

JIHOČESKÁ UNIVERZITA
v Českých Budějovicích

PEDAGOGICKÁ FAKULTA

Katedra fyziky

**Vývojový software průmyslových automatizačních řídicích systémů
SIEMENS, ALLEN-BRADLEY a SAUTER**

Bakalářská práce

Vedoucí práce: Ing. Michal Šerý

Autor: David Kocum

Anotace

Tato bakalářská práce je zaměřena na technický a softwarový popis řídicích systémů německé firmy Siemens, americké firmy Allen Bradley a švýcarské firmy Sauter. Součástí práce je vytvoření vzorové aplikace spuštění třífázového asynchronního motoru způsobem hvězda-trojúhelník ve všech třech vývojových prostředích od jednotlivých výrobců zařízení.

Abstrakt

This thesis is intent on technical and software description of control systems German firm Siemens, American firm Allen Bradley and Swiss firm Sauter. Work includes exemplary application activation three-phase induction motor in a way star - triangle in all of three development computer programmes from single producer of equipments.

**Prohlašuji, že předloženou práci jsem vypracoval samostatně a použil jen podkladů,
které jsou uvedeny v seznamu použité literatury.**

V Českých Budějovicích dne 23. listopadu 2006

.....

Děkuji Ing. Michalovi Šerému za odbornou pomoc a vedení při vypracování této bakalářské práce.

Obsah

1	ÚVOD	6
2	HARDWARE FIRMY SIEMENS	8
2.1	ŘADA SIMATIC S7-200	9
2.1.1	Centrální jednotky	9
2.1.2	Digitální vstupy a výstupy	9
2.1.3	Analogové vstupy a výstupy	10
2.1.4	Operátorské panely	11
2.1.5	Komunikace	11
2.2	ŘADA SIMATIC S7-300	12
2.2.1	Centrální procesorové jednotky	12
2.2.2	Přídavné moduly	13
2.2.3	Komunikace	14
3	HARDWARE FIRMY ALLEN BRADLEY	14
3.1	ŘADA SLC 500	15
3.1.1	Centrální procesorové jednotky	15
3.1.2	Přídavné moduly	16
3.1.3	Komunikace	17
4	HARDWARE FIRMY SAUTER	17
4.1	SAUTER ŘADA EY3600	17
5	NORMA IEC 61131	19
5.1	STRUKTURA A HISTORIE IEC 61131	19
5.2	NORMA IEC EN 61131-3	20
5.2.1	Jazyk příčkového diagramu LD	20
5.2.2	Jazyk funkčního blokového schématu FBD	20
5.2.3	Jazyk seznamu instrukcí IL	21
5.2.4	Jazyk strukturovaného textu ST	21
5.2.5	Sekvenční funkční diagram SFC	22
6	VÝVOJOVÉ PROSTŘEDÍ STEP 7 – MICRO/WIN	22
6.1	PŘÍSTUP K DATŮM	23
6.2	PŘÍSTUP K DATŮM V PAMĚŤOVÝCH OBLASTECH	24
6.3	ZÁKLADNÍ PRVKY PROGRAMU	26
6.4	POPIS VÝVOJOVÉHO PROSTŘEDÍ	27
6.5	PŘÍKLAD VLOŽENÍ PŘÍKAZU V EDITORU LAD	30
6.6	NEJPOUŽÍVANĚJŠÍ PŘÍKAZY	31
6.6.1	Instrukce bitové logiky – kontakty	31
6.6.2	Instrukce bitové logiky – cívky	32
6.6.3	Instrukce porovnání řetězců	32
6.6.4	Instrukce porovnání číselných hodnot	33
6.6.5	Instrukce logických operací – AND, OR, XOR	33
6.6.6	Instrukce logických operací – Inverze	34
6.6.7	Instrukce časovače	34
6.6.8	Instrukce přesunu	34
7	VYTVOŘENÍ VZOROVÉ APLIKACE	35
7.1	ZADÁNÍ APLIKACE	35
7.2	ŘEŠENÍ APLIKACE	36
7.3	ZPRACOVÁNÍ VE VÝVOJOVÉM PROSTŘEDÍ MICRO/WIN	37
7.4	ZPRACOVÁNÍ VE VÝVOJOVÉM PROSTŘEDÍ RSLOGIX 500	38
7.5	ZPRACOVÁNÍ VE VÝVOJOVÉM PROSTŘEDÍ EY3600-CASE	39
8	ZÁVĚR	40
9	POUŽITÁ LITERATURA	41

1 ÚVOD

Rychlý rozvoj výpočetní techniky a informačních technologií přináší i do oblasti průmyslové automatizace nejen dílčí vylepšení tradičních postupů a technologií, navíc však poskytuje i zcela nové možnosti.

Do provozu jsou uváděny komplexní architektury automatizovaných řídicích systémů pro ovládání složitých technologií jednotlivých výrobních linek i celých závodů. Vyspělá průmyslová výroba je však velmi komplikovaný celek, ve kterém z dlouhodobého hlediska nikdy neprobíhají všechny činnosti přesně podle původních předpokladů.

Obecně lze konstatovat, že všechny softwarové produkty procházejí rychlým a neustálým vývojem, ve kterém se průběžně odráží vznik nových technologií v tomto oboru.

Ani automatizační software není výjimkou, i když zavádění nových softwarových technologií není v tomto tržním sektoru obvykle tak rychlé jako v případě softwaru používaného pro administrativní účely.

Tento konzervatismus je pochopitelný, uvážíme-li, že jde o ovládání výrobních technologií, při jejichž případné poruše by mohly vzniknout značné materiální škody nebo nastat havárie ohrožující životy lidí. Průmyslová automatizace tudíž nutně vyžaduje pouze spolehlivé a důkladně ověřené informační technologie.

Jelikož však mnoho informací z výrobních procesů není určeno pouze úzkému okruhu operátorů, je výhodou, jsou-li v dodaném řešení použity otevřené a všeobecně používané technologie, které mohou standardními způsoby předávat potřebné datové údaje také dalším uživatelům v podnikové hierarchii.

Při vývoji sofistikovaných softwarových aplikací pro tento tržní segment však nestačí jen znalost moderních informačních technologií, ale jejich dodavatel by měl vyhovět i mnoha dalším požadavkům zákazníků, kteří ve svých závodech často provozují zdánlivě nekompatibilní automatizační řídicí systémy od různých výrobců a na různých výrobních technologiích.

Automatizační řídicí systémy se vyvinuly z počítačů v 70. letech minulého století. Důvodem vzniku byla malá odolnost počítačů v průmyslovém prostředí.

Automaty jsou odolnější především v těchto oblastech:

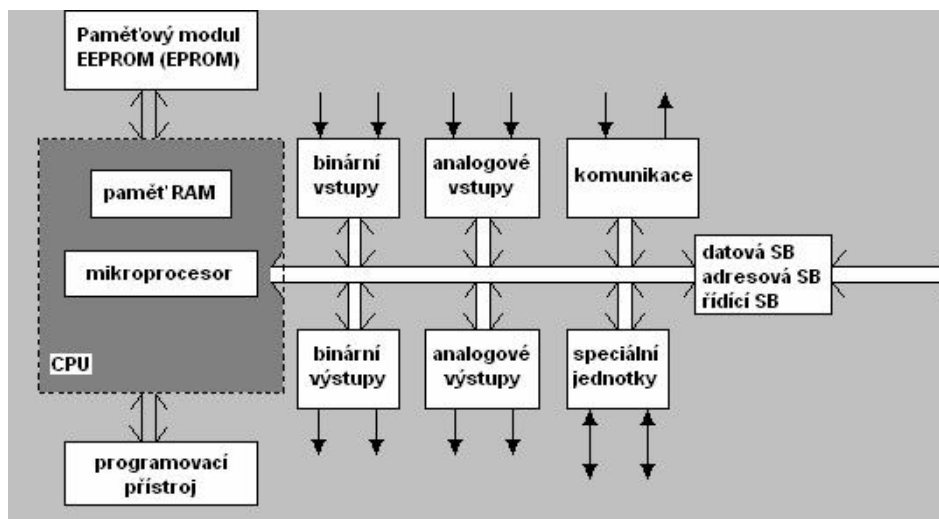
Vyšší odolnost vůči vibracím - například u počítače by nevydržel pevný disk a také jeho celková konstrukce je stavebnicová a při vibracích se "rozpadá". U automatu disk není a jeho mechanická konstrukce je pevnější.

Vyšší odolnost vůči prachu a vlhkosti - do počítače je prach vháněn ventilátorem a zde se usazuje. V kombinaci s vysokou vlhkostí může způsobit zkrat. U automatů ventilátor není, protože mají podstatně nižší spotřebu a není je tedy nutno chladit.

Vyšší odolnost při výkyvech napájení - elektrická síť může kolísat, například při rozběhu nebo zastavení stroje který odebírá větší množství proudu. Zdrojové jednotky jsou dělány tak, aby vydržely některé proudové špičky a záložní baterie je schopna nahradit výpadky proudu.

Vyšší odolnost vůči elektromagnetickým polím - u automatu není tolik součástek, které by se daly zmagnetizovat a většina výrobců se snaží tyto prvky odstínit. Všechny nevýhody počítačů se dají vylepšit. Odpružení disku, vzduchotěsné oddělení od okolního prostředí s vlastním chlazením, ale narůstá nám zde cena.

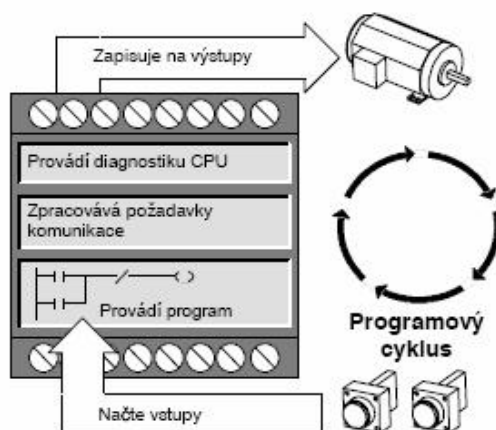
I automaty mají své nevýhody, jednou z hlavních je nižší rychlost oproti počítačům. Současné automaty například nedokáží pracovat s obrazem. U takovýchto výrob, například osazování desek plošných spojů, musíme postupovat opačně, to znamená přizpůsobovat výrobu průmyslovým počítačům. U této výroby nemáme problém s vibracemi, ale snažíme se o čisté prostředí bez prachu. Počítače mají vlastní elektrický okruh, který je zálohován. Kromě vysoké odolnosti hardwarové mají automaty i vyšší odolnost softwarovou. Software je vytvořený přímo pro daný automat a je stabilnější než běžné OS u počítačů.



Obrázek č. 1: Obecná struktura průmyslového automatu

Dříve se používaly průmyslové automaty jen pro logické řízení, tj. zpracování dvou stavových veličin.

Dnešní průmyslové automaty jsou rozšířeny o regulační funkce. Zpracovávají i signály analogové, eventuálně i číslicové. Jádrem průmyslového automatu je mikropočítač, tj. mikroprocesor + paměť + vstupní/výstupní obvody. Mikroprocesor vykonává veškeré logické a aritmetické operace. Uživatel nemusí znát mikroprocesor ani použitý OS. Chod průmyslového automatu je řízen systémovým SW od výrobce tak, aby uživatelský SW byl co nejjednodušší. Program je psán ve speciálním jazyce, přizpůsobeným pro daný automat. Nezbytným příslušenstvím průmyslového automatu je programovací prostředí sloužící k editaci a vytváření programů. Dále k monitorování a ke kontrole chodu průmyslového automatu, k dokumentaci, k manipulaci s programy. Univerzálnost automatu je dosažena volitelným počtem a sortimentem vstupně/výstupních jednotek. Pro náročné a speciální algoritmy jsou specializované jednotky - "inteligentní periferie"; například ovládání krokových motorků, vícenásobné regulátory.



Obrázek č. 2: Obecný časový chod programu průmyslového automatu

2 HARDWARE FIRMY SIEMENS

Ještě nedávno byly rozvaděče vybavovány téměř výhradně standardními jednoúčelovými spínacími přístroji. Jedná se o časová a pomocná relé, ministrykače a převodníky signálů. Pro tato řešení má Siemens stále v nabídce celou škálu přístrojů různých napětových tříd i výkonů. SIMIREL jsou přístroje pro spínání, monitorování a převod signálů. Jedná se o relé, časová a termistorová relé, převodníky, ovládací relé. SIRIUS je řada nízkonapětových spínacích prvků až do výkonu 250 kW. Řada obsahuje motorové jističe, stykače, nadproudová elektronická i standardní relé a kompaktní softstartéry.

