SIEMENS

SIMATIC

Programovatelný automat S7-200 Systémový manuál

Předmluva, Obsah

Úvod do řídicích systémů SIMATIC S7–200 1

Úvod do programování 2

Instalace S7-200 3

Základní principy 4

Pojmy, konvence a možnosti programování 5

Instrukční soubor S7-200 6

Síťová komunikace 7

Návod pro odstraňování poruch hardwaru a adaptace softwaru 8

Řízení pohybu v otevřené smyčce u automatu S7-200 9

Tvorba programu pro modemový modul 10

Použití knihovny USS protokolu pro řízení pohonu MicroMaster 11

Použití knihovny protokolu Modbus 12

Použití receptur 13

Použití datových záznamů 14

Adaptivní PID a Ovládací panel PID 15

Přílohy

Index

Tento manuál má objednací číslo:
6ES7298-8FA24-8BH0

Vydání 06/2004
A5E00307987–01

Bezpečnostní pokyny

Tento manuál obsahuje upozornění, která byste měli dodržovat, abyste zajistili svou vlastní bezpečnost i ochranu výrobku a k němu připojeného zařízení. Tato upozornění jsou v manuálu zvýrazněna výstražným trojúhelníkem a podle úrovně nebezpečí jsou označena takto:



Nebezpečí

Nebezpečí znamená bezprostředně nebezpečnou situaci, která bez přijetí adekvátních opatření bude mít za následek smrt nebo vážný úraz.



Varování

Výstraha znamená potenciálně nebezpečnou situaci, která by bez přijetí adekvátních opatření mohla mít za následek smrt nebo vážný úraz.



Upozornění

Pozor! použito s výstražným bezpečnostním symbolem znamená potenciálně nebezpečnou situaci, která by bez přijetí adekvátních opatření mohla mít za následek lehký nebo střední úraz.

Upozornění

Upozornění! použito bez výstražného bezpečnostního symbolu znamená potenciálně nebezpečnou situaci, která by bez přijetí adekvátních opatření mohla mít za následek poškození majetku.

Poznámka

Znamená možnou situaci, která by bez přijetí adekvátních opatření mohla mít za následek nežádoucí výsledek nebo stav.

Kvalifikovaný personál

Montáž tohoto zařízení smí provádět pouze **kvalifikovaný personál**. Kvalifikované osoby jsou definovány jako osoby, které jsou oprávněny zprovozňovat, uzemňovat a opatřovat bezpečnostními tabulkami obvody, zařízení a systémy v souladu se zavedenými bezpečnostními zvyklostmi a normami.

Správné užívání

Věnujte pozornost následujícímu:



Varování

Toto zařízení a jeho komponenty mohou být používány pouze pro aplikace popsané v katalogu nebo v technických popisech a pouze ve spojení s těmi zařízeními nebo komponentami jiných výrobců, které byly schváleny nebo doporučeny společností Siemens.

Tento výrobek bude fungovat správně a bezpečně pouze tehdy, je–li přepravován, skladován, připraven k provozu a instalován správně a je provozován a udržován podle doporučení.

Ochranné známky

SIMATIC®, SIMATIC HMI® a SIMATIC NET® jsou registrované ochranné známky společnosti SIEMENS AG.

Některé jiné názvy použité v těchto dokumentech jsou také registrované ochranné známky; jejich použití třetí stranou pro její vlastní účely může znamenat porušení vlastnických práv.

Copyright Siemens AG 2004 Všechna práva vyhrazena

Kopírování, rozšiování nebo použití tohoto dokumentu nebo jeho obsahu bez výslovného písemného souhlasu autora není povoleno. Osoby, které toto poruší, jsou povinny nahradit škody. Všechna práva včetně práv tvořených udělením patentu nebo registrací užitkového modelu nebo vzoru jsou vyhrazena.

Siemens AG
Bereich Automation and Drives
Geschaeftsgebiet Industrial Automation Systems
Postfach 4848, D- 90327 Nürnberg

Odmítnutí odpovědnosti

Ověřili jsme si shodu obsahu tohoto manuálu s popisovaným hardwarem a softwarem. Protože odchylky není možné úplně vyloučit, nemůžeme garantovat plnou shodu. Údaje v tomto manuálu jsou však pravidelně kontrolovány a eventuální potřebné opravy jsou zahrnuty do následujících vydání. Návrhy na zlepšení jsou vítány.

E Siemens AG 2004
Změny technických údajů jsou vyhrazeny.

Předmluva

Cíl manuálu

Série S7-200 je řada malých -programovatelných automatů (mikro – PLC) určených k řízení v různých automatizačních aplikacích. Díky kompaktnímu designu, nízkým nákladům a výkonnému instrukčnímu souboru je S7-200 perfektním řešením pro řízení malých aplikací. Široká nabídka různých modelů S7-200 a programovací nástroj na bázi Windows - poskytují flexibilitu potřebnou pro řešení vašich automatizačních potřeb.

Tento manuál poskytuje informace o instalaci a programování mikro – PLC S7-200 a je určen pro inženýry, programátory, montážní pracovníky a elektrikáře, kteří mají obecné znalosti o programovatelných automatech.

Základní požadované znalosti

K pochopení tohoto manuálu je nutné mít všeobecné znalosti o programovatelných automatech.

Rozsah manuálu

Tento manuál je platný pro STEP 7 – Micro/WIN, verze 4.0 a pro řadu CPU S7-200. Úplný seznam výrobků S7-200 a objednacích čísel popsaných v tomto manuálu viz přílohu A.

Změny oproti předchozí verzi

Tento manuál byl revidován a zahrnuje:

- CPU S7-200 modely CPU 221, CPU 222, CPU 224, CPU 224XP a CPU 226, které zahrnují:

  Novou hardwarovou podporu CPU: možnost vypnout editaci v režimu RUN a tím získat více paměti pro program, CPU 224XP podporuje integrovaný analogový I/O a dva komunikační porty. CPU 226 obsahuje další vstupní filtry a zachycování pulzů.

- Podporu nového paměťového modulu: utilitu Prohlížeče průzkumníka S7-200, přenosy paměťových modulů, porovnání a volby programování

- STEP 7 – Micro/WIN, verze 4.0, 32bitový- programovací softwarový balík pro S7-200 obsahuje:

  Nové a vylepšené nástroje, které podporují nejnovější doplňky CPU: Ovládací panel automatického ladění PID, zabudovaný Průvodce řízení polohy PLC, Průvodce datovým záznamem a Průvodce recepturou.

  Nový diagnostický nástroj: konfigurování diagnostické kontrolky

  Nové instrukce: letní čas (READ_RTCX a SET_RTCX), intervalové časovače (BITIM, CITIM), vymazání události přerušení (CLR_EVNT) a diagnostická kontrolka (DIAG_LED)

  POU a doplňky knihoven: nové řetězcové konstanty, přidaná podpora nepřímého adresování více typů pamětí, zlepšená podpora parametrizace čtení a zápisu knihovny USS pro pohony Siemens master.

  Vylepšený blok dat: stránky bloku dat, automatická inkrementace bloku dat

  Zlepšená využitelnost programovacího balíku STEP 7 – Micro/WIN

Certifikace

Řada SIMATIC S7-200 splňuje následující certifikaci:

- ❏ Underwriters Laboratories, Inc. UL 508 (průmyslová řídící zařízení), Registrační číslo E75310
- ❏ Canadian Standards Association (Kanadské sdružení pro normy): CSA C22.2 číslo 142 (Zařízení pro řízení procesu)
- ❏ Factory Mutual Research: Třída číslo 3600, třída číslo 3611, FM třída I, oddělení 2, nebezpečná místa skupiny A, B, C a D, T4A a třída I, pásmo 2, IIC, T4



Tip

Řada SIMATIC S7-200 splňuje normu CSA.

Logo cULus znamená, že S7-200 byl testován a certifikován laboratořemi Underwriters Laboratories (UL) podle norem UL 508 a CSA 22.2 č. 142.

Označení CE

Řada SIMATIC S7-200 splňuje požadavky a pokyny pro ochranu uvedené v následujících směrnicích EU:

- ❏ Směrnice EC 73/23/EEC ”Směrnice pro nízké napětí”
- ❏ Směrnice EC 89/336/EEC ”Směrnice pro EMC”

Poraďte se s vaším oblastním zástupcem firmy Siemens ohledně shody výrobku se směrnicí ATEX Evropského společenství 94/9/EC.

C-Tick

Výrobky SIMATIC S7-200 splňují požadavky australské normy AS/NZS 2064.

Normy:

Výrobky SIMATIC S7-200 splňují požadavky a kritéria IEC 61131–2, Programovatelné automaty – požadavky na zařízení.

Další informace o shodě jsou v příloze A.

Umístění této dokumentace v informačním prostředí

Řada výrobků	Dokumentace	Objednací číslo
S7–200	Uživatelský manuál pro dotykový panel TP070 (v angličtině)	6AV6 591–1DC01–0AB0
	Návod k obsluze TP170 micro (v angličtině)	6AV6 691–1DB01–0AB0
	Manuál pro komunikaci S7-200 v protokolu Point-to-Point Interface (v angličtině/němčině)	6ES7 298–8GA00–8XH 0
	Manuál pro komunikační procesor CP 243–2 SIMATIC NET AS-Interface Master (v angličtině)	6GK7 243–2AX00–8BA 0
	CP 243–1 IT Internetový modul (s elektronickou dokumentací na CD)	6GK7 243–1GX00–0XE0
	CP 243–1 Ethernetový modul (s elektronickou dokumentací na CD)	6GK7 243–1GX00–0XE0
	Systémový manuál pro programovatelný automat S7-200 (v angličtině)	6ES7 298–8FA24–8BH0

Jak na to

Pokud - mikro PLC S7-200 používáte poprvé, měli byste si přečíst celý *Systémový manuál pro programovatelný automat S7-200* . Jste – li zkušený uživatel, stačí vám podle obsahu nebo rejstříku vyhledat konkrétní informace.

