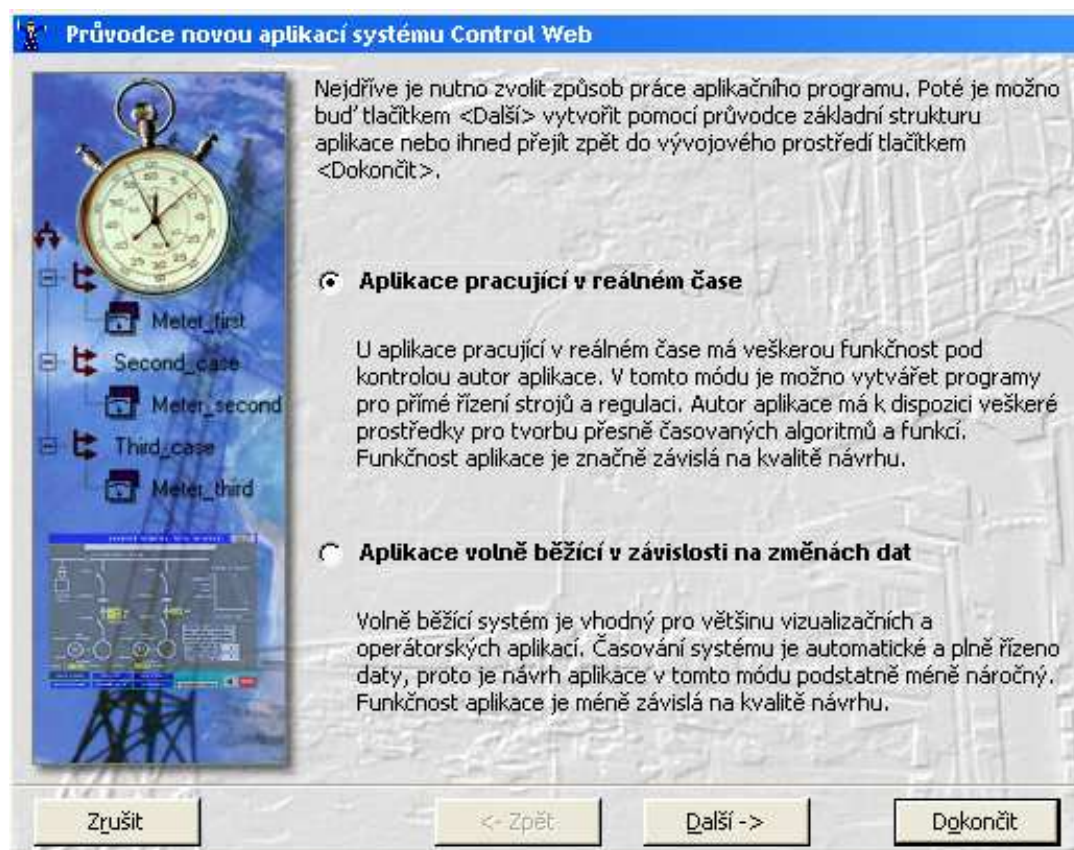
Je to nápověda která pomáhá pochopit, jak celý systém funguje. Jde to velmi podrobný popis a najít dílčí informaci je někdy zdlouhavější záležitostí. Při doplnění systému Control Web 2000 o další modul případně ovladač, je tato dokumentace rozšířena.