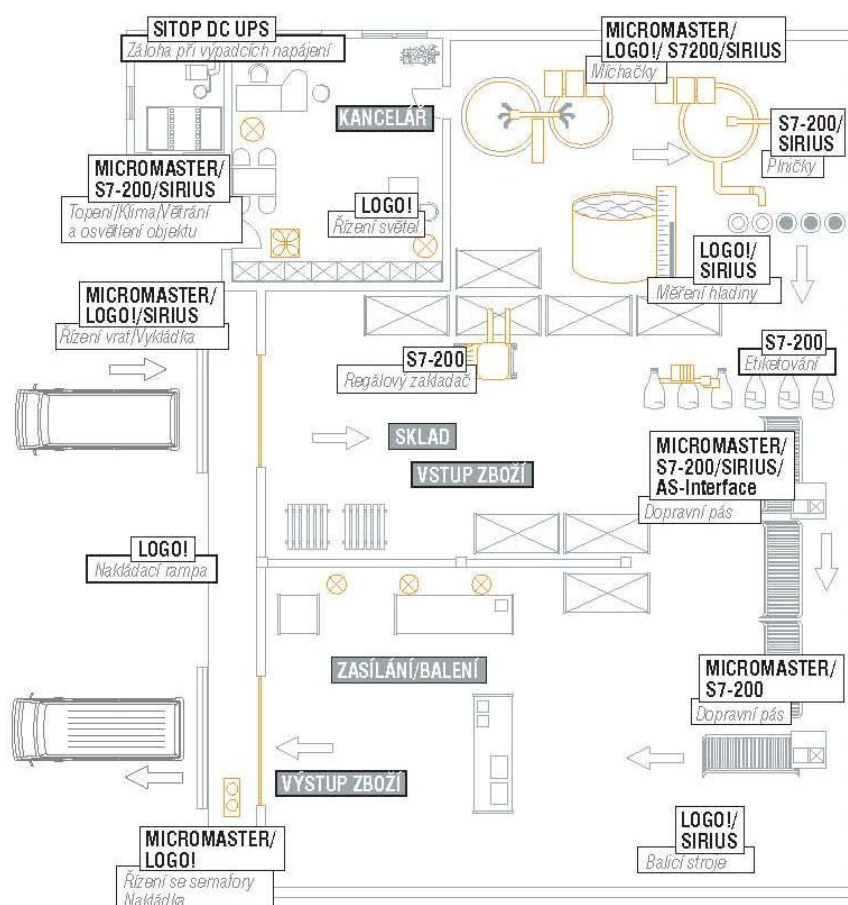
Již v roce 1996 vytvořila firma Siemens novou třídu řídicích prvků, kterou pojmenovala mikrosystémy. Dnes je to již sedmá generace tohoto zařízení, která je neustále vylepšována.

V rámci automatizační techniky firma SIEMENS prosazuje myšlenku tzv. plně integrované automatizace (Totally Integrated Automation). Tento pojem se vztahuje na vlastní řídicí členy (PLC /průmyslová PC), dále na přístroje pro ovládání a sledování procesu (HMI – Human Machine Interface) a síťová řešení.

Klíčovou vlastností je tzv. trojí průchodnost systému:

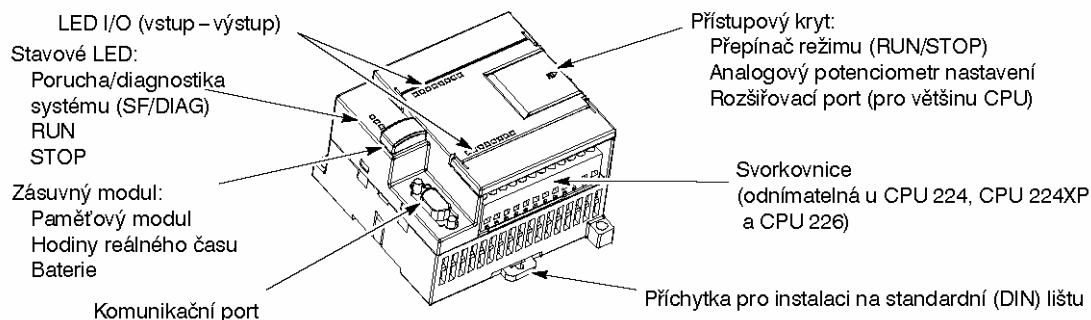
- sdílená databáze objektů řídicího systému
- síťové propojení mezi hierarchickými úrovněmi
- jednotnost softwarových prostředků pro konfiguraci celého systému

Sama tato myšlenka naznačuje, že zařízení, která takové požadavky splňují, nemohou být zároveň vzorem jednoduchosti. Naopak je možné pomocí těchto zařízení zvládnout i ty nejnáročnější aplikace.



Obrázek č. 4: Příklad návržení aplikací v průmyslu

2.1 ŘADA SIMATIC S7-200



Obrázek č. 5: Popis jednotlivých částí PLC SIMATIC S7-200

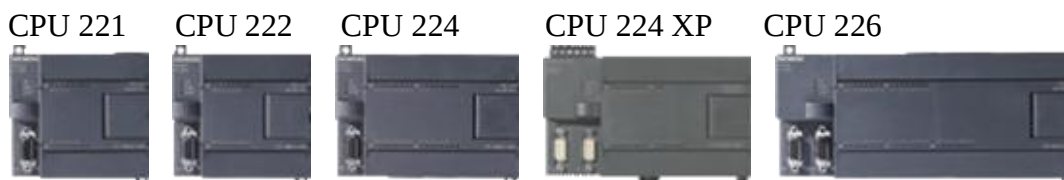
SIMATIC S7-200 je řada malých programovatelných automatů (mikro-PLC) určených k řízení v jednodušších automatizačních aplikacích. S7-200 sleduje stav vstupů a podle uživatelského programu řídí výstupy. Uživatelský program může obsahovat Booleovu logiku, čítače, časovače, složité matematické operace a komunikaci s jinými inteligentními zařízeními.

Významné vlastnosti S7-200:

- malý a kompaktní design
- výkonná instrukční sada stejná pro všechny modely CPU
- systém časových přerušení i přerušení od události
- vysokorychlostní čítače do 200 kHz a pulzní výstupy do 100 kHz

2.1.1 Centrální jednotky

Na výběr je pět centrálních jednotek, které se liší v možnostech dalšího rozšíření, ve velikosti paměti pro program/data a v počtu integrovaných vstupů/výstupů.



Obrázek č. 6: Jednotlivé typy PLC SIMATIC S7-200

Centrální jednotku je možné rozšířit určitým počtem rozšiřovacích modulů, kterých existuje několik typů.

2.1.2 Digitální vstupy a výstupy



Obrázek č. 7: Jednotlivé typy digitálních vstupních a výstupních modulů

Využití pro rozšíření počtu digitálních vstupů a výstupů. K dispozici je několik variant vstupních, výstupních a kombinovaných vstupně/výstupních modulů.

Varianty :

EM 221 (vstupní)

- 8 digitálních vstupů 24 V DC
- 8 digitálních vstupů 120/230 V AC
- 16 digitálních vstupů 24 V DC

EM 222 (výstupní)

- 4 digitální tranzistorové výstupy 24 V DC, 5A
- 4 digitální releové výstupy, 10 A
- 8 digitálních výstupů 24 V DC
- 8 digitálních výstupů 120/230 V AC

EM 223 (kombinovaný)

- 4 digitální vstupy 24 V DC + 4 digitální tranzistorové výstupy
- 4 digitální vstupy 24 V DC + 4 digitální releové výstupy
- 8 digitálních vstupů 24 V DC + 8 digitálních tranzistorových výstupů
- 8 digitálních vstupů 24 V DC + 8 digitálních releových výstupů
- 16 digitálních vstupů 24 V DC + 16 digitálních tranzistorových výstupů
- 16 digitálních vstupů 24 V DC + 16 digitálních releových výstupů

2.1.3 Analogové vstupy a výstupy



Obrázek č. 8: Jednotlivé typy analogových vstupních a výstupních modulů

Využití pro rozšíření S7-200 o analogové vstupy a výstupy. Je k dispozici několik variant analogových modulů.

Vstupní moduly jsou s napěťovými nebo proudovými rozsahy, existují i speciální vstupní moduly určené pro připojení odporových snímačů a termočlánků s různými rozsahy.

Dále je k dispozici výstupní a kombinovaný vstupně/výstupní modul.

Varianty:

EM 231 (vstupní)

- 4 analogové vstupy / 12 bitů
- 4 analogové vstupy / 15 bitů + snímač pro měření odporů
- 4 analogové vstupy / 15 bitů + snímač pro termočlánky

EM 232 (výstupní)

- 2 analogové výstupy / 12 bitů

EM 235 (kombinovaný)

- 4 analogové vstupy / 12 bitů + 1 analogový výstup / 12 bitů

2.1.4 Operátorské panely

Operátorský panel TD 200



Obrázek č. 9: Operátorský panel TD 200

- nejjednodušší textový displej pro S7-200
- zobrazuje textové zprávy, změny v řídicím programu, nastavení vstupů a výstupů
- uživatelské menu se dvěma úrovněmi (64 obrazovek menu)
- přímé propojení s CPU pomocí dodávaného kabelu nebo zapojení do sítě (také pomocí EM 277)
- není potřeba samostatný parametrizační software, konfigurace probíhá v prostředí Micro/WIN
- nastavení adresy a kontrastu přímo v menu přístroje

Operátorský panel TP 170 micro



Obrázek č. 10: Operátorský panel TP 170 micro

- dotykový operátorský panel pro ovládání a monitorování menších strojů a strojních zařízení
- levné řešení s grafickým zobrazováním všech potřebných funkcí pro jednodušší úlohy
- bodový 5.7" STN dotykový displej, modré zobrazování (4 úrovně)
- vektorová grafika
- instalace možná jak horizontálně, tak i vertikálně
- on-line přepínání mezi 5 jazyky
- určen pro Simatic S7-200, komunikace s PLC je zajištěna přes integrované PPI rozhraní

2.1.5 Komunikace

Siemens nabízí dvě programovací možnosti připojení počítače k S7-200: přímé připojení pomocí PPI Multi-Master kabelu nebo kartou komunikačního procesoru (CP) a MPI kabelem.

PPI Multi-Master programovací kabel je nejběžnější a nejekonomičtější způsob připojení počítače k S7-200. Tímto kabelem se propojí komunikační port S7-200 se sériovým komunikačním portem počítače. PPI Multi-Master programovací kabel se může rovněž použít pro připojení dalších zařízení k S7-200.

RS-485

Integrovaný port RS-485 umožňuje datové přenosy až do rychlosti 187,5 kbit/s. Systémová sběrnice může mít až 126 stanic. Takto můžete mít bez problémů propojen dostatečný počet řídicích stanic včetně operátorských panelů.

PPI

Automaty S7-200 je možné propojovat vzájemně pomocí protokolu PPI (point-to-point). Je také možné spojit S7-200 s vyššími automaty S7-300/400. V tomto případě se chová S7-200 jako MPI slave.

FREEPORT

Pomocí volně programovatelného sériového protokolu (tzv. Freeport) je možné spojit S7-200 s libovolným zařízením, které komunikuje po sériové lince. Takto je možné připojit např. tiskárnu, GSM modem, čtečku čárového kódu apod.

ETHERNET

Pro spojení automatu s průmyslovým Ethernetem slouží rozšiřovací modul CP 243-1. Tento modul otevírá spoustu možností, neboť s jeho použitím je možné komunikovat prakticky s kteroukoliv stanicí na síti Ethernet, ať už je to jiný automat S7-200 nebo S7-300/400 popř. PC s OPC serverem nebo vizualizačním nástrojem WinCC.

2.2 ŘADA SIMATIC S7-300

Průmyslový řídicí systém Simatic S7-300 je z široké nabídky společnosti Siemens nejprodávanější. Je určen pro realizaci velmi rozmanitých automatizačních úloh středního rozsahu. Je univerzální automatizační platformou pro systémová řešení s hlavním důrazem na výrobní techniku. Představuje optimální volbu pro centrální i distribuované řízení.



Obrázek č. 11: PLC SIMATIC S7-300

2.2.1 Centrální procesorové jednotky

Co se týče nabídky centrálních procesorových jednotek (CPU) pro Simatic S7-300, může si uživatel vybírat v těchto kategoriích:

- standardní CPU
- kompaktní CPU
- bezpečnostní CPU (F-systémy)
- technologické CPU
- kompletní přístroje Simatic C7

Standardní CPU

V kategorii standardních CPU lze volit z několika typů. Všechny jednotky jsou standardně osazeny programovacími a komunikačními rozhraními MPI, v některých je zabudováno i rozhraní Profibus (typy 315-2DP, 317-2DP, 318-2DP). Novým trendem v současné automatizaci je orientace na standard Ethernet i ve výrobních provozech. Tomu plně vyhovují nové CPU s integrovaným ethernetovým rozhraním (315-2PN/DP, 317-2PN/DP). Právě díky nim je nyní připojení a obsluha distribuovaných jednotek přes Ethernet jednodušší a zmíněné jednotky lze přes toto rozhraní rovněž programovat. Parametry jednotlivých typů CPU jsou rovnoměrně odstupňovány tak, aby si uživatel mohl velmi snadno vybrat vhodnou jednotku pro danou automatizační úlohu.

Kompaktní CPU

Jako kompaktní se označují CPU doplněné digitálními a analogovými I/O a nejčastěji vyžadovanými základními technologickými funkcemi jako rychlé čítání, měření frekvence, nastavování polohy a PID regulace. Všechny kompaktní CPU jsou standardně vybaveny komunikačním rozhraním MPI. Výkonnější procesorové jednotky navíc mají rozhraní Profibus (313C-2DP, 314C-2DP) nebo RS-422, RS-485 (313C-2PtP, 314C-2PtP). Jsou cenově velmi výhodné pro úlohy, které vystačí s příslušným počtem vstupů a výstupů. Jinak k nim lze snadno připojovat další moduly vstupů a výstupů ve stejném rozsahu jako u standardní CPU.