Systémový manuál pro programovatelný automat S7-200 je rozdělen na následující témata:

- Kapitola 1 (Úvod do řídicích systémů SIMATIC S7 – 200) uvádí přehled některých vlastností řady malých programovatelných automatů S7-200 (micro PLC).
- Kapitola 2 (Úvod do programování) učí vytvořit vzorový řídicí program pro S7-200 a provést jeho download.
- Kapitola 3 (Instalace S7-200) obsahuje rozměry a základní návody pro instalaci modulu CPU S7-200 a rozšiřovacích vstupních/výstupních modulů.
- Kapitola 4 (Základní principy) poskytuje informace o provozu S7-200.
- Kapitola 5 (Pojmy, konvence a funkce programování) uvádí informace o funkcích STEP 7 – Micro/WIN, programových editorech a typech instrukcí (IEC 1131-3 nebo SIMATIC), typech dat S7-200 a návodech pro tvorbu programů.
- Kapitola 6 (Soubor instrukcí pro S7-200) uvádí popisy a příklady programovacích instrukcí podporovaných S7-200.
- Kapitola 7 (Síťová komunikace) poskytuje informace pro nastavení různých konfigurací síťe podporovaných automatem S7-200.
- Kapitola 8 (Návod pro odstraňování poruch hardwaru a nástroje pro ladění softwaru) uvádí informace pro odstranění problémů hardwaru S7-200 a informace o funkcích STEP 7 – Micro/WIN, které pomáhají při ladění uživatelského programu.
- Kapitola 9 (Řízení pohybu otevřené smyčky s automatem S7-200) uvádí informace o třech způsobech řízení pohybu otevřené smyčky: Pulzní šířková modulace, výstup série impulzů a polohovací modul EM 253.
- Kapitola 10 (Tvorba programu pro modemový modul) uvádí informace o instrukcích a pomůcce wizard, které se používají při tvorbě programu pro modemový modul EM 241.
- Kapitola 11 (Použití knihovny protokolu USS pro řízení pohonu MicroMaster) poskytuje informace o instrukcích používaných pro tvorbu a řízení programu pro pohon MicroMaster. Informuje také o tom, jak konfigurovat pohony MicroMaster 3 a MicroMaster 4.
- Kapitola 12 (Užívání knihovny protokolu Modbus) uvádí informace o instrukcích používaných pro tvorbu programu, který pro komunikaci používá protokol Modbus.
- Kapitola 13 (Použití receptur) uvádí informace o organizaci a načítání receptur automatizačního programu do paměťového modulu.
- Kapitola 14 (Použití datových záznamů) uvádí informace o ukládání naměřených procesních dat do paměťového modulu.
- Kapitola 15 (Automatické ladění PID a ovládací panel automatického ladění PID) uvádí informace o používání těchto funkcí ke zvýšení funkčnosti a k usnadnění použití funkce PID v automatu S7-200.
- V příloze A (Technická specifikace) jsou technické informace a tabulky s údaji o hardwaru automatu S7-200.

Ostatní přílohy obsahují další rychle dostupné informace, jako jsou popis chybových kódu, popis oblasti speciální paměti (SM), čísla součástí pro objednávání zařízení S7-200 a doby provádění instrukcí STL.

Kromě tohoto manuálu je software STEP 7 – Micro/WIN vybaven i rozsáhlou online nápovědou pro uvedení do programování automatu S7-200. Součástí koupě STEP 7 – Micro/WIN je i bezplatné dokumentační CD. Na něm najdete tipy pro použití, elektronickou verzi tohoto manuálu a jiné informace.

Online nápověda

Stačí stisknout klávesu a pomoc je tu! Stisknutím klávesy F1 získáte přístup k rozsáhlé online nápovědě pro STEP 7–Micro/WIN. Online nápověda obsahuje užitečné informace o tom, jak začít programovat S7-200, i mnoho dalších témat.

Elektronický manuál

Na dokumentačním CD je k dispozici elektronická verze tohoto Systémového manuálu pro S7-200. Můžete ji nahrát do svého počítače a získáte tak snadný přístup k informacím obsaženým v manuálu při práci se softwarem STEP 7–Micro/WIN.

Tipy pro programování

Dokumentační CD obsahuje Programovací tipy, sadu příkladů použití se vzorovými programy. Posouzení nebo úprava těchto příkladů vám může pomoci nalézt účinná nebo úplně nová řešení pro vaši vlastní aplikaci. Nejaktuálnější verzi Programovacích tipů také můžete najít na internetové stránce S7-200.

Recyklace a likvidace

Obraťte se na firmu autorizovanou pro likvidaci elektronického odpadu, která vám sdělí informace o environmentálně bezpečné recyklaci a likvidaci vašeho zařízení.

Další podpora

Místní zastoupení nebo distributor Siemens

Pro pomoc při řešení jakýchkoliv technických otázek, školení pro výrobky S7-200 a objednávky výrobků S7-200 kontaktujte místního distributora nebo zastoupení Siemens. Protože obchodní zástupci jsou technicky vyškolení a mají konkrétní znalosti o vašem provozu, technologii a průmyslovém odvětví i o jednotlivých výrobcích Siemens, které používáte, mohou vám dát nejrychlejší a nejefektivnější odpovědi na jakékoliv problémy, se kterými byste se mohli setkat.

Servis a podpora na Internetu

Kromě dokumentace nabízíme online podporu "Know-how" na internetové adrese:

http://www.siemens.com/automation/service&support

kde najdete:

- www.siemens.cz/micro *české internetové stránky pro podporu malých řídicích systémů LOGO! a SIMATIC S7-200*
- www.siemens.com/S7–200 *informace o výrobku S7-200*

  Internetová stránka S7-200 obsahuje často kladené dotazy (FAQ), Programovací tipy (příklady použití a vzorové programy), informace o nově nabízených výrobcích a aktualizace nebo download pro výrobky.
- Zpravodaj, který vám nepřetržitě poskytuje aktuální informace o vašich výrobcích.--
- Správné dokumenty pomocí naší vyhledávací funkce (Search) v Service & Support.
- Fórum, kde si uživatelé a odborníci z celého světa vyměňují své zkušenosti.
- Váš místní zástupce pro Automation & Drives.
- Informace o službách zákazníkům, opravách, náhradních dílech atd. pod "Services".

Technické služby

K dispozici je vám také vysoce kvalifikovaný personál Centra technických služeb S7-200, který vám pomůže vyřešit jakýkoliv problém, se kterým byste se mohli setkat. Můžete jej volat celých 24 hodin, 7 dnů v týdnu.

Technická podpora A&D

Celosvětově, 24 hodin denně:



Celosvětová (Nürnberg) **Technická podpora** 24 hodin denně, 365 dnů v roce Tel.: +49 (180) 5050-222 Fax: +49 (180) 5050-223 mailto:adsupport@siemens.com GMT: +1:00	**Spojené státy** (Johnson City) **Technická podpora a autorizace** Místní čas: pondělí-pátek 8:00 – 17:00 Tel.: +1 (423) 262 2522 +1 (800) 333–7421 (jen USA) Fax: +1 (423) 262 2289 mailto:simatic.hotline@sea.siemens.com GMT: –5:00	**Asie** / **Austrálie** (Peking) **Technická podpora a autorizace** Místní čas: pondělí-pátek 8:00 – 17:00 Tel.: +86 10 64 75 75 75 Fax: +86 10 64 74 74 74 mailto:adsupport.asia@siemens.com GMT: +8:00
Evropa / Afrika (Nürnberg) **Autorizace** Místní čas: pondělí-pátek 8:00 - 17:00 Tel.: +49 (180) 5050-222 Fax: +49 (180) 5050-223 mailto:adsupport@siemens.com GMT: +1:00	**Česká republika** Místní čas: pondělí-pátek 8:00 – 17:00 Tel.: +420 233 32 411 +420 541 191 740 mailto:micro@rg.siemens.cz GMT: +1:00	
Komunikace na horkých linkách SIMATIC a autorizace probíhá obecně v němčině a angličtině.		

Obsah

1 Úvod do řídicích systémů SIMATIC S7–200 1

Co je nového? 2
S7-200 CPU 2
Rozšiřovací moduly S7-200 3
Programovací balík STEP 7–Micro/WIN 4
Možnosti komunikace 4
Operátorské panely 5

2 Úvod do programování 7

Připojení CPU S7-200 8
Vytvoření vzorového programu 10
Download vzorového programu 14
Uvedení S7-200 do režimu RUN 14

3 Instalace S7-200 15

Návod pro instalaci zařízení S7-200 16
Montáž a demontáž modulů S7-200 17
Návod pro uzemnění a elektrické zapojení 20

4 Základní principy 23

Jak S7-200 vykonává řídicí program 24
Přístup k datům 26
Ukládání a obnova dat 36
Výběr pracovního režimu CPU S7-200 40
Použití Průvodníka S7-200 40
Vlastnosti programovatelného automatu S7-200 41

5 Pojmy, konvence a možnosti programování 51

Návod pro návrh systému 52
Základní prvky programu 53
Použití STEP 7–Micro/WIN pro vytvoření programu 55
Volba mezi instrukčním souborem SIMATIC a IEC 1131–3 57
Vysvětlení konvencí použitých v programových editorech 58
Použití průvodců jako pomůcky pro tvorbu řídícího programu 60
Ošetření chyb S7-200 60
Přiřazení adres a počátečních hodnot v editoru datového bloku 62
Použití tabulky symbolů pro symbolické adresování proměnných 62
Použití lokálních proměnných 63
Použití stavové diagramu pro monitorování uživatelského programu 63
Vytvoření knihovny instrukcí 64
Funkce pro adaptaci uživatelského programu 64

6 Instrukční soubor S7-200 65

Konvence použité při popisu instrukcí 67
Rozsahy a vlastnosti paměti S7-200 68
Instrukce bitové logiky 70
Kontakty 70
Cívky 73
Práce se zásobníkem 75
Bistabilní klopný obvod 77
Instrukce hodin 78
Instrukce pro komunikaci 81
Čtení a zápis do sítě 81
Vysílání a příjem (Freeport) 86
Zjištění a nastavení adresy portu 95
Instrukce porovnávání 96
Porovnávání číselných hodnot 96
Porovnávání řetězce 98
Instrukce pro konverzi 99
Standardní konverze 99
Konverze na ASCII 103
Konverze řetězců 107
Zakódování a dekódování 112
Instrukce čítače 113
Instrukce čítače SIMATIC 113
Instrukce čítače IEC 116
Instrukce vysokorychlostního čítače 118
Instrukce pulzního výstupu 133
Matematické instrukce 140
Sčítání, odčítání, násobení a dělení 140
Násobení celých čísel s 32bitovým výsledkem a dělení celých čísel se zbytkem 142
Numerické funkce 143
Inkrementace a dekrementace 144
Instrukce PID algoritmu 145
Instrukce přerušení 153
Instrukce logické operace 161
Inverze 161
AND, OR a X OR 162
Instrukce přesunu 164
Přesun byte, word, double word nebo real 164
Okamžitý přesun bytu (čtení a zápis) 165
Přesun bloku 166
Řízení programu 167
Podmíněný konec programu 167
Stop 167
Reset Watchdogu 167
Smyčka For–Next 169
Skokové instrukce 171
Sekvenčního řídící relé (SCR) 172
Instrukce diagnostických kontrolek 178