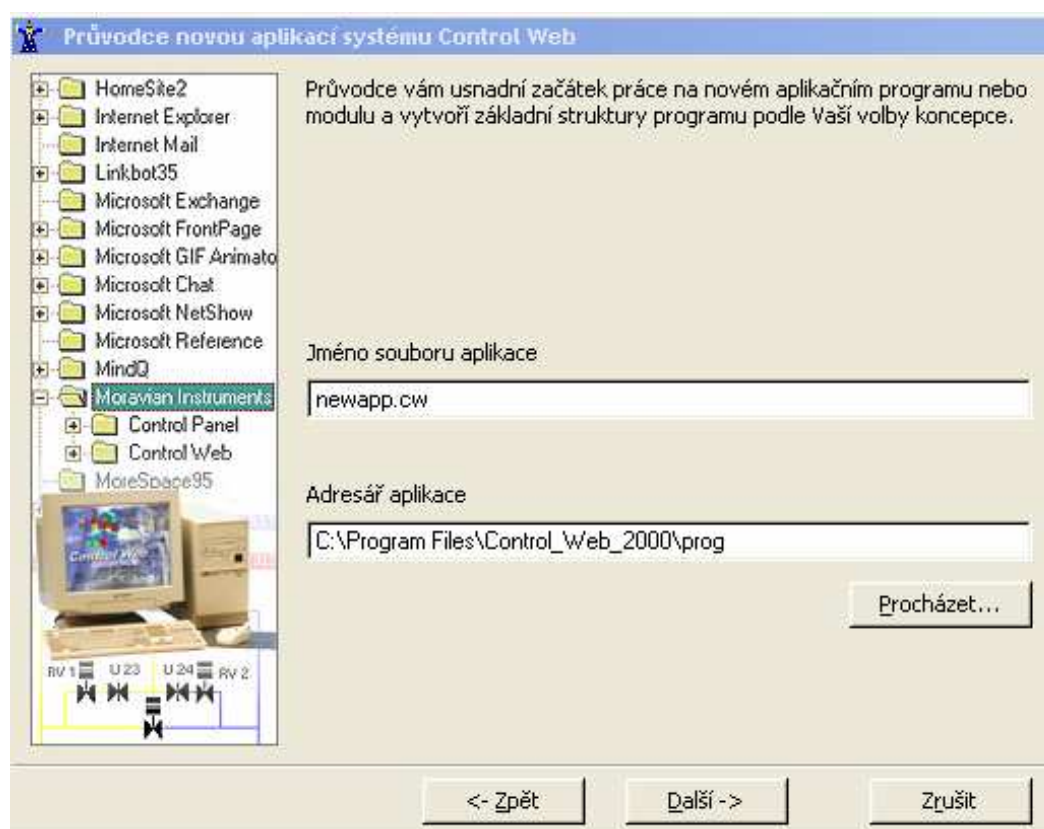
Obrázek 41: Záložka dokumentace

3.4.3 Vytváření aplikace

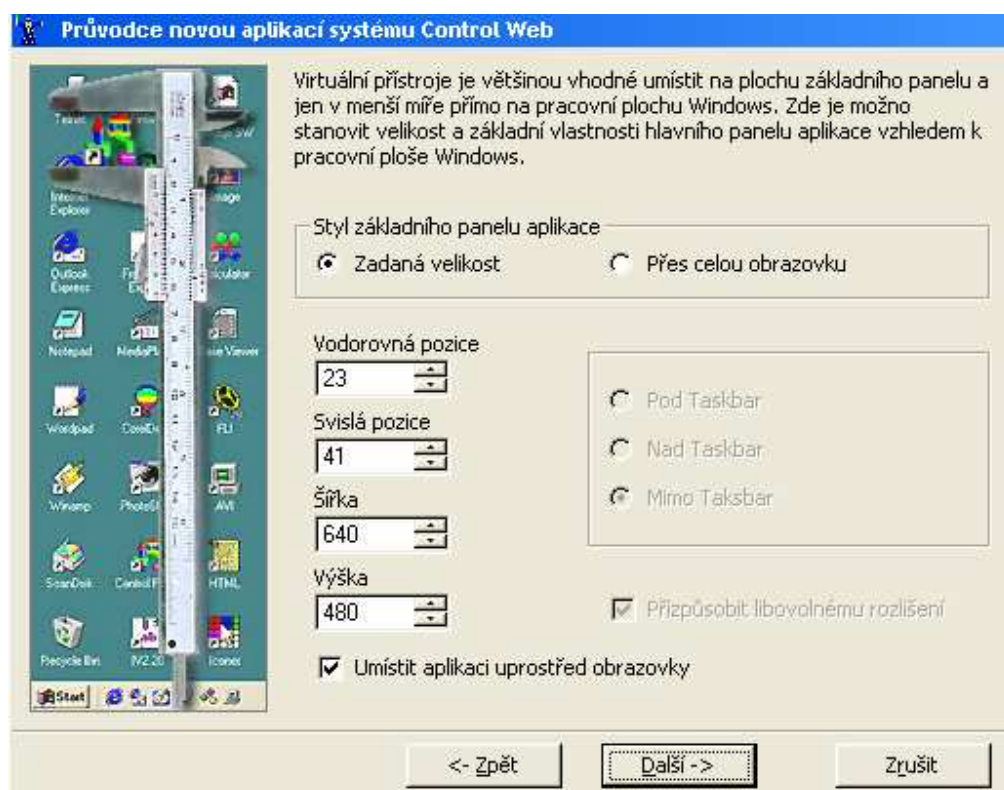
Po zvolení možnosti „Nová aplikace“ se objeví okno s možností zvolení časování a to, jestli aplikace pojede v reálném čase nebo jestli bude aplikace řízena daty viz. Obrázek 42. V našem případě zvolíme aplikaci v reálném čase. A zvolíme možnost další. Na další stránce zadáváme název souboru, ve kterém bude uložena aplikace, včetně přípona *.CW a taky adresář do kterého aplikaci uložíme viz. Obrázek 43. Další zadávanou možností je zvolení velikosti aplikace viz Obrázek 44. Na Obrázek 45 se volí zda aplikace bude mít jeden více panelů, v našem případě zvolíme pouze jeden panel. A na poslední stránce průvodce jsou možnosti časování a startu aplikace viz. .