Bezpečnostní CPU (F-systémy)

Bezpečnostní systémy se používají všude tam, kde je třeba zajistit co nejvyšší stupeň bezpečnosti obsluhy, výrobního zařízení či okolního prostředí – např. je-li se třeba bez ohledu na náklady vyvarovat nehod a poškození zdraví či životního prostředí v důsledku poruchy. Uživatel může vytvářet bezpečnostní řídicí systémy v centrálním i distribuovaném provedení. Hlavním znakem je spojení standardní provozní automatizace a bezpečnostní techniky do jediného systému. To znamená, že po síti Profibus-DP mezi centrálním řídicím systémem a distribuovanými moduly I/O probíhá nejen „běžná“ komunikace, ale také bezpečnostně orientovaná komunikace (použití profilu Profisafe) a není nutná žádná samostatná bezpečnostní komunikační linka. Toto spojení standardní a bezpečnostně orientované automatizace značně snižuje výdaje na moderní zabezpečené provozy. V současné době nabízí firma Siemens dva typy bezpečnostních CPU, a to 315F-2DP a 317F-2DP. Jejich vynikající parametry poskytují uživateli velmi výkonné CPU pro bezpečnostní aplikace střední třídy. Bezpečnostně orientovaný program se vytváří ve standardních programovacích jazycích reléových schémat (LD) a funkčních bloků (FBD) podle IEC 61131-3 při použití certifikovaných příkladů, které jsou k dispozici ve speciální, tzv. F-knihovně. Pro decentralizované struktury jsou k dispozici bezpečnostní komponenty z řad ET 200S a ET 200M.

Technologické CPU

V technologické CPU 317T-2 DP jsou přímo začleněny výkonné technologické funkce a funkce pro řízení pohybu. Je navržena pro dynamické řízení pohybu v několika osách současně. Předprogramované funkce pro řízení pohybu podle standardu vydaného organizací PLCopen, integrované I/O, izochronní režim sběrnice Profibus-DP – to vše přispívá k pohodlnému a flexibilnímu řízení pohybu současně v několika osách (např. nastavování polohy, synchronizace a spínání s použitím vaček). Osy lze snadno konfigurovat a parametrizovat ve vývojovém prostředí STEP 7.

Kompletní přístroje (Simatic C7)

Kategorie kompletních přístrojů představuje z technického i cenového hlediska ideální spojení řídicího systému a jeho operátorského rozhraní. Jednotlivé typy CPU jsou doplněny (vedle analogových a digitálních I/O) textovými nebo grafickými displeji s vynikající zobrazovací schopností a dlouhou dobou života. Řídicí systém se ovládá prostřednictvím membránové klávesnice nebo dotykové obrazovky (podle typu). Výkonnost jednotlivých typů je opět odstupňována tak, aby bylo možné snadno najít odpovídající typ. Právě nyní jsou k dispozici produkty s označením C7-613, C7-635 a C7-636. Tyto jednotky lze dále rozšířit o standardní I/O moduly řady S7-300, a to hned několika způsoby.

2.2.2 Přídavné moduly

Celý řídicí systém je tvořen zdrojem napájení a sestavou přídavných karet (modulů), které doplňují jednotku CPU. Všechny moduly lze jednoduše instalovat na profilovou lištu. K dispozici je široká paleta různých modulů, které umožňují RS optimálně přizpůsobit požadavkům řízeného procesu či technologie:

- signálové moduly (SM): DI/DO, AI/AO pro všechny typy běžných procesních signálů
- moduly rozhraní (IM): pro víceřadá uspořádání
- funkční moduly (FM): pro zpracování komplexních nebo časově kritických procesů nezávisle na CPU, např. rychlé čítání, měření frekvence, polohování, PID algoritmy
- komunikační procesory (CP): zprostředkovávají propojení různých sběrnicových systémů nebo spojení s dalšími přístroji sériovou linkou

2.2.3 Komunikace

MPI

MPI (multi point interface) je úsporné řešení pro komunikaci s programovacími přístroji a PC, HMI® systémy a dalšími řídicími systémy SIMATIC S7/C7/WinAC. Celkem lze propojit 125 MPI stanic s přenosovou rychlostí 187.5 kbit/s, např. pro výměnu procesních dat mezi různými řídicími systémy nebo využít pro spojení HMI služby bez jakéhokoliv programování.

Profibus DP

Pro optimální konfiguraci rozsáhlejších distribuovaných sítí lze SIMATIC S7-300 napojit na PROFIBUS DP (dle EN 50170). Tím se otevírají komunikační možnosti i pro další partnery. Od řídicích systémů SIMATIC až po přístrojovou techniku od jiných výrobců. Je možné konfigurovat také komunikace se stávajícími systémy předchozí generace SIMATIC S5 nebo SIMATIC 505. Distribuované v/v lze konfigurovat pomocí vývojového prostředí STEP 7 úplně stejně jako centralizované. SIMATIC S7-300 lze použít jako master i jako slave. Podpora standardu DP V1 dovoluje programování a optimalizaci polních přístrojů během provozu, což znamená kratší časy nastavení přístroje.

Ethernet (Profinet)

Nová CPU s integrovaným rozhraním PROFINet jsou určena pro budoucí trend automatizace založené na komponentech (CbA), lze je přes toto rozhraní programovat a přes Ethernet přistupovat k jednotkám HMI. Tím lze uspořit nejen prostředky za komunikační procesor, který je k těmto účelům jinak vyžadován, ale i místo v rozvaděči.

3 HARDWARE FIRMY ALLEN BRADLEY

Firma byla založena v roce 1903. V roce 1985 koupila firmu Allen-Bradley společnost Rockwell International. Po začlenění Reliance Electric k firmě Rockwell International vznikl v roce 1995 základ silné skupiny vyrábějící prostředky průmyslové automatizace. Zaměření na automatizaci nakonec u firmy Rockwell International převážilo. Po prodeji aktivit v oblasti avioniky vznikla v roce 2001 firma Rockwell Automation. Sídlem Rockwell International a později Rockwell Automation je od roku 1999 Milwaukee, rodiště firmy Allen-Bradley.



Obrázek č. 12: PLC SLC 500

3.1 ŘADA SLC 500

Řídicí systém SLC 500 je programovatelný logický automat firmy Allen Bradley (AB). Řídicí systém se vyrábí ve dvou hardwarových kategoriích. První kategorií jsou pevné automaty, druhou modulární automaty. Modulární automaty jsou v současnosti velmi rozšířené, a to hlavně díky své flexibilitě. Z jednotlivých modulů je možné sestavit automat určený na míru pro svoji řízenou aplikaci.

3.1.1 Centrální procesorové jednotky

Procesory pro modulární automat SLC 500 se vyrábějí v pěti variantách, vzájemně se lišících zejména v komunikačních protokolech, velikosti paměti, rychlosti procesoru a instrukční sadou. Jednotlivé varianty mají označení SLC 5/01, SLC 5/02, SLC 5/03, SLC 5/04, SLC 5/05.



Obrázek č. 13: Varianty procesorových jednotek pro SLC 500

SLC 5/01

Procesor SLC 5/01 je používán v modulární hardwarové konfiguraci a poskytuje:

- dvě možné velikosti programové paměti - 1K nebo 4K instrukcí
- ovládání až 3840 vstupů a výstupů
- výkonnou instrukční sadu určenou pro programování v příčkové logice
- podprogramy
- komunikační kanál DH-485, umožňující komunikaci mezi dvěma uzly (peer-to-peer, pouze pro odezvy).
- zálohování paměti procesoru L511 kondenzátorem nebo baterií u L514

SLC 5/02

Procesory SLC 5/02 nabízejí rozšířenou instrukční sadu, zlepšenou diagnostiku, rychlejší vykonávání instrukcí a oproti procesoru SLC 5/01 rozšířené komunikační možnosti.

SLC 5/03

Procesory SLC 5/03 značně zvyšují výkonnost systému, a to zkrácením doby vykonávání instrukcí na 1 ms pro 1K instrukcí typického uživatelského programu. To je předurčuje pro použití ve vysokorychlostních aplikacích, jako je vysokorychlostní balení, třídění a manipulace materiálem. S možností on-line editace představuje procesor 5/03 skvělé řešení pro řízení spojitých procesních operací bez nutnosti přerušení činnosti. Zabudovaný komunikační kanál RS-232C poskytuje flexibilní spojení s externími inteligentními zařízeními bez nutnosti nasazení dalších modulů

SLC 5/04

Procesory SLC 5/04 poskytují stejné funkční vlastnosti jako procesor SLC 5/03 a navíc komunikaci po síti DH+ místo DH-485. Komunikace po DH+ je 3–12krát rychlejší než komunikace po síti DH-485 a zvyšuje tak výkon systému. Navíc je procesor SLC 5/04 přibližně o 15 % rychlejší než procesor 5/03.

SLC 5/05

Procesory SLC 5/05 poskytují obdobné funkční vlastnosti jako procesor SLC 5/04 a navíc komunikaci po síti Ethernet místo DH+. Síť Ethernet s přenosovou rychlostí 10Mb/s je vysoce výkonná komunikační síť a lze ji použít jak pro nahrání programu do procesoru, on-line editaci, tak pro zasílání zpráv, výměnu dat s operátorskými rozhraními (např.s vizualizačním softwarem RSVIEW 32). Různé velikosti pamětí umožňují zvolit konfiguraci přesně podle požadavku aplikace.

3.1.2 Přídavné moduly

Do dalších pozic pro moduly se mohou již umisťovat libovolné moduly dle uvážení návrháře. Takovými moduly mohou být například moduly diskrétních vstupů, výstupů, případně kombinované moduly. Tyto jednotky se vyrábějí v široké řadě se 4, 8, 12, 16 a 32 vstupy (výstupy) a mohou zpracovávat jak střídavé, tak stejnosměrné napětí v různých napěťových rozsazích. Stav vstupu a výstupu je na každé jednotce signalizován sérií LED diod.



Obrázek č. 14: Standardní digitální vstupně/výstupní modul

Dalšími jednotkami používanými při řízení složitějších technologií jsou analogové jednotky. Jednak to mohou být jednotky pracující přímo s napětím, proudem ve standardizovaném rozsahu, účelové analogové jednotky určené pro měření teploty nebo odporu.



Obrázek č. 15: Standardní analogový vstupně/výstupní modul

Pro měření rychle se měnících diskrétních signálů jsou určeny čítací jednotky a s nimi do jedné kategorie spadající polohovací a pohybové moduly. Tyto moduly jsou určeny pro řízení polohovacích zařízení.

Poslední kategorií modulů jsou speciální moduly zajišťující nejrůznější operace a služby. Do této kategorie patří například: komunikační jednotky, moduly pro řízení vstřikovacích lisů, ale i moduly rozšiřující schopnosti automatu o možnost programování v jazyce BASIC. Modulární systém SLC 500 je tak možné sestavit z více než 60 typů různých jednotek.

3.1.3 Komunikace

RS-232

- sériová komunikační linka
- protokol DF1 (Full Duplex)
- rychlost 19,2 kb/s
- pro spojení dvou zařízení
- pro spojení PC – PLC je nutno použít křížený kabel

DH-485

- sériová komunikační linka
- rychlost 19,2 kb/s
- vzdálenost 1250 m
- počet stanic 0-31

4 HARDWARE FIRMY SAUTER

Roku 1910 založil Fritz Sauter v Grindelwaldu (Švýcarsko) malou dílnu, ve které se ručně vyráběly časové spínače a spínací hodiny. Díky spolehlivosti a kvalitě se tyto přístroje dobře ujaly na trhu a tento počáteční úspěch vedl k rozšíření výroby. Krátce nato se výroba časových spínačů a spínacích hodin přemístila do Basileje (Švýcarsko). Tam začal Fritz Sauter vyvíjet svůj první zásobník teplé vody s elektrickým ohřevem. V roce 1920 transformuje svoji firmu na akciovou společnost a za pět let zakládá dceřinou společnost Sauter-Cumulus ve Freiburgu (Německo). Neustále se zvyšující produkce výrobků v oblasti tepelné techniky si vynutila velké rozšíření výroby a vývoj vlastních přesných regulačních přístrojů. Dnes patří firma Fr. Sauter AG se sítí dceřiných společností a zastoupení ve více než 40 státech na pěti kontinentech světa k nejvýznamnějším výrobcům regulačních a řídicích systémů pro oblast topení, chlazení, větrání a klimatizace.

4.1 SAUTER ŘADA EY3600

Nova 220



Obrázek č. 16: Stanice nova220

Stanice nova220 je tou větší z kompaktních jednotek výrobkové řady systému EY3600. Slouží k ovládání a regulaci topení, vzduchotechniky a chlazení i pro všechny ostatní úlohy běžně se vyskytující při řízení provozu budov.

Má celkem 48 vstupů a 18 výstupů. Délka cyklu umožňuje řešit v oblasti řídicí techniky i rychlé úkoly. Lze ji napojit na síť novaNet a zapojit do komunikace bez dalších dodatečných opatření. Programuje se prostřednictvím PC softwarem EY3600-CASE dle IEC 1131-3 (editorem diagramů funkčních modulů FBD).

Stanice obsahuje všechny součástky a rozhraní, která jsou zapotřebí pro provoz, připojení provozních prostředků a komunikaci jak s ostatními automatizačními stanicemi, tak se zobrazovacími přístroji.

Automatizační stanice nova220 disponuje rychlým provozním programem. Tento program přečte každých 150 ms všechny vstupy, zpracuje parametrované moduly, zaktualizuje výstupy a zrealizuje nezbytnou komunikaci s ostatními stanicemi nebo se zobrazovacími PC.

Stanice se programuje (regulační obvody a parametry) prostřednictvím sítě novaNet. Data se pak uloží do bateriově zálohované paměti. Životnost této baterie je minimálně 10 let. Trvale lze tato data zabezpečit v uživatelské paměti EPROM.

Každá automatizační stanice musí mít bezpodmínečně svou adresu. Ta se nastavuje kódovacími přepínači. V síti automatizačních stanic může být napojeno až 28672 účastnických stanic.

K dispozici je ovládací panel nova240 (EYT 240 F001), který se ke stanici připojuje konektorem RJ-45. Tento panel umožňuje zpracovávat všechna data stanice (s výjimkou historické databanky), tzn. načítat měřené hodnoty, poplachy a stavy, měnit žádané hodnoty a zadávat regulační povely.