Instrukce posunu a rotace 179
Posunu vpravo a vlevo 179
Rotace vpravo a vlevo 179
Posunutí bitů registru 181
Výměna bytů 183
Instrukce pro řetězce 184
Instrukce pro práci s tabulkou 189
Přidávání do tabulky 189
Instrukce FIFO a-LIFO 190
Vyplnění paměti 192
Prohledávání tabulky 193
Instrukce časovače 196
Instrukce časovače SIMATIC 196
Instrukce časovače IEC 201
Intervalové hodiny 203
Instrukce podprogramů 204

7 Síťová komunikace 209

Vysvětlení základů síťové komunikace S7-200 210
Výběr síťového komunikačního protokolu 214
Instalace a odstranění komunikačních rozhraní 220
Návrh sítě 221
Vytvoření protokolů definovaných uživatelem v režimu Freeport 225
Použití modemu a STEP 7–Micro/WIN 228
Témata pro pokročilé 233
Konfigurace RS-232/PPI Multi-Master kabelu pro dálkové ovládání 239

8 Návod pro odstraňování poruch hardwaru a ladění softwaru 243

Funkce pro adaptaci uživatelského programu 244
Zobrazení stavu programu 246
Použití stavové tabulky pro monitorování a modifikaci dat 247
Force (vnucení) specifických hodnot 248
Zadání běhu programu o zadaný počet programových cyklů 248
Návod pro odstraňování poruch hardwaru 249

9 Řízení pohybu v otevřené smyčce u automatu S7-200 251

Přehled 252
Použití výstupu PWM (pulzní šířková modulace) 253
Základní informace pro řízení pohybu v otevřené smyčce pomocí krokových motorů a servomotorů 255
Instrukce vytvořené pomocí ”Průvodce polohováním” 260
Chybové kódy pro instrukce PTO 264
Funkce polohovacího modulu 265
Konfigurace polohovacího modulu 267
Polohovací instrukce vytvořené pomocí ”Průvodce polohováním” 273
Vzorové programy pro polohovací modul 285

Monitorování polohovacího modulu “Control panelem EM 253” 290
Chybové kódy pro polohovací modul a polohovací instrukce 292
Témata pro pokročilé 294
Vysvětlení režimů vyhledávání RP podporovaných polohovacím modulem 303

10 Tvorba programu pro modemový modul 307

Funkce modemového modulu 308
Použití průvodce pro konfiguraci modemového modulu 314
Přehled instrukcí a omezení pro modem 318
Instrukce pro modemový modul 319
Vzorový program 323
CPU S7-200 podporující inteligentní moduly 323
Byty speciální paměti pro modemový modul 323
Témata pro pokročilé 325
Formát telefonního čísla pro zprávu 327
Formát textové zprávy 328
Formát zprávy pro přenos dat CPU 329

11 Použití knihovny USS protokolu pro řízení pohonu MicroMaster 331

Požadavky použití USS protokolu 332
Výpočet doby potřebné pro komunikaci s pohonem 332
Použití instrukcí USS 333
Instrukce USS protokolu 334
Vzorové programy USS protokolu 341
Chybové kódy pro provádění USS 342
Připojení a nastavení pohonu MicroMaster řady 3 342
Připojení a nastavení pohonu MicroMaster řady 4 345

12 Použití knihovny protokolu Modbus 347

Požadavky použití protokolu Modbus 348
Inicializace a doba provádění protokolu Modbus 348
Adresování Modbus 349
Použití instrukcí protokolu Modbus slave 350
Instrukce protokolu Modbus slave 351

13 Použití receptur 355

Přehled 356
Definice receptur a terminologie 357
Použití “Průvodce recepturami” 357
Instrukce vytvořené “Průvodcem recepturami” 361

14 Použití datového záznamu 363

Přehled 364
Použití “Průvodce datovým záznamem” 365
Instrukce vytvořená “Průvodcem datovým záznamem” 369

15 Adaptivní PID a Ovládací panel PID 371

Vysvětlení adaptace PID 372
Rozšířená tabulka smyčky 372
Požadavky 375
Automatická hystereze a automatická odchylka 375
Sekvence adaptace 376
Výjimečné stavy 376
Poznámky týkající se PV mimo rozsah (výsledný kód 3) 378
Ovládací panel PID 378

A Technická specifikace 381

Obecné technické specifikace 382
Specifikace CPU 385
Specifikace digitálních rozšiřovacích modulů 394
Specifikace analogových rozšiřovacích modulů 400
Specifikace termočlánkových a RTD rozšiřovacích modulů 411
Specifikace modulu EM 277 PROFIBUS–DP 423
Specifikace modemového modulu EM 241 435
Specifikace polohovacího modulu EM 253 437
Specifikace ethernetového modulu (CP 243–1) 443
Specifikace internetového modulu (CP 243–1 IT) 445
Specifikace modulu AS–Interface (CP 243–2) 448
Volitelné zásuvné moduly 450
Kabel pro rozšiřovací moduly 451
RS-232/PPI Multi-Master nebo USB/PPI Multi-Master kabel 452
Simulátory vstupu 456

B Výpočet energetické bilance 457

C Chybové kódy 461

Kódy a hlášení kritických chyb 462
Problémy programování při běhu programu 463
Porušení pravidel kompilace 464

D Speciální paměťové bity (SM) 465

SMB0: Stavové bity 466
SMB1: Stavové bity 466
SMB2: Znak přijatý v režimu Freeport 467
SMB3: Chyba parity v režimu Freeport 467
SMB4: Přeplnění fronty 467
SMB5: Stav vstupů a výstupů 468
SMB6: Registr ID CPU 468
SMB7: Rezervován 468
SMB8 až SMB21: ID a chybové registry I/O modulu 469
SMW22 až SMW26: Délka programového cyklu 470
SMB28 a SMB29: Analogové potenciometry 470
SMB30 a SMB130: Řídicí registry pro Freeport 470
SMB31 a SMW32: Řízení zápisu do trvalé paměti (EEPROM) 471
SMB34 a SMB35: Registry časového intervalu pro časovaná přerušení 471
SMB36 až SMB65: Registr HSC0, HSC1 a HSC2 471
SMB66 až SMB85: Registry PTO/PWM 473
SMB86 až SMB94 a SMB186 až SMB194: Řízení příjmu zpráv 474
SMW98: Chyby na rozšiřovací I/O sběrnici 475
SMB130: Řídicí registr pro Freeport (viz SMB30) 475
SMB131 až SMB165: Registr HSC3, HSC4 a HSC5 475
SMB166 až SMB185: Tabulka definice profilu PTO0, PTO1 476
SMB186 až SMB194: Řízení příjmu zpráv (viz SMB86 až SMB94) 476
SMB200 až SMB549: Stav inteligentních modulů 477

E Objednací čísla S7-200 479

F Doby provádění instrukcí STL 483

G Stručný přehled informací o S7-200 489

1

Úvod do řídicích systémů SIMATIC S7–200

SIMATIC S7-200 je řada malých programovatelných automatů (mikro–PLC), určených k řízení v různých automatizačních aplikacích.

Zařízení S7-200 monitoruje vstupy a řídí výstupy pomocí uživatelského programu, který může obsahovat Booleovu logiku, počítání, časování, složité matematické operace a komunikaci s jinými inteligentními zařízeními. Kompaktní design, flexibilní konfigurace a výkonný instrukční soubor jsou důvody, proč je zařízení S7-200 optimálním řešením pro řízení široké škály aplikací.

Přehled kapitoly

Co je nového? 2
S7-200 CPU 2
Rozšiřovací moduly S7-200 3
Programovací balík STEP 7–Micro/WIN 4
Možnosti komunikace 4
Operátorské panely 5

Co je nového?

Zařízení SIMATIC S7-200 obsahuje tyto nové funkce. V tabulce 1-1 je seznam CPU S7-200, které podporují tyto nové funkce.

- CPU S7-200 modely CPU 221, CPU 222, CPU 224, CPU 224XP a CPU 226 zahrnují:

  Novou hardwarovou podporu CPU: možnost vypnout editaci v režimu RUN a tím získat více paměti pro program, CPU 224XP podporuje integrovaný analogový I/O a dva komunikační porty. CPU 226 obsahuje další vstupní filtry a zachycování pulzů.

- Podporu nového paměťového modulu: Utilitu prohlížeče průzkumníka S7-200, přenosy paměťových modulů, porovnání a volby programování

- STEP 7–Micro/WIN, verze 4.0, 32bitový programovací softwarový balík pro S7-200 obsahuje:

  Nové a vylepšené nástroje, které podporují nejnovější doplňky CPU: Ovládací panel automatického ladění PID, zabudovaný průvodce řízení polohy PLC, průvodce zápisem dat a průvodce recepturou.

  Nový diagnostický nástroj: programovatelná diagnostická kontrolka

  Nové instrukce: letní čas (READ_RTCX a SET_RTCX), intervalové hodiny (BITIM, CITIM), vymazání události pro přerušení (CLR_EVNT) a diagnostická kontrolka (DIAG_LED).

  POU a doplňky knihoven: nové řetězcové konstanty, přidaná podpora nepřímého adresování u více typů pamětí, zlepšená podpora parametrizace čtení a zápisu knihovny USS pro diskové jednotky master Siemens

  Vylepšený blok dat: stránky bloku dat, automatická inkrementace bloku dat

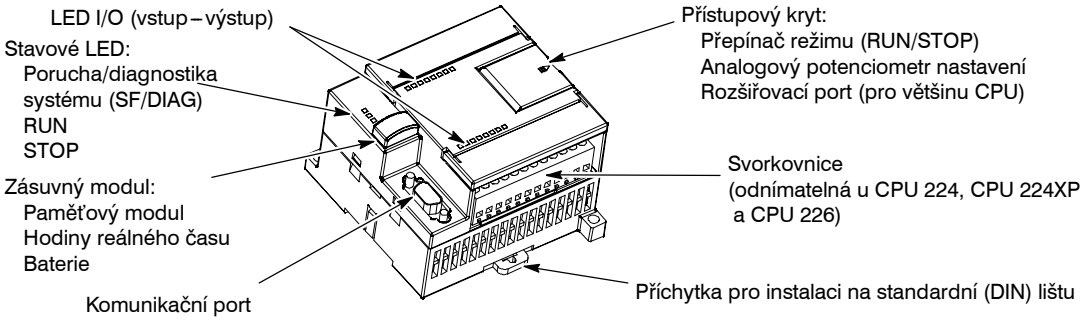
  Zlepšená využitelnost programovacího balíku STEP 7–Micro/WIN

Tabulka 1-1 S7-200 CPU

S7-200 CPU	Objednací číslo
CPU 221 DC/DC/DC 6 vstupů/4 výstupy	6ES7 211–0AA23–0XB0
CPU 221 AC/DC/relé 6 vstupů/4 relé	6ES7 211–0BA23–0XB0
CPU 222 DC/DC/DC 8 vstupů/6 výstupů	6ES7 212–1AB23–0XB0
CPU 222 AC/DC/relé 8 vstupů/6 relé	6ES7 212–1BB23–0XB0
CPU 224 DC/DC/DC 14 vstupů/10 výstupů	6ES7 214–1AD23–0XB0
CPU 224 AC/DC/relé 14 vstupů/10 relé	6ES7 214–1BD23–0XB0
CPU 224XP DC/DC/DC 14 vstupů/10 výstupů	6ES7 214–1AD23–0XB0
CPU 224XP AC/DC/relé 14 vstupů/10 relé	6ES7 214–1BD23–0XB0
CPU 226 DC/DC/DC 24 vstupů/16 výstupů	6ES7 216–2AD23–0XB0
CPU 226 AC/DC/relé 24 vstupů/16 relé	6ES7 216–2BD23–0XB0

S7-200 CPU

S7-200 obsahuje mikroprocesor, integrovaný napájecí zdroj, vstupní a výstupní obvody v kompaktním pouzdru, které tak tvoří výkonný programovatelný automat (mikro–PLC). Viz obrázek 1-1. Po downloadu vašeho programu bude S7-200 obsahovat logiku potřebnou k monitorování a řízení vstupních a výstupních zařízení aplikace.