Obrázek 42: Průvodce - zvolení časování



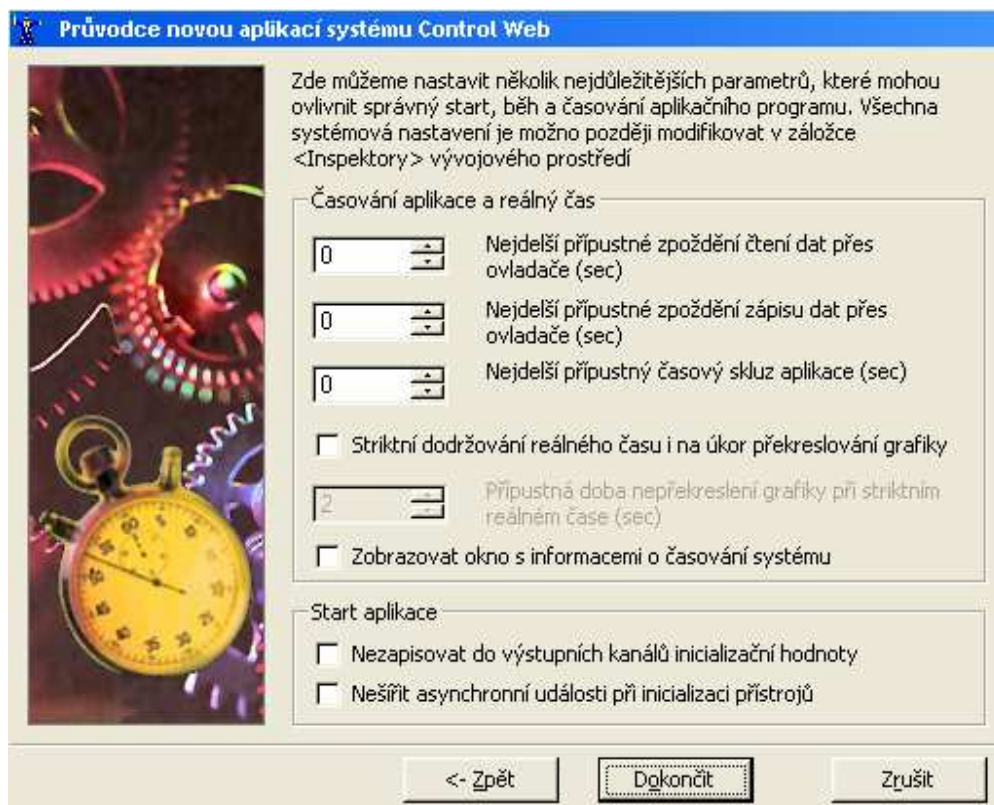
Obrázek 43: Průvodce - název souboru a výchozího adresáře



Obrázek 44: Průvodce - zvolení velikosti obrazovky



Obrázek 45: Průvodce - nastavení možnosti počtu panelů



Obrázek 46: Průvodce - nastavení časová a startu aplikace

Po dokončení průvodce se program automaticky přepne do textového editoru.

3.4.4 Nastavení konstant, ovladače a kanálů

My se přepneme do záložky datové inspektory. Kde nastavíme v záložkách, které jsou na pravé straně. V záložce konstanty nastavíme periodu vzorkování. Pak zvolíme ovladač a k němu nakonfigurované soubory modbus.dmf, ve kterém jsou nakonfigurované virtuální kanály, 100-150 reálný vstup a 151-199 reálný výstup. Přidáme také parametrický soubor modbus.par, kde je nakonfigurované připojení a taky ze kterých registrů daného zařízení budeme číst nebo zapisovat data z virtuálních kanálů. Z důvodu lepší přehlednosti na záložce kanály přiřadíme použitým virtuálním kanálům názvy. V záložce proměnné nastavíme typy proměnné a jejich hodnoty:

- Max_hladina a min_hladina – pro nastavení maximální a minimální hladiny
- Max_teplota a min_teplota – pro nastavení maximální a minimální teploty
- Max_podtlak a min_podtlak – pro nastavení maximální a minimální hladiny

3.4.5 Nastavení souborů modbus.dmf a modbus.par

V těchto souborech se nastavují vlastnosti komunikace sběrnice modbus a také rozsah a typy kanálů.