Obrázek č. 17: Ovládací panel EYT 240

Nova 106



Obrázek č. 18: Nosič karet EYU 108 a EYU 109

Nosiče karet EYU 108 F001, resp. EYU 109 F001 vytvářejí spolu se zásuvnými kartami modulární automatizační stanici, kterou lze osazovat podle požadavků uživatele a potřeb zařízení. Mohou pojmout 5 nebo 11 zásuvných karet, což představuje jednu kartu UPS, jednu procesorovou /napájecí kartu a 3 nebo 9 funkčních karet. Na základní desce nosiče karet jsou umístěny svorky pro připojení napájecího zdroje, provozních prostředků a sítě novaNet.

Karty se na sběrnici připojují pomocí řadových konektorů. Každému zásuvnému místu je přiřazeno 16 svorek. Na tyto svorky se připojují provozní prostředky (beznapěťové kontakty, snímače teploty, motory, světelné zdroje atd.), přičemž k dispozici je 24 nebo 60 hardwarových adres.

Nosič karet představuje základ modulární automatizační stanice. Bezpodmínečně vyžaduje procesorovou/napájecí kartu, která se zasouvá do pozice B. Do pozice A je možné osadit kartu UPS spolu s 12 V baterií. Ta zajišťuje nepřerušitelné napájení všech karet i v případě, že dojde k výpadku dodávky proudu. Do ostatních pozic je možné zasouvat funkční karty libovolně, přičemž

platí, že každé ze zásuvných míst 1–3 nebo 1-7 lze osadit osmifunkčními kartami, každé ze zásuvných míst 8 a 9 dvoufunkčními kartami.

Procesorová a napájecí karta představuje srdce automatizační stanice nova106. Tato karta je centrální procesorovou jednotkou. Realizuje komunikaci jak s ostatními automatizačními stanicemi, tak se zobrazovací a řídicí úrovní. Pro funkční karty zajišťuje všechna potřebná napětí. Délka cyklu umožňuje řešit v oblasti regulační techniky i rychlé úkoly. Lze ji napojit na síť novaNet a zapojit do komunikace bez dalších dodatečných opatření. Programuje se prostřednictvím PC softwarem EY3600-CASE dle IEC 1131-3 (editorem diagramů funkčních modulů FBD). Na kartě se nachází paměť EPROM, zálohovací baterie, bloky přepínačů pro adresování stanic, tři signalizační LED a zásuvka pro přístroj používaný k ručnímu ovládání a pro PC.



Obrázek č. 19: Procesorová karta EYL 106

5 **NORMA IEC 61131**

Mezinárodní norma IEC 61131 se věnuje programovatelným automatům – jejich funkci, provedení hardwaru, komunikací a způsobům jejich programování. Odborné veřejnosti je nejznámější její třetí část, tj. IEC 61131-3, týkající se programování automatů (forma uživatelských programů, deklarace proměnných a typů dat, způsoby aktivace programů a popis programovacích jazyků). Poměrně krátce, od roku 2000, platí také sedmá část normy (IEC 61131-7), věnovaná programování fuzzy řadičů a rozvíjející tak třetí část této normy o jazyk fuzzy řízení.

5.1 **STRUKTURA A HISTORIE IEC 61131**

Norma IEC 61131, po technické stránce zpracovávaná organizací PLCopen a oficiálně vydaná Mezinárodní elektrotechnickou komisí (International Electrotechnical Commission – IEC). Zabývá se programovatelnými automaty, v anglicky psaných textech nazývanými Programmable Logic Controllers a běžně označovanými zkratkou PLC. V současné době se již lze setkat také s výstižnějším označením Programmable Automation Controller a zkratkou PAC – alespoň pro kategorii vyspělých programovatelných automatů. Sama norma IEC 61131 označuje předmět svého zájmu jako Programmable Controllers se zkratkou PC, což může být v souvislosti s hromadným používáním stejné zkratky pro osobní počítače (Personal Computers) poněkud zavádějící. Původně měla norma v označení jen čtyřmístné číslo IEC 1131. Počáteční šestka byla přidána později, a to v souvislosti s přijetím normy IEC 1131 do systému norem Evropské unie.

Norma IEC 61131 má již osm částí. První (vydaná v roce 1992) obsahuje základní informace z oboru PLC, část 2 (1993) shrnuje požadavky na provedení a testování hardwaru PLC, část 3 (1993) je věnována programování PLC a příslušným programovacím jazykům, část 4 (1995) sjednocuje požadavky na uživatelské příručky, část 5 (v tisku) na komunikace a část 6 (ve studijní fázi) na komunikace programovatelných automatů prostřednictvím průmyslových komunikačních sběrnic. Část 8 (2000 a dále upravovaná) obsahuje směrnice pro používání a implementaci programovacích jazyků.

5.2 NORMA IEC EN 61131-3

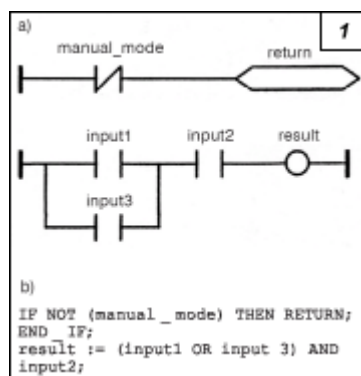
V rámci normy IEC 61131-3 jsou doporučovány čtyři programovací jazyky s přesně definovanou sémantikou a syntaxí: LD, FBD, IL a ST. Jako pátý programovací jazyk se často uvádí sekvenční funkční diagram – SFC, který však není v normě zařazen mezi jazyky, ale mezi tzv. společnými prvky, neboť tvoří jakousi nadstavbu pro strukturování celé aplikace. Dále existuje např. nabídka tzv. inženýrských nástrojů STEP7, kam patří S7-Graph, S7-HiGraph a CFC. To jsou však spíše grafické nástroje pro programování, nikoli jazyky s přesně definovanou syntaxí a sémantikou.

Kodifikované označení v angličtině		Označení obvyklé v němčině		Vhodný název v češtině
zkratka	název	zkratka	název	
LD	<i>Ladder Diagram</i>	KOP	<i>Kontaktplan</i>	reléové schéma
FBD	<i>Function Block Diagram</i>	FUP	<i>Funktionplan</i>	jazyk funkčních bloků
IL	<i>Instruction List</i>	AWL	<i>Anweisungsliste</i>	jazyk mnemokódů
ST	<i>Structured Text</i>	ST	<i>Strukturierter Text</i>	strukturovaný text
SFC	<i>Sequential Function Chart</i>	AS	<i>Ablaufsprache</i>	jazyk sekvenčního programování

Obrázek č. 20: Souhrnný přehled programovacích jazyků

5.2.1 Jazyk příčkového diagramu LD

Grafický jazyk LD (Ladder Diagram) je někdy také nazýván jazykem kontaktních schémat a je založen na grafické reprezentaci reléové logiky. Organizační jednotka programu je vyjádřena sítí propojených grafických prvků. Sít' v jazyku LD je zleva i zprava ohraničena svislými čarami, které se nazývají levá a pravá napájecí sběrnice. Mezi nimi je tzv. příčka, která může být rozvětvena. Každý úsek příčky, vodorovný nebo svislý, může být ve stavu on nebo off. Do příček mohou být včleněny kontakty (spínací, rozpínací apod.), cívky (cívka, negovaná cívka apod.) a dále funkce a funkční bloky.

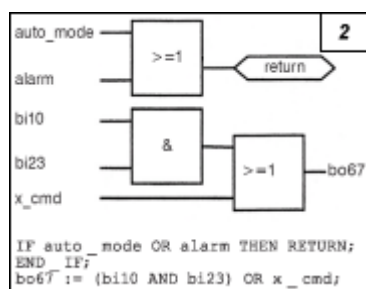


Obrázek č. 21: Příklad zápisu programu a) zápis v jazyce LD, b) ekvivalentní zápis v ST

5.2.2 Jazyk funkčního blokového schématu FBD

Druhým grafickým jazykem je FBD (Function Block Diagram), který vyjadřuje chování funkcí, funkčních bloků a programů jako soubor vzájemně provázaných grafických bloků podobně jako v elektronických obvodových diagramech. Jde o systém prvků, které zpracovávají signály. Často se zde používají standardní funkční bloky, jako jsou např. bistabilní prvky (paměti s dominantním vypnutím nebo sepnutím, semafor), prvky pro detekci náběžné a sestupné hrany,

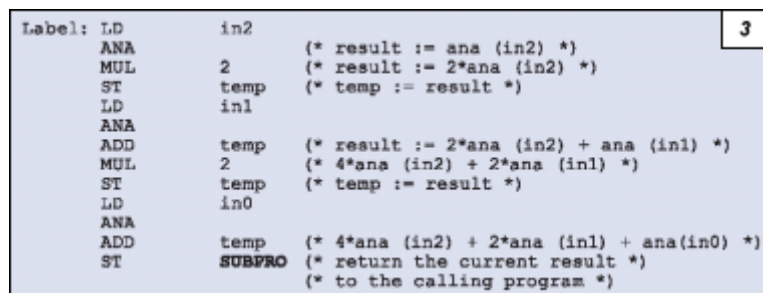
čítače, časovače a komunikační bloky definované v normě IEC 1131-5. Podle potřeby jsou doplňovány speciální bloky a každá firma nabízí ve svém programovacím prostředí poněkud odlišný soubor bloků (např. spínací hodiny týdenní, roční, generátory impulzů, komparátory apod.).



Obrázek č. 22: Příklad zápisu programu a) zápis v jazyce FBD, b) ekvivalentní zápis v ST

5.2.3 Jazyk seznamu instrukcí IL

Textový jazyk IL (Instruction List) označovaný také jako jazyk pokynů (povelů), seznam instrukcí (STL – Statement List) poněkud připomíná assembler. Programová organizační jednotka je složena ze sekvence instrukcí, z nichž každá začíná na novém řádku a může obsahovat návěští (nepovinné) ukončené dvojtečkou, operátor (např. AND, &, ADD, CAL apod.), který může být případně doplněn tzv. modifikátorem, operand a někdy také komentář (nepovinný). Pomocí modifikátorů se vyjadřují negace, podmíněnost a nepodmíněnost instrukce skoků, volání a návratů a priorit.



Obrázek č. 23: Příklad zápisu programu v jazyce IL

5.2.4 Jazyk strukturovaného textu ST

Textový jazyk ST (Structured Text) je výkonný vyšší programovací jazyk, který má kořeny v jazycích Pascal a C. Syntaxe jazyka je dána povolenými výrazy a příkazy. Vyhodnocením výrazu vyjde hodnota v některém z definovaných datových typů. Výraz se skládá z operátorů a operandů. Operandem může být konstanta, proměnná, funkce nebo jiný výraz. Operátory pro jazyk ST jsou definovány pro sedmnáct typů operací (vyhodnocení funkce, negace, násobení, booleovské funkce AND, XOR a OR apod.). Je definováno deset typů příkazů (přiřazení, vyvolání funkce, návrat, výběr apod.). Příkazy jsou odděleny středníkem a může jich být více na jednom řádku. Jazyk ST je vhodným nástrojem pro definování komplexních funkčních bloků, které pak mohou být použity v libovolném programovacím jazyku.

```

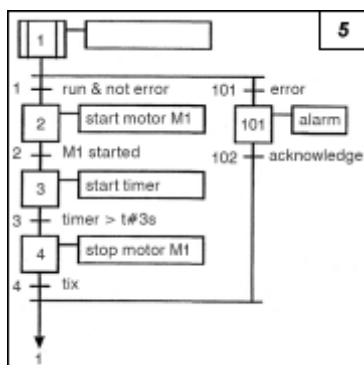
FUNCTION_BLOCK fbStartStop
VAR_INPUT
    start : BOOL R_EDGE ;
    stop : BOOL R_EDGE ;
END_VAR
VAR
    VAR
END_VAR
VAR_OUTPUT
    vystup : BOOL ;
END_VAR
Vystup := (vystup OR start) and
not stop ;
END_FUNCTION_BLOCK

```

Obrázek č. 24: Příklad zápisu programu v jazyce ST

5.2.5 Sekvenční funkční diagram SFC

SFC (Sequential Function Chart) popisuje sekvenční chování řídicího programu. Je odvozen ze symboliky Petriho sítí, ale liší se od nich tím, že grafická reprezentace se zde převádí přímo do souboru výkonných řídicích prvků. SFC strukturalizuje vnitřní organizaci programu a umožňuje rozložit úlohu řízení na zvládnutelné části a zachovat přitom přehled o chování celku. Sekvenční funkční diagram se skládá z kroků a přechodů. Každý krok reprezentuje stav řízeného systému a má k sobě přiřazen blok akcí. Přechod je spojen s podmínkami, které musí být splněny, aby mohl být deaktivován krok, který přechodu předchází a naopak aktivován krok, který následuje. Každý prvek, tzn. přechod i blok akcí, může být naprogramován v libovolném jazyce definovaném v normě, včetně vlastního SFC. K základním strukturám SFC patří lineární sekvence, alternativní větvení se spojením alternativních větví a paralelní souběh více větví s jejich následnou synchronizací.



Obrázek č. 25: Příklad zápisu programu v jazyce SFC

6 VÝVOJOVÉ PROSTŘEDÍ STEP 7 – MICRO/WIN

Vývojovým prostředím pro automaty SIMATIC S7-200 je software STEP 7 – Micro/WIN. Tento software byl navržen tak, aby i začátečník, který má alespoň základní znalosti práce s PC, dokázal během několika hodin systematické práce automat naprogramovat. Obrovskou výhodou jsou tzv. průvodci, kteří automatizují rutinní a složitější úlohy, jako např. automatické nastavení PID regulátoru, komunikace a polohování.