Obr. 1-1 S7-200 Mikro–PLC

Siemens dodává různé modely S7-200 s různými parametry a schopnostmi, které pomáhají vytvořit efektivní řešení pro vaše aplikace. Tabulka 1-2 stručně srovnává některé parametry CPU. Podrobné informace o konkrétním CPU jsou uvedeny v příloze A.

Tabulka 1-2 Porovnání modelů S7-200 CPU

Parametr	CPU 221	CPU 222	CPU 224	CPU 224XP	CPU 226
Rozměry (mm)	90 x 80 x 62	90 x 80 x 62	120,5 x 80 x 62	140 x 80 x 62	190 x 80 x 62
Paměť pro program: s editací v režimu RUN bez editace v režimu RUN	 4096 bytů 4096 bytů	 4096 bytů 4096 bytů	 8192 bytů 12288 bytů	 12288 bytů 16384 bytů	 16384 bytů 24576 bytů
Paměť pro data	2048 bytů	2048 bytů	8192 bytů	10240 bytů	10240 bytů
Zálohování dat	standardně 50 hodin	standardně 50 hodin	standardně 100 hodin	standardně 100 hodin	standardně 100 hodin
Integrované I/O digitální analogové	 6 vst./4 výst. –	 8 vst./6 výst. –	 14 vst./10 výst. –	 14 vst./10 výst. 2 vst./1 výst.	 24 vst./16 výst. –
Rozšiřovací moduly	0 modulů	2 moduly[1]	7 modulů[1]	7 modulů[1]	7 modulů[1]
Vysokorychlostní čítače Jednofázové Dvoufázové	 4 při 30 kHz 2 při 20 kHz	 4 při 30 kHz 2 při 20 kHz	 6 při 30 kHz 4 při 20 kHz	 4 při 30 kHz 2 při 200 kHz 3 při 20 kHz 1 při 100 kHz	 6 při 30 kHz 4 při 20 kHz
Pulzní výstupy (DC)	2 při 20 kHz	2 při 20 kHz	2 při 20 kHz	2 při 100 kHz	2 při 20 kHz
Analogové potenciometry	1	1	2	2	2
Hodiny reálného času	Zásuvný modul	Zásuvný modul	Integrované	Integrované	Integrované
Komunikační porty	1 RS–485	1 RS–485	1 RS–485	2 RS–485	2 RS–485
Matematika s pohyblivou řádovou čárkou	Ano				
Velikost registru obrazu digitálních I/O	256 (128 vst., 128 výst.)				
Rychlost booleovských instrukcí	0,22 mikrosekund/instrukce				

1 Musíte spočítat energetickou bilanci, abyste stanovili, kolik energie (nebo proudu) může CPU S7-200 poskytnout dané konfiguraci. Pokud je překročena energetická bilance CPU, možná nebudete moci připojit maximální počet modulů. Požadavky na napájení CPU a rozšiřovacích modulů najdete v příloze A a výpočet energetické bilance v příloze B.

Rozšiřovací moduly S7-200

Pro lepší řešení požadavků vaší aplikace obsahuje řada S7-200 širokou škálu rozšiřovacích modulů. Těmito rozšiřovacími moduly můžete do S7-200 přidat další funkce. V tabulce 1-3 je seznam rozšiřovacích modulů, které jsou v současné době k dispozici. Podrobné informace o konkrétním modulu jsou uvedeny v příloze A.

Tabulka 1-3 Rozšiřovací moduly S7-200

Rozšiřovací moduly		Typy		
Digitální moduly	vstup	8 x DC vst	8 x AC vst	16 x DC vst
	výstup	4 x DC 8 x DC výst	4 x relé 8 x AC výst	8 x relé
	kombinace	4 x DC vst / 4 x DC výst	8 x DC vst / 8 x DC výst	16 x DC vst / 16 x DC výst
		4 x DC vst / 4 x relé	8 x DC vst / 8 x relé	16 x DC vst / 16 x relé
Analogové moduly	vstup	4 x analogový vstup	4 x termočlánek vstup	2 x RTD vstup
	výstup	2 x analogový výstup		
	kombinace	4 x analogový vstup / 1 x analogový výstup		
Inteligentní moduly		Polohovací Ethernetový	Modemový Internetový	PROFIBUS-DP
Jiné moduly		AS interface		

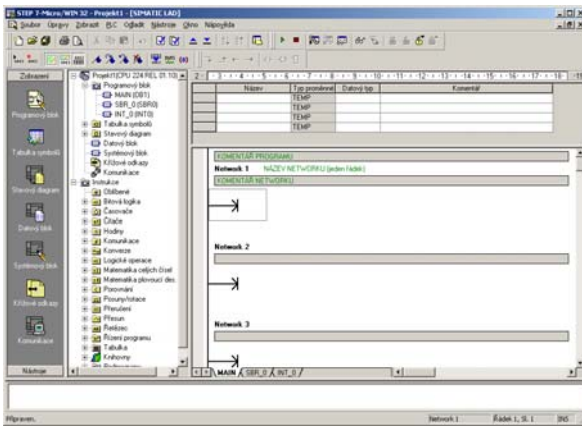
Programovací balík STEP 7–Micro/WIN

Programovací balík STEP 7–Micro/WIN poskytuje uživatelsky příjemné prostředí pro vytváření, editaci a monitorování logiky, nutné k řízení aplikace. STEP 7–Micro/WIN obsahuje tři programové editory pro komfort a efektivitu při vytváření řídicího programu pro aplikaci. Abyste lépe nalezli informace, které potřebujete, je STEP 7–Micro/WIN vybaven rozsáhlým systémem online nápovědy a CD dokumentací, která obsahuje elektronickou verzi tohoto manuálu, tipy pro aplikace a další užitečné informace.

Požadavky na počítač

STEP 7–Micro/WIN běží na osobním počítači nebo na programovacím přístroji Siemens (např. PG 760). Váš počítač nebo programovací přístroj by měly splňovat minimálně tyto požadavky:

- ❏ Operační systém:
  Windows 2000, Windows XP
  (Professional nebo Home)
- ❏ Alespoň 100 MB volného prostoru na HDD
- ❏ Myš (doporučeno)



Obr. 1-2 STEP 7–Micro/WIN

Instalace STEP 7–Micro/WIN

Vložte CD STEP 7–Micro/WIN CD do mechaniky CD-ROM počítače. Automaticky se spustí pomocník instalace, který vás povede instalačním procesem. Další informace o instalaci STEP 7–Micro/WIN získáte v souboru ”Readme”.

Tip
K instalaci STEP 7–Micro/WIN na operační systém Windows 2000 nebo Windows XP (Professional nebo Home) se musíte přihlásit s oprávněním administrátora.

Možnosti 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.

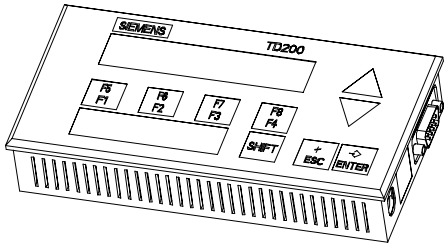
Operátorské panely

Textový displej (TD 200 a TD 200C)

TD 200 a TD 200C jsou textová zobrazovací zařízení o 2 řádcích a 20 znacích, která mohou být připojena k S7-200. Pomocí průvodce TD 200 můžete snadno programovat S7-200, aby zobrazoval textové zprávy a jiná data týkající se aplikace.

TD 200 a TD 200C poskytují základní rozhraní tím, že umožňují prohlížet, sledovat a měnit proměnné procesu vaší aplikace.

Další informace o textových displejích jsou uvedeny v *uživatelském manuálu k textovému displeji (TD) SIMATIC* na CD k STEP 7–Micro/WIN.



Obr. 1-3 Textový displej (TD 200 a TD 200C)



TD 200

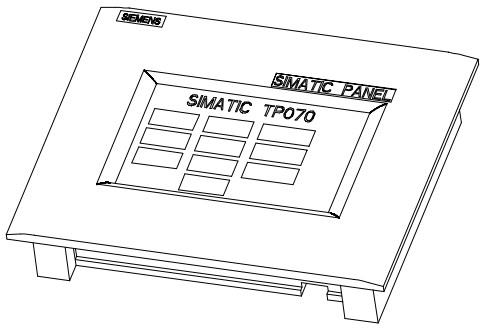
Průvodce konfigurací TD 200 ve STEP 7–MicroWIN pomáhá konfigurovat zprávy TD 200 rychle a snadno. Chcete–li spustit průvodce TD 200, zvolte příkaz menu **Nástroje > Průvodce TD 200**.

Dotykové panely TP070 a TP170 micro

TP070 a TP170 micro jsou dotykem ovládané zobrazovací jednotky, které lze připojit k S7-200. Tato jednotka umožňuje vytvořit vlastní uživatelské prostředí.

Tyto jednotky mohou zobrazovat uživatelskou grafiku, sloupcové grafy, aplikační proměnné, uživatelská tlačítka a další prvky pomocí uživatelsky příjemného dotykového panelu.

Doplňkový software pro programování dotykových panelů TP070 a TP170 micro najdete v příloze E.



Obr. 1-4 Dotykový panel

Úvod do programování

2

STEP 7–Micro/WIN umožňuje snadno naprogramovat systém S7-200. Pouze v několika krátkých krocích se můžete pomocí příkladu naučit, jak připojit, naprogramovat a provozovat S7-200.

Pro tento příklad potřebujete pouze PPI Multi-Master kabel, CPU S7-200 a počítač se softwarem STEP 7–Micro/WIN.