V souboru Modbus.dmf se nastavují vstupní a výstupní virtuální kanály. Tvar zápisu je následovný: kanály od – do, typ proměnné a směr toku dat.

Příklad:

Begin

100-150 real input

151-199 boolean output

end.

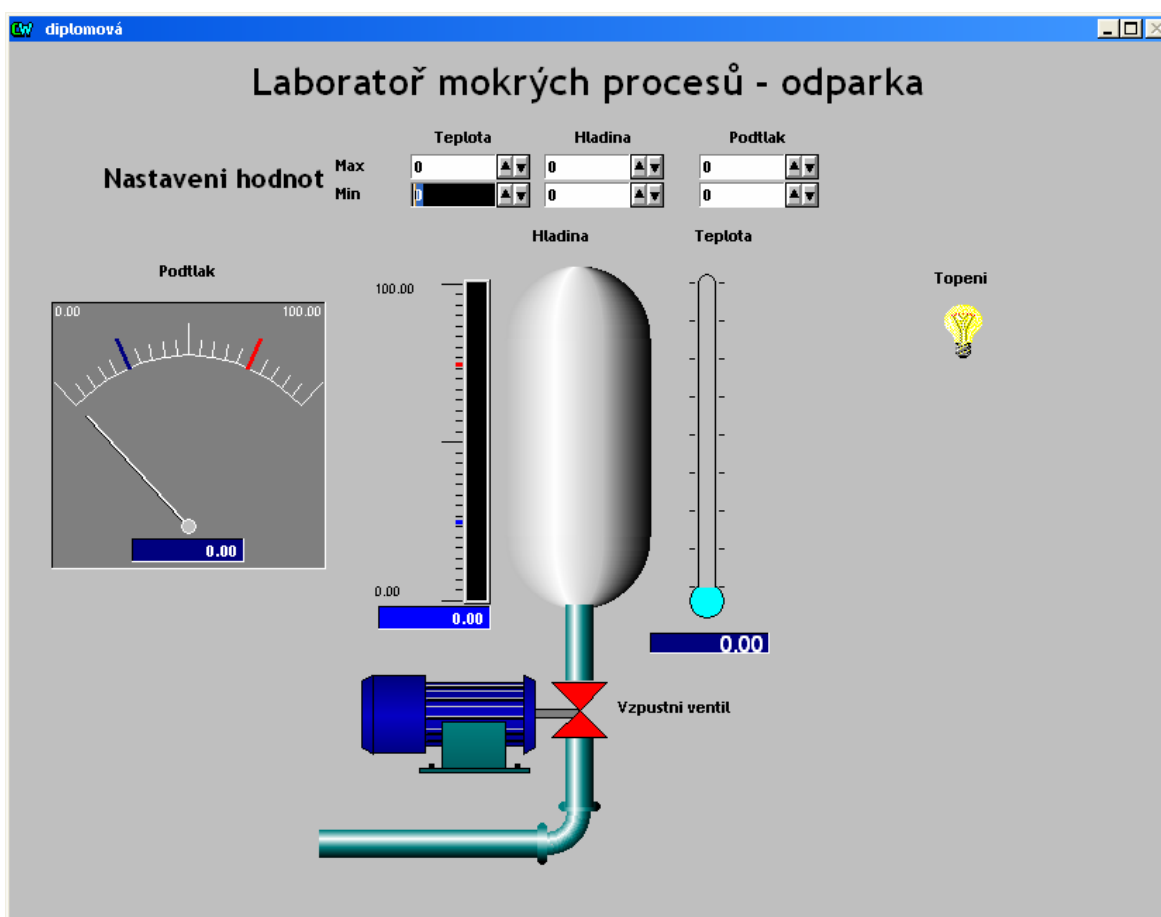
V souboru modbus.par se nastavují vlastnosti přenosové linky a přiřazení virtuálních kanálů na paměťové registry zařízení. Nastavení musí být shodné s nastavením na zařízení. V našem případě bylo téměř vše přednastaveno, pouze jsme nastavili jiný typ parity na „even“ (sudou), sériový port, na který je zařízení připojeno. Dále se nastavuje počet Bitů, které přenáší data v našem případě je to 8 (RTU –mód). Celý soubor je umístěn v příloze P1.