Programování probíhá metodou drag & drop. Uživatel má výběr mezi třemi režimy editace: LAD (kontaktní schéma), FBD (funkční bloky) a STL (výpis instrukcí). Mezi těmito režimy je možné kdykoliv přepínat. Software má rozsáhlé možnosti ladění programu, on-line sledování stavu, grafické zobrazení průběhu procesních dat, přímé vnucení hodnot apod. Nezanedbatelná je i možnost změny uživatelského programu za běhu, bez nutnosti zastavení automatu.

Pro zjednodušení programování je možné vytvořit vlastní knihovny, které obsahují často používané podprogramy. Takto se ušetří spousta času při programování. Knihovny je možné opatřit heslem, aby byla zajištěna ochrana Vašeho know-how.

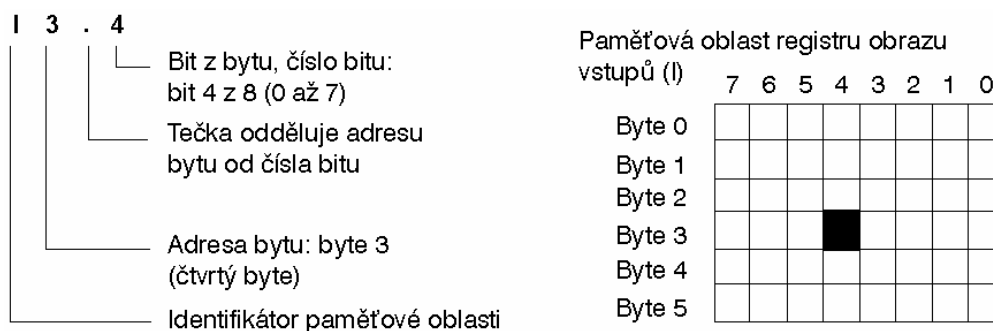
6.1 PŘÍSTUP K DATŮM

Programovatelný automat S7-200 uchovává informace v různých paměťových oblastech, které mají jedinečné adresy. Je možné explicitně určit konkrétní adresu paměti. To programu umožní přímý přístup k informacím. Tabulka ukazuje rozsahy, které mohou být reprezentovány daty různých velikostí.

Reprezentace	Byte (B)	Word (W)	Double word (D)
Unsigned integer	0 až 255 0 až FF	0 až 65 535 0 až FFFF	0 až 4 294 967 295 0 až FFFF FFFF
Signed integer	-128 až +127 80 až 7F	-32 768 až +32 767 8000 až 7FFF	-2 147 483 648 až +2 147 483 647 8000 0000 až 7FFF FFFF
Real IEEE 32bitová pohyblivá řádová čárka	Nežve	Nežve	+1.175495E-38 až +3.402823E+38 (kladné) -1.175495E-38 až -3.402823E+38 (záporné)

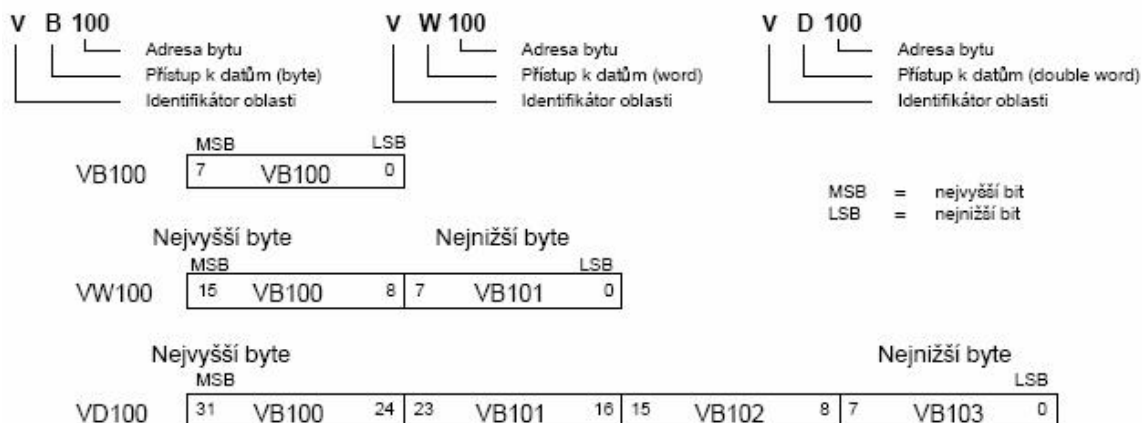
Obrázek č. 26: Desítkové a šestnáctkové rozsahy pro různé velikosti dat

Při přístupu k nějakému bitu v paměťové oblasti je nutné specifikovat adresu, která se skládá z identifikátoru paměťové oblasti, adresy bytu a čísla bitu. Následující obrázek ukazuje příklad přístupu k bitu (často se nazývá adresování "byte.bit"). V tomto příkladu za paměťovou oblastí a adresou bytu (I = vstup a 3 = byte 3) následuje tečka ("."), která odděluje bitovou adresu (bit 4).



Obrázek č. 27: Adresování byte.bit

K datům ve většině paměťových oblastí (V, I, Q, M, S, L a SM) typu byte, word nebo double word se může přistupovat v tzv. bytovém adresním formátu. Pro přístup k bytu, word nebo double word v paměti je nutné vytvořit adresu podobně, jako byla vytvořena adresa pro přístup k bitu. Adresa se skládá z identifikátoru oblasti, z označení velikosti dat a z adresy počátečního bytu hodnoty bytu, word nebo double word.



Obrázek č. 28: Porovnání přístupu do stejné oblasti k byte, word a double word

6.2 PŘÍSTUP K DATŮM V PAMĚŤOVÝCH OBLASTECH

Registr obrazu vstupů: I

Automat S7-200 na začátku každého programového cyklu načte aktuální hodnoty vstupních bodů a zapíše je do registru obrazu vstupů. Přístupovat k registru obrazu vstupů je možné v bitech, bytech, word nebo double word:

Bit: I [adresa bytu].[adresa bitu] I0.1
Byte, word, double word: I [velikost] [adresa počátečního bytu] IB4

Registr obrazu výstupů: Q

Na konci každého programového cyklu zapisuje S7-200 hodnoty uložené v registru obrazu výstupů na výstupy. K registru obrazu výstupů je možné přistupovat v bitech, bytech, word nebo double word:

Bit: Q [adresa bytu].[adresa bitu] Q1.1
Byte, word, double word: Q [velikost] [adresa počátečního bytu] QB5

Oblast proměnné paměti: V

Paměť V můžete používat pro uložení mezivýsledků operací prováděných řídicí logikou programu. Je možné ji také používat pro ukládání ostatních dat souvisejících s procesem nebo úlohou. Do paměťové oblasti V je možné přistupovat v bitech, bytech, word nebo double word:

Bit: V [adresa bytu].[adresa bitu] V10.2
Byte, word, double word: V [velikost] [adresa počátečního bytu] VW100

Oblast bitové paměti: M

Oblast bitové paměti (paměť M) se využívá na řídicí kontakty pro uložení mezistavu procesu nebo mezistavu jiných řídicích informací. Do bitové paměťové oblasti můžete vstupovat v bitech, bytech, word nebo double word:

Bit: M [adresa bytu].[adresa bitu] M26.7
Byte, word, double word: M [velikost] [adresa počátečního bytu] MD20

Speciální paměť: SM

SM bity slouží pro výměnu informací mezi CPU a programem. Tyto bity se používají pro nastavení a řízení některých speciálních funkcí CPU S7-200, jako jsou: bit prvního programového cyklu; bit, který se mění s pevnou frekvencí; nebo bit stavu matematických nebo

provozních instrukcí. K datům v této paměti můžete přistupovat po bitech, bytech, word nebo double word:

Bit:	SM [adresa bytu].[adresa bitu]	SM0.1
Byte, word, double word:	SM [velikost] [adresa počátečního bytu]	SMB86

Oblast lokální paměti: L

Programovatelný automat S7-200 má 64 bytů lokální paměti, z nichž 60 bytů může být použito jako zápisníková paměť nebo pro předávání formálních parametrů podprogramům.

Lokální paměť je podobná paměti V s jedinou výjimkou. Paměť V je globální, zatímco paměť L je lokální. Globální paměť je taková paměť, jejíž data je možné číst a zapisovat z libovolné části programu (hlavního programu, podprogramů nebo přerušení). Lokální paměť je taková paměť, která je určena pouze pro nějakou část programu. S7-200 přiděluje 64 bytů paměti L pro hlavní program, 64 bytů pro každou úroveň vnoření podprogramu a 64 bytů pro přerušení.

Paměť určená pro hlavní program se nedá adresovat (není možné číst nebo zapisovat data) z podprogramů nebo z přerušení. Podprogram nemůže adresovat část paměti L přidělenou hlavnímu programu, přerušení nebo jinému podprogramu. Podobně nemůže přerušení adresovat paměť L hlavního programu nebo jiného podprogramu.

Programovatelný automat S7-200 přiděluje paměť L podle potřeby. To znamená, že během provádění hlavního programu neexistuje paměť L pro podprogramy a pro přerušení. Až v okamžiku, kdy je vyvolán podprogram nebo přerušení, přidělí se pro něj lokální paměť podle potřeby. Nové přidělení paměti L může opět použít stejná paměťová místa jiného podprogramu nebo přerušení.

Při přidělení paměti L nevymaže automat její paměťová místa; tato místa mohou obsahovat libovolné hodnoty. Při odevzdávání formálních parametrů do programu vloží S7-200 parametry na odpovídající místo v paměti L volaného podprogramu. Místa paměti L, která neobdrží žádnou hodnotu kvůli tomu, že je vypuštěn krok formálního parametru, se neresetují a mohou v době přidělení obsahovat libovolnou hodnotu.

Bit:	L [adresa bytu].[adresa bitu]	L0.0
Byte, word, double word:	L [velikost] [adresa počátečního bytu]	LB33

Analogové vstupy: AI

Automat S7-200 převádí analogové hodnoty (jako je teplota nebo napětí) na digitální hodnoty o délce word (16 bitů). Tyto hodnoty se dají získat pomocí identifikátoru oblasti (AI), velikosti dat (W) a adresy počátečního bytu. Protože analogové vstupy mají velikost word a vždy začínají na adrese bytu se sudým číslem (jako je 0, 2 nebo 4), přistupuje se k nim vždy se sudou adresou bytu (např. AIWO, AIW2 nebo AIW4). Analogové vstupní hodnoty se dají pouze číst.

Formát:	AIW [adresa počátečního bytu]	AIW4
---------	-------------------------------	------

Analogové výstupy: AQ

Automat S7-200 převádí 16bitovou digitální hodnotu velikosti word na proud nebo na napětí úměrné digitální hodnotě. Tyto hodnoty se dají zapsat pomocí identifikátoru oblasti (AQ), velikosti dat (W) a adresy počátečního bytu. Protože analogové výstupy mají velikost word a vždy začínají na adrese bytu se sudým číslem (jako je 0, 2 nebo 4), zapisují se vždy se sudou adresou bytu (například AIWO, AIW2 nebo AIW4). Hodnoty analogových výstupů se dají pouze zapisovat.

Formát:	AQW[adresa počátečního bytu]	AQW4
---------	------------------------------	------

6.3 ZÁKLADNÍ PRVKY PROGRAMU

Hlavní program

Hlavní program obsahuje instrukce, které řídí aplikaci. S7-200 vykonává tyto instrukce postupně za sebou v časovém sledu jednou za programový cyklus. Hlavní program se také označuje jako OB1.

Podprogramy

Tyto volitelné prvky uživatelského programu se provádějí pouze tehdy, jsou-li volány: hlavním programem, podprogramem přerušení nebo jiným podprogramem. Podprogramy jsou užitečné v případě, kdy chcete nějakou funkci provádět opakovaně. Pohodlnější než přepisovat logiku pro každé místo v programu, kde chcete, aby se daná funkce vyskytla, je zapsat logiku jednou do podprogramu a volat podprogram tolikrát, kolikrát je to během hlavního programu potřeba.

Podprogramy mají několik výhod:

- použití podprogramů zmenšuje celkovou velikost uživatelského programu
- použití podprogramu zkracuje dobu trvání programového cyklu, protože se tak přesune kód mimo hlavní program
- S7-200 vyhodnocuje kód v hlavním programu v každém programovém cyklu, ať se kód provádí, nebo ne
- v podprogramu ale vyhodnocuje S7-200 kód pouze tehdy, je-li tento podprogram volán; v programových cyklech, ve kterých volán není, ho nevyhodnocuje
- použití podprogramů vytváří kód, který je přenositelný
- kód funkce je možné uložit jako podprogram a ten pak zkopírovat do jiných programů pouze s minimálními úpravami

Podprogramy přerušení

Tyto volitelné prvky uživatelského programu reagují na konkrétní události přerušení. Program se navrhuje tak, aby ošetřil předdefinovanou událost přerušení. Kdykoliv ke specifikované události dojde, S7-200 provede podprogram přerušení.

Podprogramy přerušení nejsou volány hlavním uživatelským programem. Pouze se přiřadí k definované události podprogram přerušení a S7-200 provede instrukce podprogramu pouze při výskytu této události.

Ostatní prvky programu

Ostatní bloky obsahují informace pro S7-200. Download těchto bloků se může provést spolu s downloadem uživatelského programu.