Přehled kapitoly

Připojení CPU S7-200 8
Vytvoření vzorového programu 10
Download vzorového programu 14
Uvedení S7-200 do režimu RUN 14

Připojení CPU S7-200

Připojení S7-200 je snadné. Pro tento příklad je potřeba pouze připojit napájení k CPU S7-200 a pak připojit komunikační kabel mezi programovací zařízení a S7-200.

Připojení napájení k S7-200 CPU

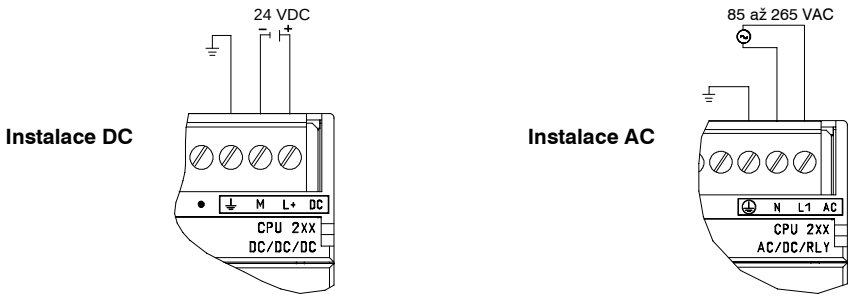
Prvním krokem je připojení S7-200 k napájecímu zdroji. Obrázek 2-1 ukazuje zapojení pro stejnosměrný nebo střídavý model S7-200.

Před montáží nebo demontáží elektrického zařízení zajistěte, aby bylo odpojeno napájení. Vždy dodržujte příslušné bezpečnostní předpisy. Zajistěte, aby bylo napájení k S7-200 vypnuto před pokusem instalovat nebo demontovat jednotku S7-200.

Varování

Pokusy instalovat nebo zapojovat S7-200 nebo připojená zařízení se zapnutým napájením by mohly způsobit úraz elektrickým proudem nebo chybný chod zařízení. Nevypnutí napájení systému S7-200 a připojených zařízení během instalace nebo demontáže by mohlo mít za následek smrt nebo vážné zranění osob a poškození zařízení.

Vždy dodržujte příslušné bezpečnostní předpisy a zajistěte, aby bylo napájení S7-200 vypnuto před pokusem instalovat nebo demontovat jednotku S7-200 nebo připojená zařízení.

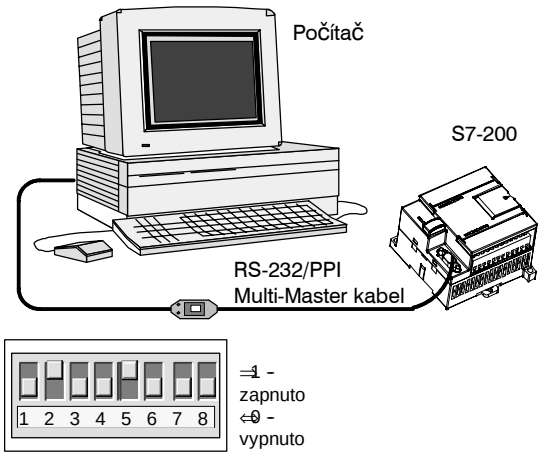


Obr. 2-1 Připojení napájení S7-200

Připojení RS-232/PPI Multi-Master kabelem

Na obrázku 2-2 je RS-232/PPI Multi-Master kabel, kterým se připojí S7-200 k počítači. Jak kabel připojit:

1. Připojte konektor RS-232 (označený ”PC”) RS-232/PPI Multi-Master kabelu ke komunikačnímu portu počítače. (Pro tento příklad jej připojte na COM 1.)
2. Připojte konektor RS-485 (označený ”PPI”) RS-232/PPI Multi-Master kabelu k portu 0 nebo k portu 1 jednotky S7-200.
3. Zkontrolujte, že jsou DIP přepínače RS-232/PPI Multi-Master kabelu nastaveny tak, jak ukazuje obrázek 2-2.



Obr. 2-2 Připojení RS-232/PPI Multi-Master kabelem

Tip

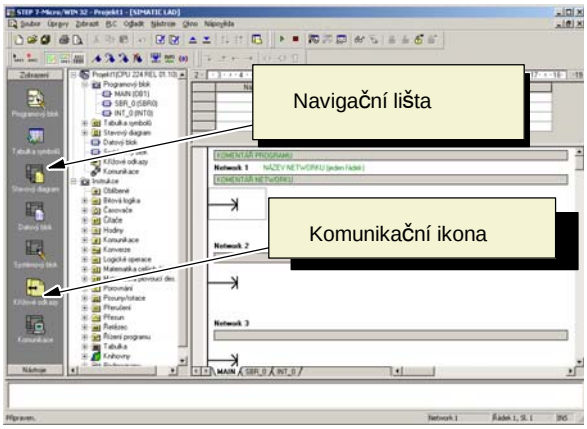
Příklady v tomto manuálu používají RS-232/PPI Multi-Master kabel. RS-232/PPI Multi-Master kabel nahrazuje dřívější PC/PPI kabel. Nabízíme také USB/PPI Multi-Master kabel. Objednací čísla – viz příloha NO TAG.

Spuštění STEP 7–Micro/WIN

Kliknutím na ikonu STEP 7–Micro/WIN otevřete nový projekt, který je zobrazen na obrázku 2-3

Všimněte si navigační lišty. K otevření prvků projektu STEP 7–Micro/WIN můžete použít ikony na navigační liště.

Klikněte na komunikační ikonu v navigační liště, zobrazí se tak komunikační dialogové okno. Pomocí tohoto dialogového okna můžete nastavit komunikaci pro STEP 7–Micro/WIN.



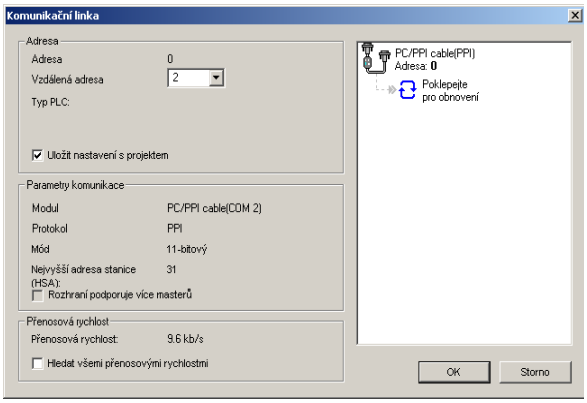
Obr. 2-3 Nový projekt STEP 7–Micro/WIN

Ověření parametrů komunikace pro STEP 7–Micro/WIN

Vzorový projekt používá implicitní nastavení pro STEP 7–Micro/WIN a RS-232/PPI Multi-Master kabel. Ověření těchto nastavení:

1. Ověřte, zda je v komunikačním dialogovém okně nastavena adresa PC/PPI kabelu na 0.
2. Ověřte, zda je rozhraní pro parametry sítě nastaveno pro PC/PPI kabel (COM1).
3. Ověřte, zda je přenosová rychlost nastavena na 9,6 kb/s.

Pokud potřebujete změnit nastavení parametrů komunikace, přejděte ke kapitole 7.



Obr. 2-4 Ověření komunikačních parametrů

Navázání komunikace s S7-200

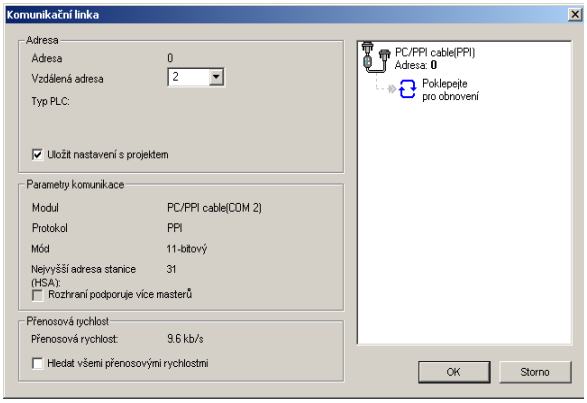
Pro připojení S7-200 použijte komunikační dialogové okno:

1. V komunikačním dialogovém okně klikněte dvakrát na obnovovací ikonu.

   STEP 7–Micro/WIN vyhledá stanici S7-200 a zobrazí ikonu CPU pro připojenou stanici S7-200.
2. Zvolte S7-200 a klikněte na OK.

Pokud STEP 7–Micro/WIN nenajde jednotku S7-200 CPU, zkontrolujte nastavení parametrů komunikace a opakujte postup.

Po navázání komunikace s S7-200 je možné vytvořit a spustit download vzorového programu.



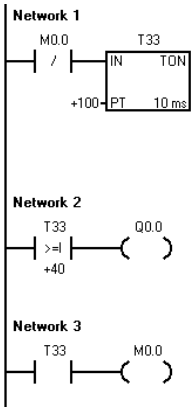
Obr. 2-5 Navázání komunikace s S7-200

Vytvoření vzorového programu

Tento příklad řídicího programu pomáhá pochopit, jak snadné je používat STEP 7–Micro/WIN. Tento program používá šest instrukcí ve třech networcích k vytvoření velmi jednoduchého samospouštěcího časovače, který se sám nuluje.

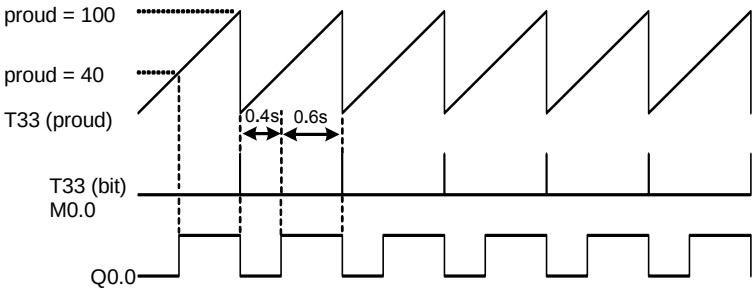
Pro tento příklad použijte editor LADDER (LAD) k zadání instrukcí k programu. Následující příklad zobrazuje úplný program v LAD i v STL. Komentář networku v STL programu vysvětluje logiku pro každý network. Diagram časování zobrazuje chod programu.