Adresy v paměťových registrech mají hodnotu v závislosti na čísle RPI (adresa na interní sběrnici) a to:

- 1) Analogové vstupy – jedná se o data z připojených senzorů a čteme z výstupních registrů zařízení KSD2-GW-MOD. Adresa v paměťovém registru je následová: $PAR = RPI + 4608$ (v našem případě je první registr $4611 = 3 + 4608$)
- 2) Binární výstupy – jedná se o dvouhodnotové výstupy a zapisujeme do vstupních registrů zařízení KSD2-GW-MOD. Adresa pak je: $PAR = 8 * RPI + 1 - \text{kanál}$ (v našem případě je první registr $48 = 8 * 6 + 1 - 1$ a maximální počet výstupních kanálů na zařízení je 8).

3.5 Aplikace pro řízení reálného systému

Celá vytvořená aplikace je graficky velice jednoduchá. Je složena ze 3 měřících vstupních členů a to podlakoměru, teploměru a sledování výšky hladiny. Výstupním členy je žárovky, pro signalizaci topení a potrubí s kompresorem, vysávající vzduch z nádrže, které jsou spínány v závislosti na nastavené hodnotě a to maximální teploty a minimálního podtlaku.



Obrázek 47: Aplikace pro řízení reálného procesu

Celá vizualizační aplikace byla navržena jako velice jednoduchá, protože se jedná pouze o vyzkoušení funkčnosti celého zapojení a výsledná aplikace bude vytvářena po celkovém dokončení laboratoře mokrých procesů.

ZÁVĚR

Tato diplomová práce se zabývá problematikou řízení a monitorování technologických procesů a nastavením protokolu modbus.

Úvodní kapitola práce je věnována rozboru a obecným pojmům datových sítí. Jsou zde vysvětleny pojmy jako je topologie běžně sítí, kódování a různé modulace přenosu, synchronizace vysílacích a přijímacích stanic, zabezpečení dat během přenosu, aby nedocházelo k jejich poškození a způsoby potvrzování přijatých zpráv. Není zde opomenut také standard referenčního modelu ISO/OSI a také metody řízení přístupu k přenosovému médiumu.

Druhá kapitola se zabývá průzkumem a popisu nejpoužívanějších průmyslových sběrnic. Pozornost je zaměřena převážně na sběrnice CAN a protokolu ModBus. Sběrnice CAN je použita pro interní komunikaci měřících a řídicích stanic s bránou, která umožňuje připojení na nadřazený řídicí systém pomocí protokolu ModBus, která je postavena na standardu RS-485. Na závěr druhé kapitoly je zmíněna stále více se rozvíjející bezdrátová síť Wi-fi, která pracuje v bezlicenčním pásmu 2,4GHz nebo 5GHz.

V praktické části je stručně popsána funkce rotační vakuové odparky, která bude umístěna v laboratoři mokrých procesu Univerzity Tomáše Bati ve Zlíně. Tato rotační odparka bude monitorována a řízena pomocí hardwarových produktů firmy Pepperl Fuchs, a průmyslového počítače firmy Adventech TPC-650T na kterém poběží RUN TIME verze monitorovacího a řídicího programu Control Web 2000 od firmy Moravské Přístroje a.s. Je podrobně popsáno nastavení měřících a řídicích stanic pomocí dodávaného servisního programu PACTware, kterým je možné také provádět diagnostiku a simulace kolizních stavů. Popsáno je i nastavení protokolu ModBus, jak v programu Control Web 2000 tak i pro bránu (gateway) KSD2-GW-MOD.

Protokol ModBus byla vybrán, že se jedná o otevřený a lze použít na typech sběrnic například: RS-485, LAN s protokolem TCP/IP. Sběrnice CAN je široce podporována výrobcí integrovaných obvodů a má velmi jednoduchou strukturu. Při nižších přenosových rychlostech lze ji použít i na velké vzdálenosti řádově několika kilometrů.