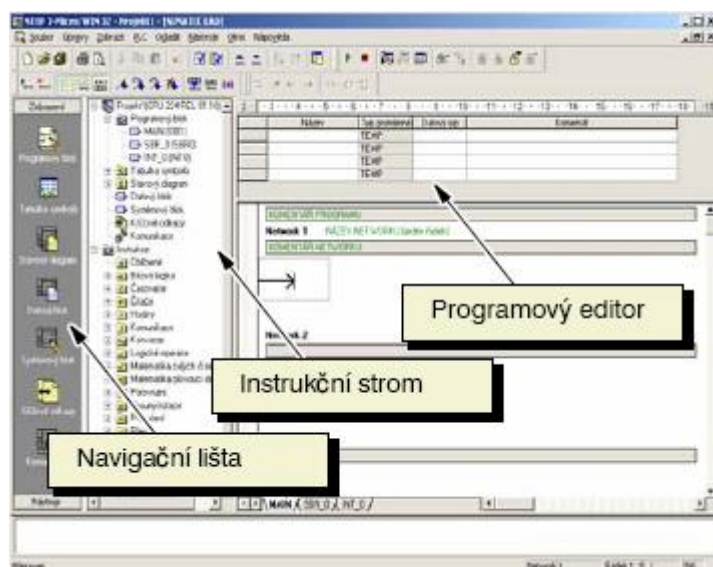
Systémový blok

- systémový blok umožňuje konfigurovat různé volitelné části hardwaru pro S7-200

Datový blok

- datový blok uchovává hodnoty proměnných (paměť V) používaných uživatelským programem
- datový blok se může použít pro vložení počátečních hodnot dat

6.4 POPIS VÝVOJOVÉHO PROSTŘEDÍ



Obrázek č. 29: Vývojové prostředí STEP 7 – micro/WIN

Jak je vidět na obrázku, okno projektu STEP 7-Micro/WIN nabízí přehledné pracovní prostředí pro vytváření řídicího programu.

Nástrojové lišty obsahují tlačítka pro zkratky k často používaným příkazům menu. Všechny nástrojové lišty se mohou zobrazit nebo skrýt.

Navigační lišta nabízí ikony pro přístup k různým programovacím prvkům STEP 7-Micro/WIN.

Strom s instrukcemi zobrazuje všechny objekty projektu a instrukce pro tvorbu řídicího programu. Jednotlivé instrukce se mohou do programu vložit metodou drag&drop nebo stačí na instrukci dvakrát kliknout, čímž se vloží na současnou pozici kurzoru v programovém editoru.

Programový editor obsahuje program a tabulku lokálních proměnných, ve které se mohou přiřadit symbolické názvy dočasným lokálním proměnným. Podprogramy a přerušení jsou zobrazeny jako záložky ve spodní části okna programového editoru. Přecházení mezi podprogramy, přerušeními a hlavním programem se provede kliknutím na příslušnou záložku.

STEP 7-Micro/WIN obsahuje tři editory pro vytváření uživatelského programu:

- kontaktní schémata (LAD)
- výpis příkazů (STL)
- funkční bloky (FBD)

S určitými omezeními mohou být programy psané v kterémkoliv z těchto programových editorů, prohlíženy a editovány ostatními programovými editory.

Vlastnosti editoru STL

Editor STL zobrazuje program jako znakově orientovaný programovací jazyk. Umožňuje vytvářet řídicí programy vkládáním textových instrukcí. Editor STL také umožňuje tvorbu programů, které by pomocí editoru LAD nebo FBD nešly vytvořit. Je to proto, že v STL se programuje v jazyce S7-200 a nikoli v jazyce grafického editoru, kde platí určitá omezení, aby byly diagramy správně nakresleny. Jak je vidět na obrázku č.30, je tato znakově orientovaná koncepce velmi podobná programování ve strojovém kódu.

LD	I0.0	//Načtení jednoho vstupu
A	I0.1	//Logický součin s jiným vstupem
=	Q1.0	//Zapsání hodnoty na výstup 1

Obrázek č. 30: Příklad programu STL

S7-200 provádí každou instrukci v pořadí určeném programem shora dolů, pak začne opět odshora. Pro řešení řídicí logiky používá STL logický zásobník. Vy vkládáte instrukce STL pro manipulaci se zásobníkem.

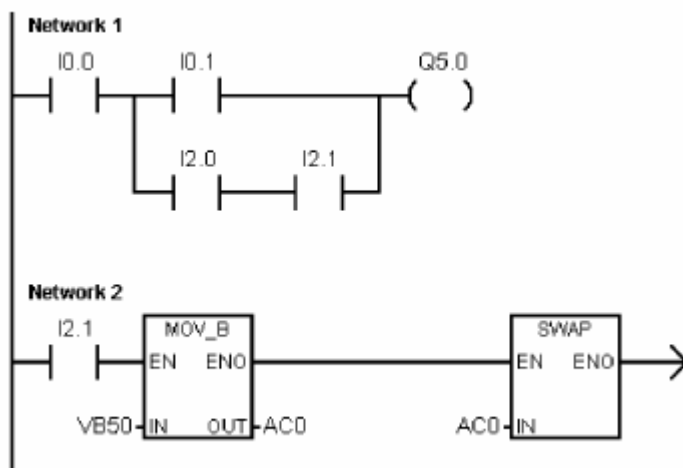
Jestliže se zvolí editor STL, musí se brát v úvahu tyto hlavní body:

- STL je nejvhodnější pro zkušené programátory
- STL někdy umožní řešit problémy, které se nedají snadno řešit pomocí editorů LAD a FBD
- editor STL můžete používat pouze s instrukčním souborem SIMATIC
- editor STL můžete vždy použít na prohlížení nebo editaci programu, který byl vytvořen pomocí editorů LAD nebo FBD; naopak to, ale není vždy možné
- nemůžete vždy použít editory LAD nebo FBD pro zobrazení programu, který byl napsán v editoru STL

Vlastnosti editoru LAD

Editor LAD zobrazuje program v grafické formě podobné schématům. Programy v kontaktním schématu umožňují simulovat tok elektrického proudu z napájecího zdroje přes řadu logických vstupních podmínek, které následně aktivují výstupní logické podmínky. Program LAD obsahuje levou napájecí lištu, která je pod napětím. Kontakty, které jsou sepnuté, umožňují tok energie do dalšího prvku; kontakty, které jsou rozepnuté, tok energie blokují.

Logika je dělena do spojitých sítí (network). CPU provádí vždy jeden network zleva doprava a pak shora dolů tak, jak je to určeno programem. Obrázek č.31 ukazuje příklad programu LAD.



Obrázek č. 31: Příklad programu LAD

Instrukce jsou představovány grafickými symboly a mají tři základní formy.

Kontakty představují logické vstupní podmínky, jako jsou spínače, tlačítka nebo vnitřní podmínky.

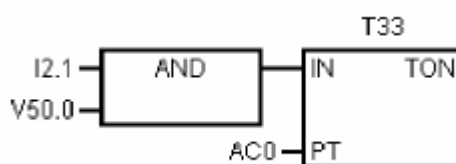
Cívky obvykle představují logické výstupní obvody, jako jsou žárovky, startéry motoru, přechodová relé nebo vnitřní výstupní podmínky.

Bloky představují doplňující instrukce, jako jsou časovače, čítače nebo matematické instrukce. Jestliže se zvolí editor LAD, musí se vzít v úvahu tyto hlavní body:

- kontaktní schémata snadno používají i programátoři začátečníci
- grafické zobrazení se snadno chápe a je populární na celém světě
- editor LAD je možné použít s instrukčním souborem SIMATIC i IEC 1131-3
- vždy se může použít editor STL pro zobrazení programu vytvořeného editorem SIMATIC LAD

Vlastnosti editoru FBD

Editor FBD zobrazuje program v grafické formě, která připomíná běžná logická schémata. Neobsahuje kontakty ani cívky, které se nalézají v editoru LAD, ale ekvivalentní instrukce, které se objevují jako blokové instrukce.



Obrázek č. 32: Příklad programu FBD

FBD nepoužívá pojem levých a pravých napájecích lišt, proto se termín "signálový tok" používá pro vyjádření analogického pojmu toku řízení přes logické bloky FBD.

Cesta logické "1" přes prvky FBD se nazývá signálový tok. Vstup a výstup signálového toku je možné přímo přiřadit operandu.

Logika programu je odvozena od spojení těchto bloků. To znamená, že výstup jedné instrukce (jako je například blok AND) je možné použít pro aktivaci jiné instrukce (např. časovače). Tato koncepce spojování umožní vyřešit mnoho různých logických problémů.

Jestliže se zvolí pro programování editor FBD, musí se brát v úvahu tyto hlavní body:

- styl zobrazení pomocí grafických logických členů umožňuje dobře sledovat tok programu
- editor FBD je možné použít s instrukčním souborem SIMATIC i IEC 1131-3
- pro zobrazení programu vytvořeného editorem SIMATIC FBD můžete vždy použít editor STL

Tabulka symbolů

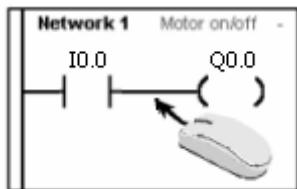
Tabulka symbolů umožňuje definovat a editovat symboly, ke kterým je odkudkoliv z programu možné přistupovat pomocí symbolických názvů. Tabulka symbolů se také označuje jako tabulka globálních proměnných. Operandy instrukcí uživatelského programu se mohou označit absolutně nebo symbolicky. Absolutní odkaz používá pro označení adresy paměťový bit nebo byte. Symbolický odkaz používá pro označení adresy kombinací alfanumerických znaků.

Tabulka symbolů																
	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
			Název			Adresa			Komentář							
1			USS_LOW_V													
2			USS_HIGH_V													
3			USS_TIME_OUT													
4			USS_I_LINE													
5			USS_DAT_PTRD													
6			USS_ACT_DRVD													
7			USS_CUR_DRVD													
8			USS_F0													
9			USS_FREQ_CHGP													

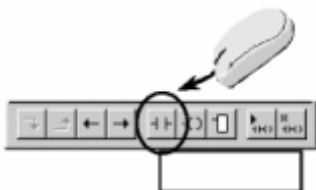
Obrázek č. 33: Tabulka symbolů

6.5 PŘÍKLAD VLOŽENÍ PŘÍKAZU V EDITORU LAD

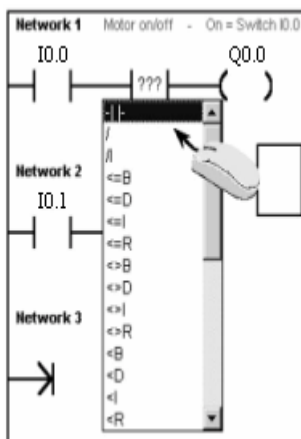
Vložení podmínky v klidu zavřeného kontaktu M0.0 mezi podmínku sepnutí vstupu I0.0 a povelu pro výstup Q0.0



1. Pro vložení podmínky je nutné označit místo v příslušném Networku, kam chceme vložit příkaz kliknutím levého tlačítka myši.



2. Pro zobrazení instrukcí bitové logiky je nutné dvakrát kliknout na ikonu Bitová logika nebo na znaménko plus (+) .



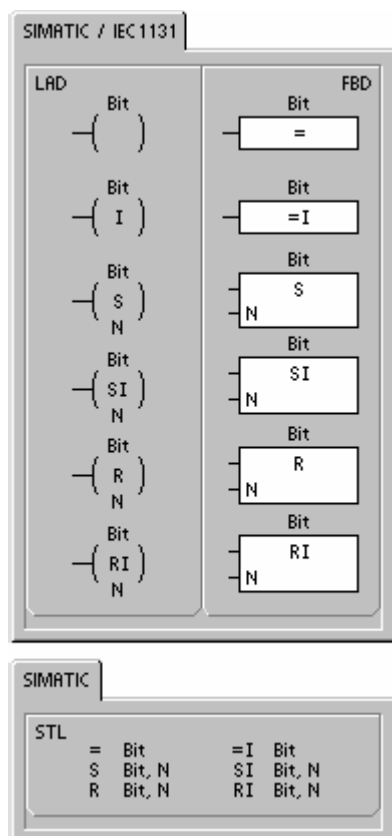
3. Zvolit v klidu zavřený kontakt.

4. Držet stisknuté levé tlačítko myši a přesunout kontakt na první network.

5. Kliknout na “???” nad kontaktem a zadat tuto adresu: M0.0

6. Stisknout Enter pro uložení adresy kontaktu.

6.6.2 Instrukce bitové logiky – cívky



Výstup

Instrukce “Výstup” (=) zapíše novou hodnotu výstupního bitu do registru obrazu procesu. Když je provedena instrukce výstupu, S7-200 nastaví nebo vynuluje výstupní bit registru obrazu procesu. Pro LAD a FBD je tento bit nastaven tak, že se rovná signálovému toku. V STL je do specifikovaného bitu zkopírována horní hodnota zásobníku.

Okamžitý výstup

Instrukce “Okamžitý výstup” (=I) okamžitě zapíše při svém provádění novou hodnotu na fyzický výstup a také na odpovídající místo registru obrazu procesu. Při provádění instrukce okamžitého výstupu je fyzický výstup (bit) okamžitě nastaven tak, že se rovná signálovému toku. Instrukce okamžitě zkopíruje horní hodnotu zásobníku do specifikovaného bitu fyzického výstupu (STL). “I” označuje okamžitou instrukci; při provádění instrukce je nová hodnota zapsána na fyzický výstup i na odpovídající místo registru obrazu procesu. Tím se liší od nepřímých instrukcí, které zapisují novou hodnotu pouze do registru obrazu procesu.