Příklad: Vzorový program pro zahájení práce se STEP 7–Micro/WIN




```
Network 1     //10ms časovač T33 časově odpojuje poté, co
              //(100 x 10 ms = 1 s) M0.0 pulz je
              // příliš rychlý k monitorování stavovým zobrazením.
LDN     M0.0
TON     T33, +100
Network 2     //Porovnání je pravdivé při
              //rychlosti, která je vidět na
              //stavovém zobrazení. Zapněte Q0.0 po
              //(40 x 10 ms = 0,4 s), pro
              // časový průběh 40 % vypnuto / 60 % zapnuto.
LDW>=   T33, +40
=       Q0.0
Network 3     //Pulz T33 (bit) je příliš rychlý k monitorování na
              //stavovém zobrazení. Reset časovače přes
              //M0.0 po době (100 x 10 ms  = 1 s).
LD      T33
=       M0.0
```

Časový diagram



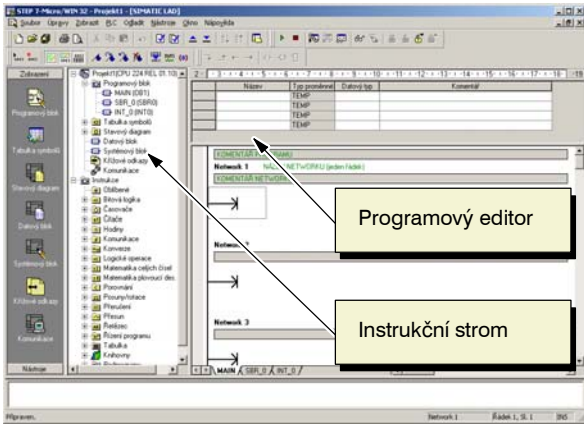
Otevření programového editoru

Kliknutím na ikonu programového bloku otevřete programový editor.
Viz obrázek 2-6.

Všimněte si instrukčního stromu a programového editoru. Instrukční strom používá k vložení LAD instrukcí do networků programového editoru tažení a puštění instrukcí z instrukčního stromu.

Ikony nástrojové lišty poskytují zkrácené povely k příkazům menu.

Po vložení a uložení programu můžete provést download programu do S7-200.

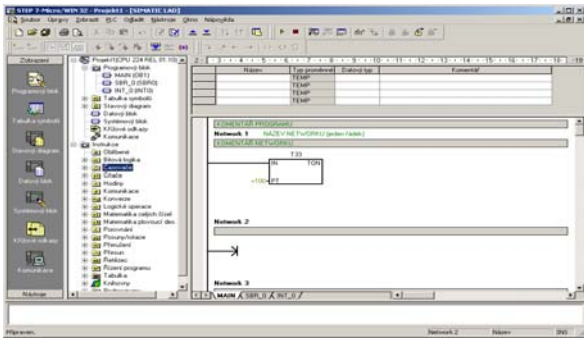


Obr. 2-6 Okno STEP 7–Micro/WIN

Zadání networku 1: spuštění časovače

Když je M0.0 vypnut (0), kontakt zapne a dodá povel ke spuštění časovače. Zadání kontaktu pro M0.0:

1. Pro zobrazení instrukcí bitové logiky dvakrát klikněte na ikonu bitová logika nebo na znaménko plus (+).
2. Zvolte v klidu zavřený kontakt.
3. Držte stisknuté levé tlačítko myši a přesuňte kontakt na první network.
4. Klikněte na "???" nad kontaktem a zadejte tuto adresu: M0.0
5. Stiskněte Enter pro uložení hodnoty.



Obr. 2-7 Network 1

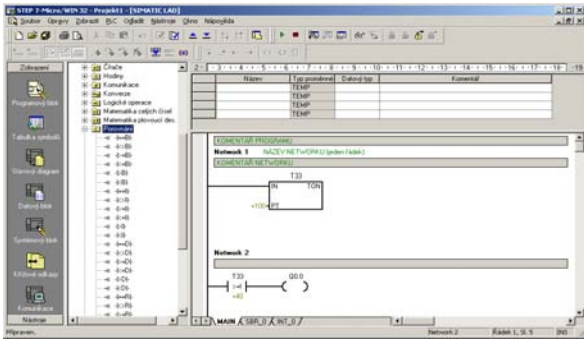
Zadání instrukce časovače T33:

1. Pro zobrazení instrukcí dvakrát klikněte na ikonu časovačů.
2. Zvolte TON (časovač zpožděného zapnutí).
3. Držte stisknuté levé tlačítko myši a přesuňte časovač na první network.
4. Klikněte na "???" nad oknem časovače a zadejte číslo časovače: T33
5. Stiskněte Enter pro vložení čísla časovače a posuňte zobrazení na parametr nastaveného času (PT).
6. Zadejte tuto hodnotu pro nastavený čas: 100
7. Stiskněte Enter pro uložení hodnoty.

Zadání networku 2: zapnutí výstupu

Pokud je hodnota časovače T33 vyšší nebo rovna 40 (40krát 10 millisekund nebo 0,4 sekundy), kontakt dá povel pro sepnutí výstupu Q0.0 jednotky S7-200.Instrukce porovnání:

1. Dvakrát klikněte na ikonu k zobrazení porovnávacích instrukcí. Zvolte instrukci >=I (větší než nebo roven integer).
2. Držte stisknuté levé tlačítko myši a přesuňte instrukci na druhý network.
3. Klikněte na "???" nad kontaktem a zadejte adresu pro hodnotu časovače: T33
4. Pro vložení adresy stiskněte Enter; posuňte zobrazení na druhou hodnotu, která má být srovnána s hodnotou časovače.
5. Zadejte hodnotu srovnání s hodnotou časovače: 40
6. Stiskněte Enter pro uložení hodnoty.



Obr. 2-8 Network 2

Zadání instrukce pro zapnutí výstupu Q0.0:

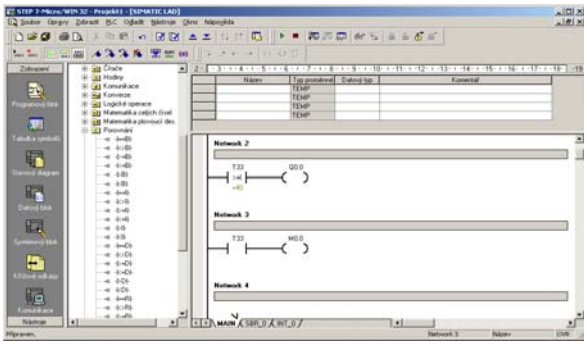
1. Pro zobrazení instrukcí bitové logiky dvakrát klikněte na ikonu bitové logiky a zvolte výstup.
2. Držte stisknuté levé tlačítko myši a přesuňte výstup na druhý network.
3. Klikněte na "???" nad cívkou a zadejte tuto adresu: Q0.0.
4. Stiskněte Enter pro uložení hodnoty.

Zadání networku 3: Nulování časovače

Jakmile časovač dosáhne přednastavenou hodnotu (100) a nastaví bit časovače, sepne se kontakt pro T33. Signál z tohoto kontaktu nastaví paměťové místo M0.0. Protože je časovač normálně zapnut při zavřeném kontaktu M0.0, vynuluje časovač změnu stavu M0.0 z vypnuto (0) na zapnuto (1).

Vložení kontaktu pro bit časovače T33:

1. Z instrukcí bitové logiky zvolte kontakt v klidu otevřen
2. Držte stisknuté levé tlačítko myši a přesuňte kontakt na třetí network.
3. Klikněte na ”???” nad kontaktem a zadejte adresu bitu časovače: T33.
4. Stiskněte Enter pro uložení hodnoty.



Obr. 2-9 Network 3

Vložení kontaktu pro nastavení M0.0:

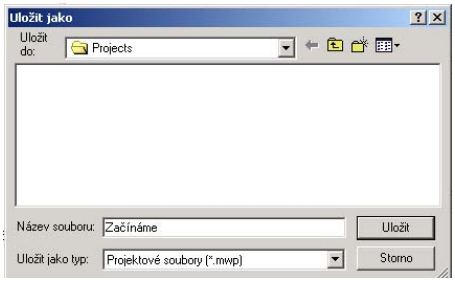
1. Zvolte výstupní kontakt z instrukcí bitové logiky.
2. Držte stisknuté levé tlačítko myši a přesuňte na třetí network.
3. Klikněte dvakrát na ”???” nad cívkou a zadejte tuto adresu: M0.0
4. Stiskněte Enter pro uložení hodnoty.

Uložení vzorového projektu

Zadáním instrukcí tří networků jste ukončili zadávání programu. Uložením programu vytvoříte projekt, který obsahuje typ jednotky S7-200 CPU a další parametry. Uložení projektu:

1. Z menu zvolte příkaz **Soubor > Uložit jako**.
2. Zadejte název projektu v dialogovém okně “Uložit jako”.
3. Kliknutím na OK projekt uložíte.

Po uložení projektu můžete provést download programu do S7-200.



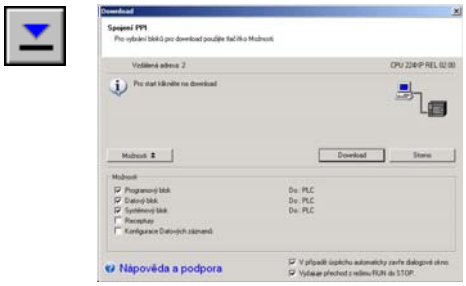
Obr. 2-10 Uložení vzorového programu

Download vzorového programu

Tip
Každý projekt STEP 7 – Micro/WIN je svázán s typem CPU (CPU 221, CPU 222, CPU 224, CPU 224XP nebo CPU 226). Pokud typ projektu nesouhlasí s CPU, STEP 7 – Micro/WIN oznámí neshodu a žádá o zásah. Pokud se tak stane, zvolte ”Pokračování downloadu”.

1. Pro provedení downloadu programu klikněte na ikonu Download na nástrojové liště nebo zvolte příkaz menu **Soubor > Download**. Viz obrázek 2-11.
2. Kliknutím na OK provedete download prvků programu do S7-200.

Pokud je S7-200 v režimu RUN, dialogové okno vás vyzve k přepnutí S7-200 do režimu STOP. Kliknutím na Ano přepnete S7-200 do režimu STOP.



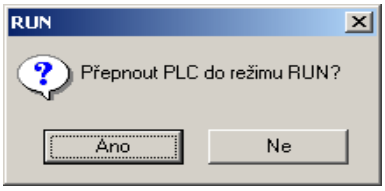
Obr. 2-11 Download programu

Uvedení S7-200 do režimu RUN

Aby STEP 7 – Micro/WIN uvedl S7-200 do režimu RUN, musí být přepínač režimu S7-200 nastaven na TERM nebo RUN. Jakmile uvedete S7-200 do režimu RUN, začne S7-200 vykonávat program:

1. Klikněte na ikonu RUN na nástrojové liště nebo zvolte příkaz menu **PLC > RUN**.
2. Kliknutím na OK změníte provozní režim S7-200.

Když přejde S7-200 do režimu RUN, výstupní LED se bude pro Q0.0 zapínat a vypínat podle běhu programu v S7-200.



Obr. 2-12 Uvedení S7-200 do režimu RUN

Blahopřejeme! Právě jste dokončili váš první program S7-200.