LITERATURA

- 1) Pham Manh Thang: Modelování recyklačních procesů v kolejeluzské technologii s cílem jejich automatického řízení. Disertační práce FT UTB Zlín 2003.
- 2) Pokorný, P.: Problematika řízení při hydrolyzačním zpracování bílkovinných druhotných surovin. Disertační práce FT UTB Zlín, 2003.
- 3) Husták, P.: Počítačové řízení rotační odparky. Diplomová práce VUT Brno, FT Zlín 1999.
- 4) Klen P. : Datová komunikace počítačů PC v distribuovaných systémech řízení. Disertační práce, VŠB-TUO Fakulta Strojní, 2000
- 5) Lawrenz, W.: CAN Systém Engineering: From Theory to Practical Applications. Spring Verlag, 1997
- 6) Škuta, J.: Řízení a monitorování technologických procesů s využitím sítí LAN a ILAN, VŠB-TUO Fakulta Strojní, 2003
- 7) Janda, P.: Inteligentní senzory v průmyslových měřících a řídicích systémech s podporou ILAN, VŠB-TUO Fakulta Strojní, 2000
- 8) Kotzian, J.: Modelování průmyslových sběrnic v distribuovaných systémech řízení, VŠB-TUO Fakulta Strojní, 2003
- 9) Fieldbus, popisy sběrnic [online] . Dostupný u URL: <http://fieldbus.feld.cvut.cz/>
- 10) Průmyslové sběrnice , odkazy [online] . Dostupný u URL: <http://www.fieldbus.cz/>
- 11) ModBus: popis sběrnice [online]. Dostupný z URL: <http://automatizace.hw.cz/view.php?cislocclanku=2004082301>
- 12) Control Web 2000: popis [online]. Dostupný z URL: <http://krt32.krt.vslib.cz/cw/>
- 13) Popisy sběrnic RS 232 a RS 485 [online]. Dostupný z URL: <http://www.hw.cz>
- 14) Popis sběrnice CAN [online]. Dostupný z URL: <http://www.elektrorevue.cz/clanky/03021/>
- 15) Firemní dokumentace firmy Pepperl-Fuchs [online]. Dostupná z URL: <http://www.pepperl-fuchs.com/>

SEZNAM POUŽITÝCH ZKRATEK

A/Č	Analogově číslicový převodník
ASCII	kódování znaků (Američan Standard Code for Information Interchange)
ASI	Číslicová sběrnice pro propojení snímačů a akčních členů (Actuator Sensor Interface)
CAN	Číslicová sběrnice vhodná pro propojení snímačů a akčních členů (Control Area Network)
CAL	Norma aplikační vrstvy pro sběrnici CAN (Can Application Layer)
CIM	Počítačově integrované výrobní systémy (Computer Intergrated Manufacturing)
CPU	Procesor (Central Procesor Unit)
Č/A	číslicově analogový převodník
DDE	Protokol definovaný pro spolupráci aplikací prostřednictvím výměny dat a povelů (Dynamic Data Exchange)
HART	Protokol pro komunikaci inteligentních senzorů (Highway Addressable Remonte Transducer)
IEEE	Institute of Electrical and Electronics Engineers
ILAN	Průmyslová lokální síť (Industrial LAN)
ISO/OSI	Referenční síťový model (Reference Model for Open Systém Interconnection)
LAN	Lokální počítačová síť (Local Area Network)
OS	Operační systém
PC	Počítač IBM kompatibilní (Personál Computer)
PLC	Programovatelný logický automat (Programmable Logic Conrollers)
SDADA/MMI	Dohled nad řízením a získáváním dat/rozhraní člověk stroj (Supervisory Kontrol and Acquisition/Man-Machine Interface)

TCP/IP	Soubor protokolů pro heterogenní počítačové sítě (Transmission Control Protocol/Internet Protocol)
--------	--

SEZNAM OBRÁZKŮ

Obrázek 1: Referenční model ISO/ OSI	15
Obrázek 2: Příklad realizace fyzické vrstvy protokolu CAN	20
Obrázek 3: Principiální struktura sítě CAN podle ISO 11898	20
Obrázek 4: Datová zpráva podle specifikace CAN 2.0A	24
Obrázek 5: Začátek datové zprávy (standardní formát) podle specifikace 2.0B	25
Obrázek 6: Začátek datové zprávy (rozšířený formát) podle specifikace 2.0B	25
Obrázek 7: Chybová zpráva protokolu CAN	26
Obrázek 8: Zpráva o přetížení	27
Obrázek 9: Struktura a rozsah současné verze protokolu MODBUS	28
Obrázek 10: Příklad struktury propojení ModBus s různými rozhraními	29
Obrázek 11: Struktura ModBus rámce	29
Obrázek 12: Ukázka bezchybné transakce dat mezi klientem a serverem	30
Obrázek 13: Datový modem ModBus s oddělenými bloky a jeho adresování	32
Obrázek 14: Bitová reprezentace RTU Módu	34
Obrázek 15: Bitová reprezentace ASCII Módu	34
Obrázek 16: ModBus TCP/IP architektura komunikace	35
Obrázek 17: Princip zahrnutí rámce MODBUS do rámce TCP	36
Obrázek 18: Struktura stanice napojené na sběrnici ASI	38
Obrázek 19: Způsob kódování signálu	39
Obrázek 20: Definice vrstev standardu FF a vztah vůči modelu OSI	41
Obrázek 21: Kódování logických úrovní (FSK Bell 202)	42
Obrázek 22 : Poloha spektra signálu HARTv telefonním hlasovém kanálu	42
Obrázek 23: Rozhodovací úrovně vysílače a přijímače	45
Obrázek 24: Formát rámce zprávy	46
Obrázek 25: Základní bajtové pole	46
Obrázek 26: Rozmístění jednotlivých bitů identifikátoru	47
Obrázek 27: Komunikační struktura sběrnice	48
Obrázek 28: Struktura sběrnice M-Bus	49
Obrázek 29: Schéma rotační odparky	56
Obrázek 30: Schéma rotační odparky z hlediska snímání a regulace	57
Obrázek 31: Úvodní okno programu PACTware	58
Obrázek 32: Nastavení ovladače sériového rozhraní	59
Obrázek 33: Nastavení parametrů zařízení KSD2-GW-MOD	59
Obrázek 34: Nastavení parametrů zařízení KSD2-CI-S.H	60
Obrázek 35: Příklad prohlížení aktuální měřené hodnoty, pomocí servisního konektoru ..	60
Obrázek 36: Nastavení parametrů pomocí servisního programu	61
Obrázek 37: Záložka moduly	66
Obrázek 38: Záložka Textový editor	67
Obrázek 39: Záložka Datové inspektory	68
Obrázek 40: Záložka grafický editor	70
Obrázek 41: Záložka dokumentace	71
Obrázek 42: Průvodce - zvolení časování	72
Obrázek 43: Průvodce - název soboru a výchozího adresáře	72
Obrázek 44: Průvodce - zvolení velikosti obrazovky	73
Obrázek 45: Průvodce - nastavení možnosti počtu panelů	73
Obrázek 46: Průvodce - nastavení časová a startu aplikace	74
Obrázek 47: Aplikace pro řízení reálného procesu	76