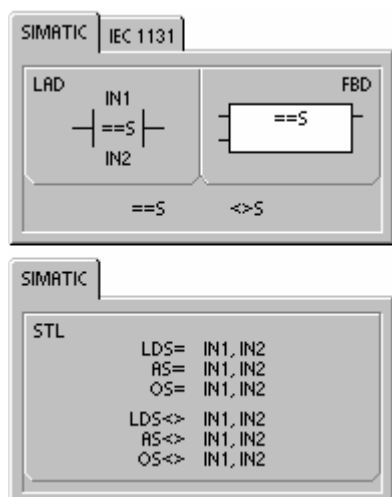
Set (nastavení) a Reset (nulování)

Instrukce Set (S) a Reset (R) nastaví nebo vynuluje specifikovaný počet výstupů (N), přičemž začíná na specifikované adrese (bitu). Může se nastavovat nebo vynulovat 1 až 255 výstupních bodů.

Set okamžitě a Reset okamžitě

Instrukce okamžitě nastaví nebo vynuluje specifikovaný počet výstupů (N), přičemž začíná na specifikované adrese (bitu). Může se okamžitě nastavovat nebo vynulovat 1 až 128 bodů

6.6.3 Instrukce porovnání řetězců

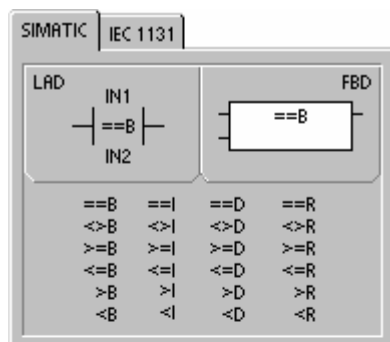


Instrukce “Porovnání řetězců” porovnává dva řetězce znaků v ASCII kódu:

IN1 = IN2 IN1 <> IN2

Když je výsledek porovnání pravdivý, instrukce porovnání zapne kontakt (LAD) nebo výstup (FBD) nebo provede instrukce Načtení, AND nebo OR s jedničkou a vrcholem zásobníku (STL).

6.6.4 Instrukce porovnání číselných hodnot



Instrukce porovnání se používají pro porovnání dvou hodnot:
 IN1 = IN2 IN1 >= IN2 IN1 <= IN2
 IN1 > IN2 IN1 < IN2 IN1 <> IN2

Operace porovnání bytu jsou bez znaménka.

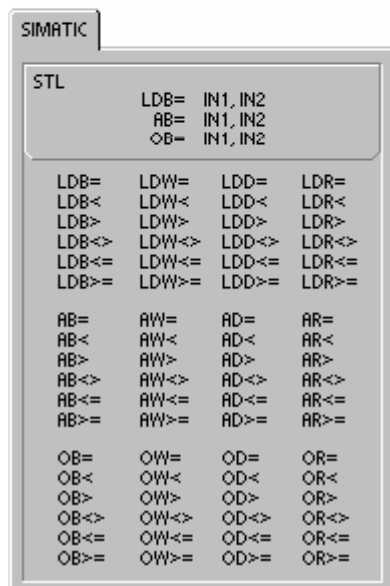
Operace porovnání celých čísel (integer) jsou se znaménkem.

Operace porovnání double word jsou se znaménkem.

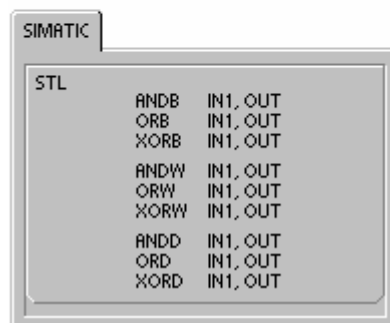
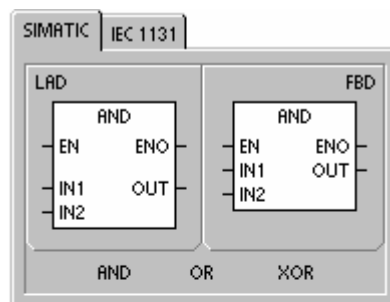
Operace porovnání reálných čísel jsou se znaménkem.

V LAD a FBD: Je-li výsledek porovnání pravdivý, instrukce nastaví kontakt (LAD) nebo výstup (FBD) na hodnotu log. 1.

V STL: Je-li výsledek porovnání pravdivý, provedou se instrukce Načtení, AND nebo OR s jedničkou a vrcholem zásobníku (STL).



6.6.5 Instrukce logických operací – AND, OR, XOR



AND byte, AND word a AND double word

Instrukce "AND byte" (ANDB), "AND word" (ANDW) a "AND double word" (ANDD) provedou logický součin odpovídajících bitů dvou vstupních hodnot IN1 a IN2 a výsledek uloží do paměťového místa OUT.

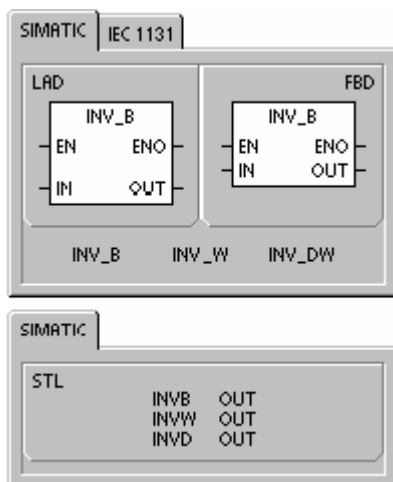
OR byte, OR word a OR double word

Instrukce "OR byte" (ORB), "OR word" (ORW) a "OR double word" (ORD) provedou logický součet odpovídajících bitů dvou vstupních hodnot IN1 a IN2 a výsledek uloží do paměťového místa OUT.

XOR byte, XOR word a XOR double word

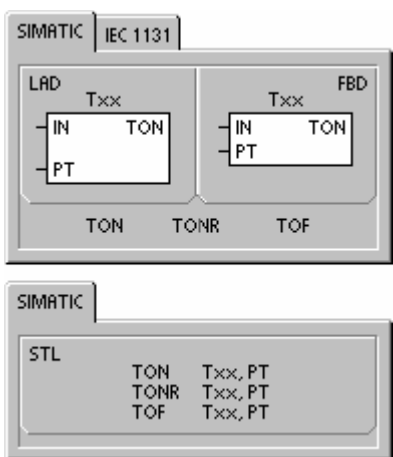
Instrukce "XOR byte" (XORB), "XOR word" (XORW) a "XOR double word" (XORD) provedou nonekvivalenci odpovídajících bitů dvou vstupních hodnot IN1 a IN2 a výsledek uloží do paměťového místa OUT.

6.6.6 Instrukce logických operací – Inverze



Inverze byte, word a double word Instrukce "Invertovat byte" (INVB), "Invertovat word" (INVW) a "Invertovat double word" (INVD) vytvoří jedničkový doplněk vstupu IN a výsledek uloží do místa paměti OUT.

6.6.7 Instrukce časovače



Časovač zapnutí

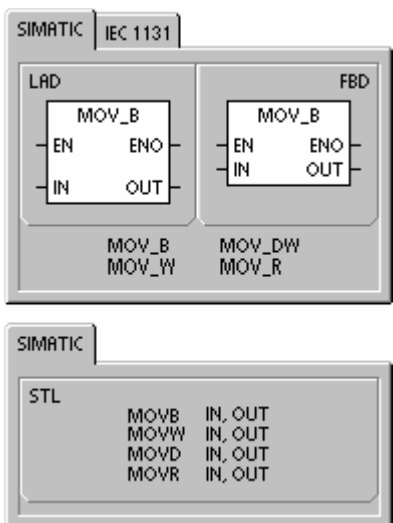
Remanentní časovač zapnutí

Instrukce "Časovač zapnutí" (TON) a "Remanentní časovač zapnutí" (TONR) odpočítávají čas od sepnutí vstupu. Číslo (Txx) rozlišuje jednotlivé časovače.

Časovač vypnutí

Instrukce "Časovač vypnutí" (TOF) se používá na zpoždění rozepnutí výstupu o pevný čas po rozepnutí vstupu.

6.6.8 Instrukce přesunu



Přesun byte, word, double word nebo real

Instrukce "Přesunout byte" (MOVB), "Přesunout word" (MOVW), "Přesunout double word" (MOVD) a "Přesunout real" (MOVR) přesunou hodnotu z paměťového místa IN na nové paměťové místo OUT, aniž by změnil původní hodnotu.

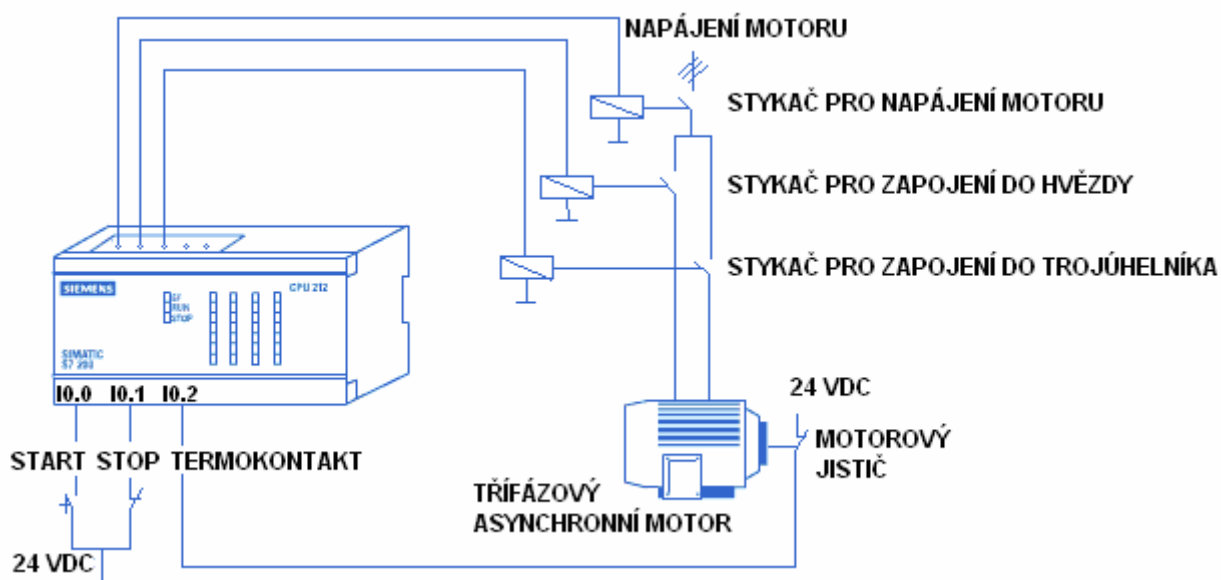
7 VYTVOŘENÍ VZOROVÉ APLIKACE

7.1 ZADÁNÍ APLIKACE

Tento vzorový program ovládá spuštění třífázového asynchronního motoru způsobem hvězda-trojúhelník.

Při zjištění sepnutí digitálního vstupního signálu I0.0 stisknutím tlačítka START dojde ke spuštění motoru do hvězdy. Po uplynutí 5 vteřin přejde motor z hvězdy do trojúhelníku.

Motor je vypnut v případě rozepnutí digitálního vstupního signálu I0.1 stisknutím tlačítka STOP pro zastavení motoru. Nebo pokud dojde k zareagování termokontaktu, který je připojen na digitální vstup I0.2.



Obrázek č. 34: Zobrazení technického řešení

7.2 ŘEŠENÍ APLIKACE

1. krok

Pokud je stisknuto tlačítko START a není stisknuto zároveň tlačítko STOP, dojde k nastavení požadavku na spuštění motoru.

2. krok

Pokud je stisknuto tlačítko STOP, dojde k zastavení motoru. Tlačítko STOP je připojeno jako rozpínací kontakt. To znamená, že dojde k rozpojení signálu do PLC pouze při stisknutí tlačítka nebo pokud například došlo k přerušení napájení tlačítka. Tato varianta je použita z důvodu, pokud by došlo k závadě tlačítka STOP, tak aby došlo k vypnutí motoru.

3. krok

Pokud je nastaven požadavek na spuštění motoru a termokontakt je bez poruchy, dojde k povolení startu motoru. Termokontakt je opět zapojen na rozpínací kontakt ze stejného důvodu jako tlačítko STOP.

4. krok

Pokud je povolen start motoru, je sepnut stykač pro napájení motoru.

5. krok

Ve chvíli, kdy je sepnut stykač pro napájení motoru je spuštěn časovač, který časuje po dobu 5 vteřin.

6. krok

Pokud je sepnut stykač pro napájení motoru a časovač stále časuje (tzn. doba časování $< 5s$), je sepnut stykač pro sepnutí motoru do hvězdy.

7. krok

Pokud je sepnut stykač pro napájení motoru a časovač dočasoval (tzn. doba časování $= 5s$) a není spuštěn stykač pro zapojení do hvězdy, je sepnut stykač pro sepnutí motoru do trojúhelníku.

7.3 ZPRACOVÁNÍ VE VÝVOJOVÉM PROSTŘEDÍ MICRO/WIN

STEP 7-Micro/WIN - Siemens - [Symbol Table]

	Symbol	Address	Comment
1	POZADAVEK	M10.0	Nastavení požadavku na zapnutí nebo vypnutí motoru dle stisknutých tlačítek.
2	POVOLIT_START	M11.0	Povolení startu je uloženo do tohoto pomocného bitu.
3			
4	M_START	I0.0	Tlačítko pro start motoru.
5	M_STOP	I0.1	Tlačítko pro zastavení motoru.
6	JISTIC	I0.2	Signál od motorového jističe o přetížení motoru.
7			
8	NAPAJENI	Q0.0	Stykač pro napájení motoru.
9	HVEZDA	Q0.1	Rozběh motoru do hvězdy.
10	TROJUHELNÍK	Q0.2	Rozběh motoru do trojúhelníka.