Program můžete sledovat volbou příkazu menu **Odladit > Stav programu**. STEP 7 – Micro/WIN zobrazí hodnoty pro instrukce. K zastavení programu uveďte S7-200 do režimu STOP kliknutím na ikonu STOP nebo volbou příkazu menu **PLC > STOP**.

Instalace S7-200

3

Zařízení S7-200 je navrženo tak, aby byla jeho instalace snadná. K připevnění modulů na panel můžete použít montážní otvory nebo vestavěné klipsy k upevnění modulů na standardní (DIN) lištu. Malá velikost S7-200 umožňuje efektivně využít prostor.

Tato kapitola poskytuje návod pro instalaci a zapojení systému S7-200.

Přehled kapitoly

Návod pro instalaci zařízení S7-200 16
Montáž a demontáž modulů S7-200 17
Návod pro uzemnění a elektrické zapojení 20

Návod pro instalaci zařízení S7-200

Zařízení S7-200 umístěte na panel nebo na standardní lištu a namontujte jej horizontálně nebo vertikálně.

Chraňte zařízení S7-200 před teplem, vysokým napětím a elektrickým rušením.

Pro umístění zařízení v systému platí všeobecné pravidlo: vždy je nutné oddělit zařízení, která vytvářejí vysoké napětí nebo velké elektrické rušení, od nízkonapěťových a logických zařízení, jako je S7-200.

Při instalaci systému S7-200 do panelu vezměte v úvahu zařízení vytvářející teplo; elektronická zařízení umístěte do chladnějších částí rozvaděče. Provoz jakéhokoli elektronického zařízení v prostředí s vysokou teplotou zkrátí dobu bezporuchového provozu.

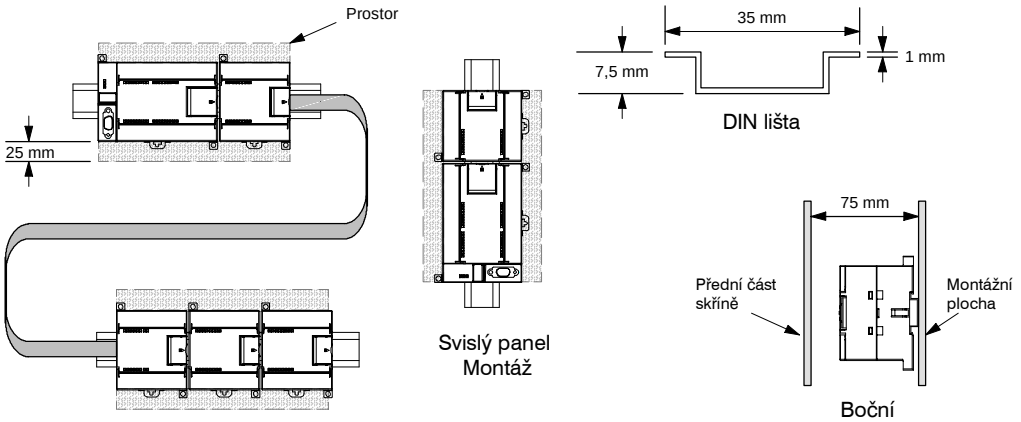
Zvažte rovněž položení kabeláže pro zařízení umístěné v panelu. Neumísťujte nízkonapěťové signální vodiče a komunikační kabely do stejné lišty s napájecími kabely střídavého napětí a vysokonapěťovými vodiči stejnosměrného napětí, které jsou rychle spínány.

Zajistěte přiměřený prostor pro chlazení a kabeláž

Zařízení S7-200 jsou určena pro přirozené konvekční chlazení. Kvůli řádnému chlazení musíte zajistit prostor asi 25 mm nad a pod zařízeními. Navíc ponechte montážní hloubku alespoň 75 mm.

Tip
Při vertikální montáži je maximální povolená teplota okolí snížena o 10 °C. CPU S7–200 namontujte pod rozšiřovací moduly.

Při plánování umístění systému S7-200 ponechejte dostatek prostoru pro kabeláž a konektory komunikačních kabelů. Pro flexibilní konfiguraci rozmístění systému S7-200 použijte rozšiřovací kabel.



Horizontální montáž na DIN lištu s volitelným rozšiřovacím kabelem (jeden kabel na systém)

Obr. 3-1 Montážní metody, orientace a prostor

Energetická bilance

Všechny CPU S7-200 mají vnitřní napájecí zdroje, jimiž jsou napájeny CPU, rozšiřovací moduly a další uživatelská zařízení pro napětí 24 V DC.

S7-200 poskytuje 5 V stejnosměrné napětí, potřebné pro rozšíření systému. Věnujte pozornost konfiguraci systému, abyste zajistili, že CPU může dodávat 5V napětí požadované zvolenými rozšiřovacími moduly. Pokud konfigurace vyžaduje větší proud, než může CPU dodávat, musíte modul demontovat nebo zvolit CPU s vyšší proudovou kapacitou. Informace o 5 V DC energetické bilanci, dodávané jednotkou S7-200, a o požadavcích na napětí 5 VDC u rozšiřovacích modulů jsou uvedeny v příloze A. Jako průvodce pro stanovení, jaký výkon (nebo proud) může CPU dodávat pro vaši konfiguraci, použijte přílohu B.

Všechny jednotky S7-200 také poskytují napájení snímačů 24 VDC, které může dodávat 24 VDC pro vstupy, pro napájení relé u rozšiřovacích modulů nebo pro jiná zařízení. Pokud požadavky na napájení překročí výkon zdroje, musíte přidat k systému externí zdroj napětí 24 VDC. V příloze A jsou uvedeny informace pro energetickou bilanci zdroje pro snímače 24 VDC pro konkrétní jednotku S7-200.

Pokud požadujete externí napájecí zdroj 24 VDC, zajistěte, aby nebyl zdroj paralelně propojen se zdrojem pro snímače S7-200 CPU. Pro zvýšení odolnosti proti elektrickému rušení se doporučuje, aby byly společné vodiče (M) různých napájecích zdrojů propojeny.



Varování

Připojení externího napájecího zdroje 24 VDC paralelně se zdrojem pro snímače S7-200 24 VDC může vést ke konfliktu mezi těmito dvěma zdroji, protože každý se snaží nastavit svou vlastní úroveň výstupního napětí.

Následkem toho může být zkrácena životnost zařízení nebo může dojít k okamžité poruše jednoho nebo obou napájecích zdrojů a k následnému nepředvídatelnému chování řídicího systému. Nepředvídatelné chování by mohlo mít za následek smrt nebo vážné poranění osob a poškození zařízení.

Stejnosměrný zdroj pro snímače S7-200 a externí zdroj by měly dodávat napětí do různých bodů.

Montáž a demontáž modulů S7-200

Zařízení S7-200 může být snadno nainstalováno na standardní DIN lištu nebo na panel.

Požadavky

Před montáží nebo demontáží elektrického zařízení zajistěte, aby bylo odpojeno napájení. Rovněž zajistěte, aby bylo napájení zařízení vypnuto.



Varování

Pokusy instalovat nebo demontovat S7-200 nebo připojené zařízení se zapnutým napájením by mohly způsobit úraz elektrickým proudem nebo chybný chod zařízení.

Nevypnutí napájení systému S7-200 a připojených zařízení během instalace nebo demontáže by mohlo mít za následek smrt nebo vážné zranění osob a poškození zařízení.

Vždy dodržujte příslušné bezpečnostní předpisy a zajistěte, aby bylo vypnuto napájení S7-200 před pokusem instalovat nebo demontovat jednotky CPU nebo připojená zařízení.

Vždy dbejte na to, abyste při jakékoli výměně nebo instalaci zařízení S7-200 použili správný modul nebo odpovídající zařízení.



Varování

Pokud nainstalujete nesprávný modul, může program v S7-200 pracovat nepředvídatelně.

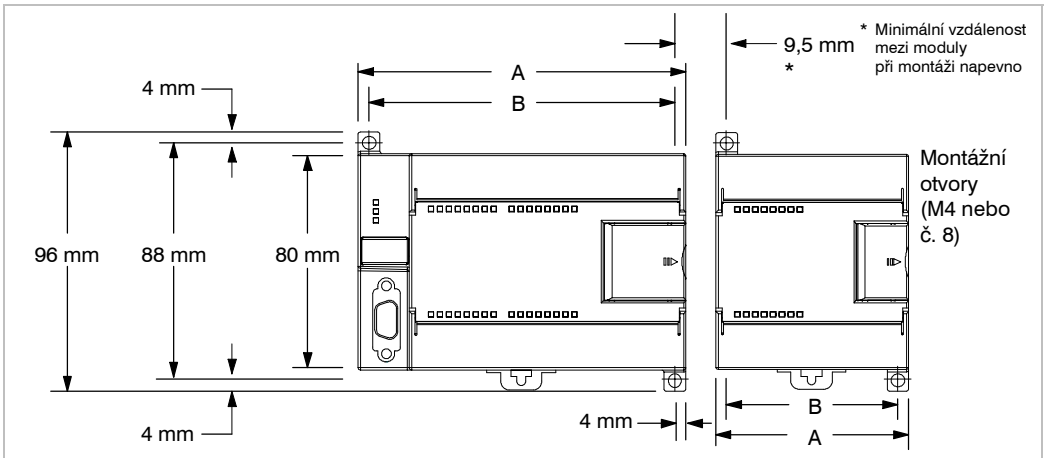
Pokud nenahradíte zařízení S7-200 stejným modelem, se stejnou orientací nebo ve stejném pořadí, mohlo by to mít za následek smrt nebo vážné zranění osob a poškození zařízení.

Nahraďte zařízení S7-200 stejným modelem a zajistěte jeho správnou orientaci a polohu.

Montážní rozměry

Jednotky S7-200 a rozšiřovací moduly mají montážní otvory, které usnadňují instalaci na panel. Montážní rozměry jsou uvedeny v tabulce 3-1.

Tabulka 3-1 Montážní rozměry



S7-200 modul		Šířka A	Šířka B
CPU 221 a CPU 222		90 mm	82 mm
CPU 224		120,5 mm	112,5 mm
CPU 224XP		140 mm	132 mm
CPU 226		196 mm	188 mm
Rozšiřovací moduly:	4 a 8bodový DC a reléový vst/výst (8I, 4Q, 8Q, 4I/4Q) a analogový výstupní (2 AQ)	46 mm	38 mm
Rozšiřovací moduly: (4AI, 4AI/1AQ), modemový	16bodový digitální vst/výst (16I, 8I/8Q), analogový vst/výst RTD, termočlánek, PROFIBUS, Ethernet, Internet, AS-rozhraní, 8bodový AC (8I a 8Q), polohovací a	71,2 mm	63,2 mm
Rozšiřovací moduly:	32bodový digitální vst/výst (16I/16Q)	137,3 mm	129,3 mm

Montáž CPU nebo rozšiřovacího modulu

Montáž S7-200 je snadná! Řiďte se těmito kroky.