SEZNAM TABULEK

tabulka 1: Srovnání parametrů RS 232C, RS 422A a RS 485	18
tabulka 2: Přehled některých parametrů	21
tabulka 3: Rozdělení oblastí datového modelu	31
tabulka 4: Porovnáním ModBusu, přes sériovou linku s modelem ISO/OSI	33
tabulka 5: Úrovně komunikačních signálů	43
tabulka 6: Kódování 1 bytu zprávy v protokolu HART	43
tabulka 7: Struktura zprávy	43
tabulka 8: Doporučené přenosové rychlosti	46
tabulka 9: Povolené rádiové frekvence pro Wi-fi v pásmu 2,4GHz	53

SEZNAM PŘÍLOH

P 1 :	Soubor Modbus.par.....	I
P 2 :	Soubor ModBus.dmf.....	III
P 3 :	Paměťové registry KSD2-GW-MOD	IV
P 4 :	CD ROM.....	VI

P 1 : Soubor Modbus.par

[comm]

baudrate = 19200

parity = even

databits = 8

stopbits = 1

rx_buffer = 1152

tx_buffer = 1152

rx_frame_buffer = 1152

tx_frame_buffer = 1152

cts_flow = false

dsr_flow = false

dtr_control = disable

rts_control = disable

dsr_sense = low

rx_interchar_timeout = 0

rx_char_timeout = 0

rx_timeout = 0

tx_char_timeout = 0

tx_timeout = 0

[Modbus]

ComDriver = cwcomm, com2

Mode = RTU

Timeout = 1000

EnableMonitor = true

DisablePresetSingleRegister = false

MaxRegistersInBlock = 100

OffsetFromOne = false

NumRepeat = 1

[Channels]

;----- =

;-----

;Block = Station, ChannelFrom, ChannelTo, Area, Offset, [Subtype], [Bidirect]

;----- =

;-----

Block = 01, 101, 103, 3x, 4611, int16,

Block = 01, 151, 152, 0x, 48,

P 2 : Soubor ModBus.dmf

begin

100 - 150 real input

151 - 170 boolean output

end.

P 3 : Paměťové registry KSD2-GW-MOD

Binární vstupy bez stavu:

	Reg.Tab.	Register address
Channel-based, max. 8 channels	2	$8 \times \text{RPI address} + \text{channel number} - 1$
Channel-based, max. 4 channels	2	$4 \times (\text{RPI address} - 3) + \text{channel number} + 1035$
Device-based, max. 8 channels, 1 device per register	4	$\text{RPI address} + 1536$
Device-based, max. 4 channels, 4 devices per register	4	$\text{Int}[(\text{RPI address} - 3)/4] + 9216$
Device-based, max. 8 channels, 2 devices per register	4	$\text{Int}[(\text{RPI address} - 2)/2] + 9345$
Device-based, max. 4 channels, 2 registers per device	4	
Device-based, max. 8 channels, 4 registers per device	4	

Binární vstupy se stavem:

	Reg.Tab.	Register address
Device-based, max. 4 channels, 2 registers per device	4	$2 \times (\text{RPI address} - 3) + 3587$
Device-based, max. 8 channels, 4 registers per device	4	$4 \times (\text{RPI address} - 3) + 3084$

Binární výstupy bez stavu:

	Reg.Tab	Register address
Channel-based, max. 8 channels	1	$8 \times \text{RPI address} + \text{channel number} - 1$
Channel-based, max. 4 channels	1	$4 \times (\text{RPI address} - 3) + \text{channel number} + 1035$
Device-based, max. 8 channels, 2 devices per register	3	$\text{Int}[(\text{RPI address} - 2)/2] + 9345$
Device-based, max. 4 channels, 4 devices per register	3	$\text{Int}(\text{RPI address}/4) + 9216$

Binární výstupy se stavem:

	Reg.Tab	Register address
Device-based, max. 4 channels, 2 registers per device	3	$2 \times (\text{RPI address} - 3) + 3587$
Device-based, max. 8 channels, 4 registers per device	3	$4 \times (\text{RPI address} - 3) + 3084$

Spojitě vstupy bez stavu:

	Reg.Tab	Signed integer format	Floating point format
Analogue inputs, 1 channel device	4	$\text{RPI address} + 4608$	$2 \times (\text{RPI address} - 3) + 6150$
Analogue inputs, 2 channel device	4	$2 \times (\text{RPI address} - 3) + 5126$	$4 \times (\text{RPI address} - 3) + 7052$
Analogue inputs, 3 channel device	4	$3 \times (\text{RPI address} - 3) + 12$	$6 \times (\text{RPI address} - 3) + 8210$

Spojité vstupy se stavem:

	Reg.Tab.	Signed integer format	Floating point format
Analogue inputs, 1 channel device	4	$2*(RPI\ address-3)+4742$	$3*(RPI\ address-3)+6409$
Analogue inputs, 2 channel device	4	$3*(RPI\ address-3)+5641$	$5*(RPI\ address-3)+7567$
Analogue inputs, 3 channel device	4	$6*(RPI\ address-3)+2072$	$8*(RPI\ address-3)+536$

Spojité výstupy bez stavu:

	Reg.Tab.	Signed integer format	Floating point format
Analogue outputs, 1 channel device	3	$RPI\ address+4608$	$2*(RPI\ address-3)+6150$
Analogue outputs, 2 channel device	3	$2*(RPI\ address-3)+5126$	$4*(RPI\ address-3)+7052$
Analogue outputs, 3 channel device	3	$3*(RPI\ address-3)+12$	$6*(RPI\ address-3)+8210$

Spojité výstupy se stavem:

	Reg.Tab.	Signed integer format	Floating point format
Analogue outputs, 1 channel device	3	$2*(RPI\ address-3)+4742$	$3*(RPI\ address-3)+6409$
Analogue outputs, 2 channel device	3	$3*(RPI\ address-3)+5641$	$5*(RPI\ address-3)+7567$
Analogue outputs, 3 channel device	3	$6*(RPI\ address-3)+2072$	$8*(RPI\ address-3)+536$

Speciální případy – Binární vstupy nebo výstupy:

Binary inputs with status, max. 4 channels	$2*(RPI\ address-3)+2751$
Binary outputs with status, max. 4 channels	$2*(RPI\ address-3)+2501$

Speciální případy – Analogové vstupy nebo výstupy a informace o stavu

	Signed integer	Floating point
Analogue inputs, 1 channel device	$RPI\ address+1248$	$2*(RPI\ address-3)+1376$
Analogue inputs, 2 channel device	$2*(RPI\ address-3)+1626$	$4*(RPI\ address-3)+1876$
Analogue outputs, 1 channel device	$RPI\ address\ 2$	$2*(RPI\ address-3)+126$
Analogue outputs, 2 channel device	$2*(RPI\ address-3)+376$	$4*(RPI\ address-3)+626$
Status information	$RPI\ address + 1126$	

Ostatní dostupné adresy v registru:

	Reg.Tab.	Register address
Device ID	3	$RPI\ address + 4096$
Device type	3	$RPI\ address + 4224$
Device diagnosis	4	$RPI\ address + 4225$
Gateway diagnosis, wordwise	4	8, 528, 1538, 2064, 3080 and 4225
Gateway diagnosis, bitwise	2	8 - 23

P 4 : CD ROM

Obsahuje:

- samotnou diplomovou práci v PDF a MS Word formátu
- katalogové listy zařízení
- programy Adobe Akrobat, Control Web 5 – demo a PactWare - basic
- popis sběrnice ModBus - manuál