Ready SYM1 POU Symbols / Row 1, Col 1 INS

STEP 7-Micro/WIN - Siemens - [SIMATIC LAD]

Network 1: Nastavení požadavku pro spuštění dle stisknutých tlačítek

```

M_START (NO) --- M_STOP (NC) --- POZADAVEK (S) 1

```

Network 2: Zrušení požadavku pro spuštění při stisknutí tlačítka STOP

```

M_STOP (NO) / --- POZADAVEK (R) 1

```

Network 3: Povolení spuštění motoru

```

POZADAVEK (NO) --- JISTIC (NC) --- POVOLIT_START (S)

```

Network 4: Sepnutí stykače pro přívod napájení motoru

```

POVOLIT_START (NO) --- NAPAJENI (S)

```

Network 5: Prodleva mezi rozběhem hvězda a trojúhelník

```

NAPAJENI (NO) --- T37 (TON)
                    IN      PT 100 ms
                    +50

```

Network 6: Sepnutí motorového spouštěče pro rozběh motoru do hvězdy

```

NAPAJENI (NO) --- T37 (NC) / --- HVEZDA (S)

```

Network 7: Sepnutí motorového spouštěče pro rozběh motoru do trojúhelníka

```

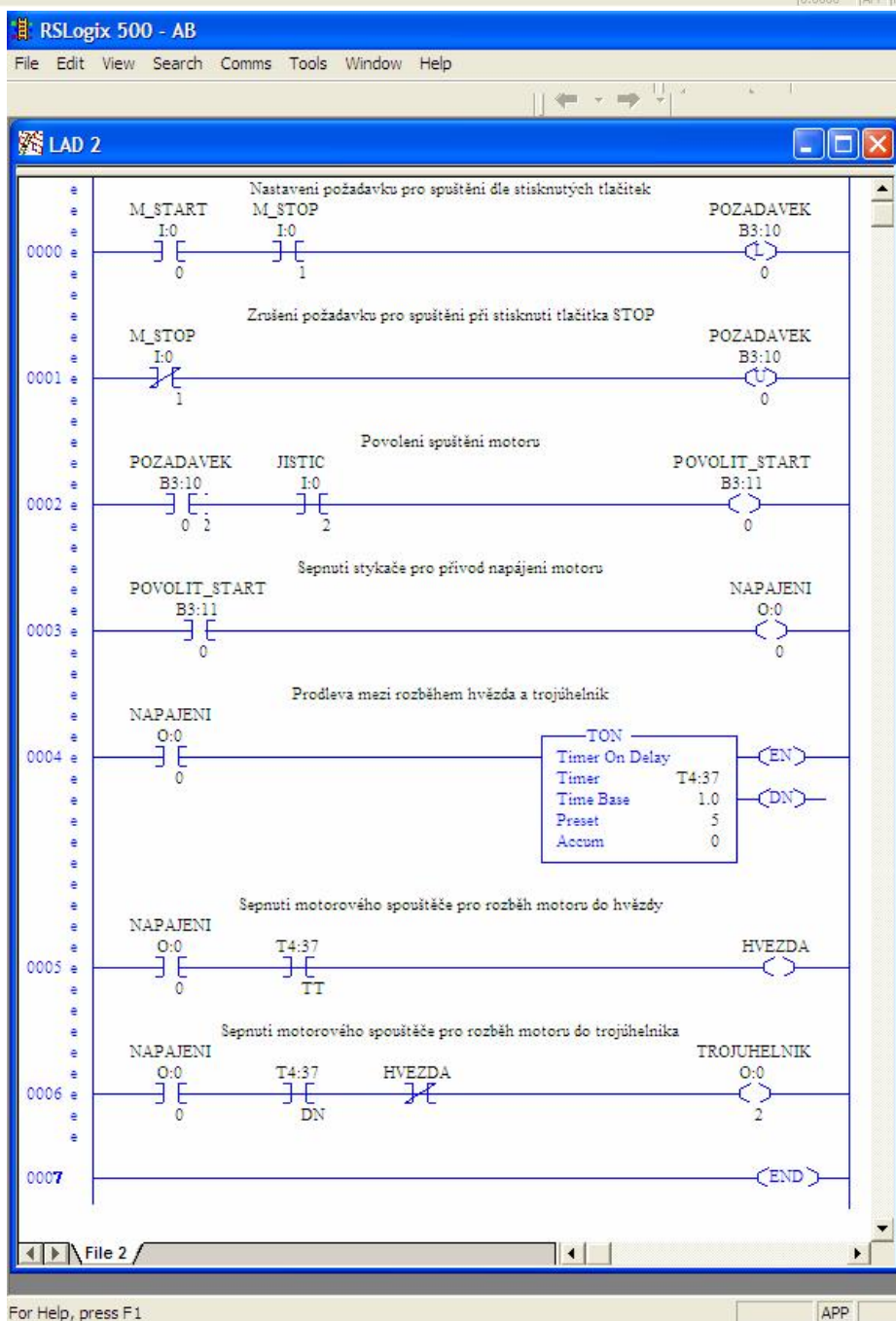
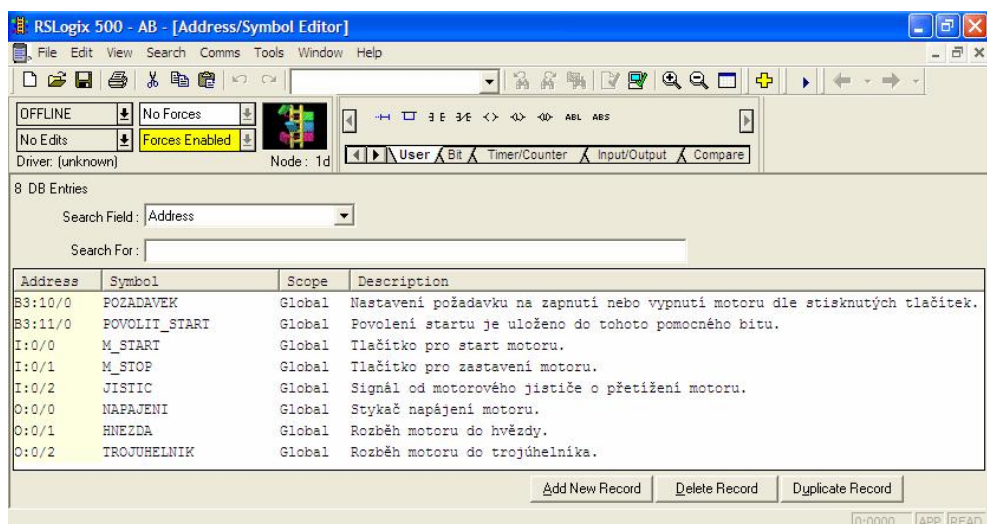
NAPAJENI (NO) --- T37 (NC) --- HVEZDA (NC) / --- TROJUHELNÍK (S)

```

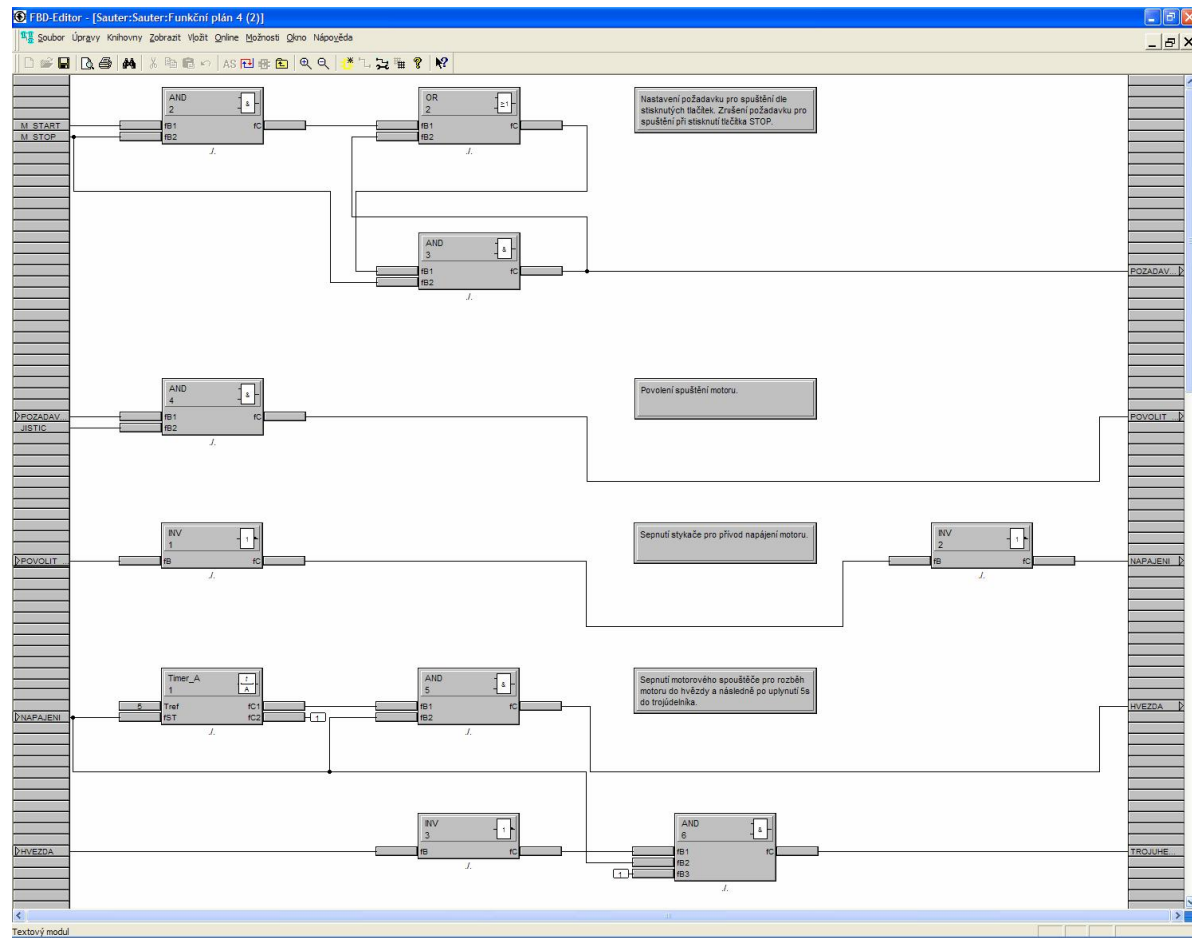
Network 8: Konec programu

OB 1 / Network 2 Title INS

7.4 ZPRACOVÁNÍ VE VÝVOJOVÉM PROSTŘEDÍ RSLOGIX 500



7.5 ZPRACOVÁNÍ VE VÝVOJOVÉM PROSTŘEDÍ EY3600-CASE



8 ZÁVĚR

Častým omylem je tvrzení či pocit, že programování je jen psaní programu ve zvoleném jazyce, že stačí naučit se vhodný programovací jazyk a po zadání úlohy sednout k počítači a psát program. Program pro PLC není ale samoúčelná kombinace znaků, grafické schéma nebo sled instrukcí či příkazů vyššího jazyka. Je to ucelený produkt a přitom neoddělitelná součást automatizačního projektu. Právě program oživuje celý automatizovaný systém – stroj nebo technologické zařízení. Kvalita programu rozhoduje o kvalitě řízení systému, o úspornosti, spolehlivosti provozu, o pohodlí obsluhy, spokojenosti koncového uživatele a výsledně o úspěšnosti celé zakázky.

Je chyba se domnívat, že okamžitě po zadání úlohy je třeba neprodleně začít s psáním instrukcí programu. Program lze sice tímto způsobem vytvořit, vynechání analytické a přípravné fáze se však projeví a vymstí později ve fázi ladění programu a při uvádění řízeného celku do provozu. Tedy ve fázi, kdy je mezi zúčastněnými nejvíce stresu a nervozity a prostředí je nejméně vhodné k soustředění se a ke hledání chyb, logických mezer a nesystematičností v řídicím programu.

Nesystematicky vytvořený a nedostatečně odladěný program je trvalým zdrojem problémů, reklamací a dodatečných nákladů. Proto se i při tvorbě programu vyplatí respektovat obvyklé vývojové fáze, obdobné jako u celkového projektu. Především úvodní analýzu požadavků, zadání úlohy a její algoritmizaci. Dále navázat systematickým návrhem koncepce a struktury programu, definováním podsystémů, modulů a funkčních bloků, které budou v programu použity a rozhraní mezi nimi. Přitom je důležité zvolit programovací jazyk vhodný pro danou úlohu. Teprve pak má smysl začít vytvářet vlastní program. Podobně jako při tvorbě hmatatelných produktů je i u softwaru základním požadavkem důsledné prověřování kvality v průběhu i na závěr celého procesu.

9 POUŽITÁ LITERATURA

- [1] Časopis AUTOMA, vydavatel FCC Public s. r. o., Pod Vodárenskou věží 4, 182 08 Praha 8, Česká republika
- [2] Časopis T+T technika a trh : vydavatel CCB, s.r.o., Okružní 17, 630 00 Brno
- [3] Časopis ELEKTRO, vydavatel: FCC PUBLIC s. r. o., Pod Vodárenskou věží 4, 182 08 Praha 8, Česká republika
- [4] Časopis AUTOMATIZACE vydavatel : Automatizace s.r.o., Mikulandská 7, 110 61 Praha 1, Česká republika
- [5] Dokumentace firmy Siemens
- [6] Dokumentace firmy Sauter
- [7] Dokumentace firmy Allan Bradley
- [8] Dokumentace firmy Controltech