Montáž na panel

1. Umístěte a vyvrtejte montážní otvory a vyřízněte do nich závit (M4 nebo podle americké normy č. 8) podle rozměrů uvedených v tabulce 3-1.
2. Připevněte modul(y) na panel pomocí příslušných šroubů.
3. Pokud používáte rozšiřovací modul, připojte plochý kabel rozšiřovacího modulu ke konektoru rozšiřovacího portu pod přístupovým krytem.

Montáž na DIN lištu

1. Lištu připevňujte k montážnímu panelu vždy po 75 mm.
2. Otevřete zajišťovací uzávěr DIN (ve spodní části modulu) a nasaďte modul zadní částí na DIN lištu.
3. Pokud používáte rozšiřovací modul, připojte plochý kabel rozšiřovacího modulu ke konektoru rozšiřovacího portu pod přístupovým krytem.
4. Otočte modul dolů k DIN liště a uzavřete uzávěr. Opatrně zkontrolujte, zda uzávěr pevně přichytil modul na lištu. Abyste zabránili poškození modulu, netlačte přímo na přední panel modulu, ale tlačte na úchytku montážního otvoru.

Tip

Použití zarážek DIN lišty by mohlo být užitečné, pokud je zařízení S7-200 v prostředí s velkými vibracemi nebo pokud bylo zařízení S7-200 namontováno vertikálně.

Pokud je Váš systém v prostředí s velkými vibracemi, bude před nimi S7-200 při montáži na panel poskytovat vyšší úroveň ochrany.

Demontáž CPU nebo rozšiřovacího modulu

Při demontáži S7-200 CPU nebo rozšiřovacího modulu dodržujte tyto kroky:

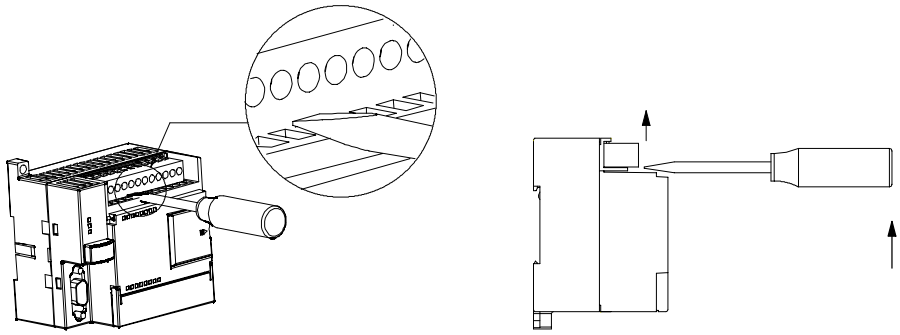
1. Odpojte napájení od S7-200.
2. Odpojte všechny vodiče a kabely, které jsou k modulu připojeny. Většina jednotek S7-200 CPU a rozšiřovacích modulů má odnímatelné konektory, které tuto práci usnadňují.
3. Pokud máte rozšiřovací moduly připojeny k jednotce, kterou demontujete, otevřete přístupový kryt a odpojte plochý kabel rozšiřovacího modulu od sousedních modulů.
4. Odšroubujte montážní šrouby nebo otevřete DIN uzávěr.
5. Vyjměte modul.

Demontáž a montáž svorkovnice

Většina modulů S7-200 má odnímatelné svorkovnice, které usnadňují montáž a výměnu modulu. V příloze A zjistíte, zda má váš modul S7-200 odnímatelné svorkovnice. Pro moduly, které nemají odnímatelné svorkovnice, si můžete objednat doplňkový konektor. Objednací čísla – viz přílohu NO TAG.

Vyjmutí svorkovnice

1. Otevřete kryt svorkovnice, abyste k ní měli přístup.
2. Vsuňte malý šroubovák do zářezu uprostřed svorkovnice.
3. Použijte šroubovák jako páku a vysuňte svorkovnici z pouzdra S7-200. Viz obrázek 3-2.



Obr. 3-2 Vyjmutí svorkovnice

Nasazení svorkovnice

1. Otevřete kryt svorkovnice.
2. Srovnejte svorkovnici s kolíky na jednotce a srovnejte okraj svorkovnice v rámu patky svorkovnice.
3. Pevně zatlačte dolů a otáčejte svorkovnici, dokud nezapadne na své místo. Opatrně zkontrolujte, zda je svorkovnice řádně srovnána a zcela zasunuta.

Návod pro uzemnění a elektrické zapojení

Správné uzemnění a zapojení všech elektrických zařízení je důležité k zajištění optimálního provozu systému a pro další ochranu vaší aplikace a jednotky S7-200 před elektrickým rušením.

Požadavky

Před uzemněním nebo montáží kabeláže elektrického zařízení zajistěte, aby bylo napájení zařízení vypnuto. Rovněž zajistěte, aby bylo napájení zařízení vypnuto.

Rovněž zajistěte, aby bylo vypnuto napájení jakéhokoli připojeného zařízení. Při zapojování S7-200 a připojených zařízení dodržujte veškeré předpisy platné pro elektroinstalaci. U místních úřadů zjistěte, které předpisy a normy jsou platné pro konkrétní účel použití.



Varování

Pokusy instalovat nebo zapojovat S7-200 nebo připojená zařízení se zapnutým napájením by mohly způsobit úraz elektrickým proudem nebo chybný chod zařízení. Nevypnutí napájení systému S7-200 a připojených zařízení během instalace nebo demontáže by mohlo mít za následek smrt nebo vážné zranění osob a poškození zařízení.

Vždy dodržujte příslušné bezpečnostní předpisy a zajistěte, aby bylo napájení S7-200 vypnuto před pokusem instalovat nebo demontovat jednotku S7-200 nebo připojená zařízení.

Při navrhování uzemnění a zapojení systému S7-200 mějte vždy na paměti bezpečnost. Elektronická řídicí zařízení jako je S7-200 mohou selhat a způsobit nepředvídatelné chování zařízení, které je řízeno nebo monitorováno. Proto byste měli zařadit ochranné obvody nezávislé na S7-200 jako ochranu před zraněním osob nebo poškozením zařízení.



Varování

Řídicí zařízení mohou selhat a dostat se do nebezpečného stavu, který může mít za následek nepředvídatelné chování řízeného zařízení. Toto nepředvídatelné chování by mohlo mít za následek smrt nebo vážné zranění osob a poškození zařízení.

Použijte funkci nouzového zastavení, elektromechanické spínače nebo jiné doplňující ochranné obvody nezávislé na S7-200.

Návod pro izolaci

Limity střídavého napájecího napětí S7-200 a napětí na vstupech/výstupech jsou počítány na 1500 VAC. Tyto izolační limity byly vyzkoušeny a odsouhlaseny jako bezpečné oddělení střídavého napětí od nízkonapěťových obvodů.

Všechny nízkonapěťové obvody připojené k S7-200, jako je napětí 24 V, musí být napájeny ze schváleného zdroje, který poskytuje bezpečnou izolaci střídavého napětí a jiných vysokonapěťových okruhů. Tyto zdroje obsahují dvojitou ochranu, která je definována v mezinárodních normách bezpečnosti elektrických zapojení, a mají výstupy, které spadají do třídy SELV, PELV, třídy 2 nebo omezené napájení podle různých norem.



Varování

Použití neoddělených nebo nesprávně oddělených zdrojů pro napájení nízkonapěťových obvodů z vedení střídavého napětí může mít za následek riziková napětí na obvodech, které mají být bezpečné na dotek, jako jsou komunikační obvody a nízkonapěťové rozvody snímačů.

Toto nepředvídatelné vysoké napětí by mohlo mít za následek smrt nebo vážné zranění osob a poškození zařízení.

Používejte pouze měniče z vysokého napětí na nízké, které jsou schváleny jako zdroje pro obvody omezeného napětí bezpečného na dotek.

Návod pro uzemnění S7-200

Nejlepší způsob, jak uzemnit zařízení, je zajistit, aby všechny společné a uzemňovací vodiče systému S7-200 a připojených zařízení byly uzemněny v jednom bodě. Ten by měl být připojen přímo k uzemnění vašeho systému.

Ke zvýšení ochrany proti elektrickému rušení se doporučuje, aby všechny společné vratné vodiče stejnosměrného napětí byly připojeny k jedinému bodu uzemnění. Společný vodič (M) napájení snímačů 24 VDC připojte k uzemnění.

Všechny uzemňovací vodiče by měly být co nejkratší a měly by být použity vodiče s vyšším průřezem jako 2 mm^2 (14 AWG).

Při umísťování uzemnění nezapomeňte vzít v úvahu požadavky na bezpečné uzemnění a řádnou funkci ochranných jisticích zařízení.

Návod pro zapojení S7-200

Při zapojování systému namontujte jeden odpojovací vypínač, který současně odpojí napájení od jednotky S7-200 a od všech vstupních a výstupních obvodů. Pro omezení poruchových proudů na napájecích vodičích nainstalujte nadproudovou ochranu, jako je pojistka nebo stykačový jistič. Další ochranu můžete přidat nainstalováním pojistky nebo jiného omezovače proudu do každého výstupního obvodu.

Namontujte příslušná zařízení na omezení proudových nárazů pro všechny okruhy, které by mohly být vystaveny bleskovým výbojům.

Neumísťujte nízkonapěťové signálové vodiče a komunikační kabely do stejné lišty s napájecími kabely střídavého napětí a vysokonapěťovými vodiči stejnosměrného napětí, které jsou rychle spínány. Vždy pokládejte vodiče v párech s nulovým nebo společným vodičem, spárovaným s živým nebo signálovým vodičem.

Použijte co nejkratší vodiče, abyste zajistili, že jsou řádně dimenzovány pro přenos požadovaného proudu. Ke konektoru může být připojen vodič o průřezu od 2 mm^2 do 0,3 mm^2 (14 AWG až 22 AWG). Kvůli optimální ochraně proti elektrickému rušení použijte stíněné vodiče. Standardně uzemněné stínění dává u S7-200 nejlepší výsledky.

Při zapojování vstupních obvodů, které jsou napájeny externím napájecím zdrojem, nainstalujte do tohoto obvodu nadproudovou ochranu. Externí ochrana není nutná u obvodů, které jsou napájeny zdrojem pro snímače 24 VDC z jednotky S7-200, protože zdroj pro snímače už proudové omezení obsahuje.

Většina modulů S7-200 má odnímatelné svorkovnice kvůli snadnějšímu zapojování. (V příloze A zjistíte, zda má váš modul odnímatelné svorkovnice). Studeným spojům předejdete tak, že zajistíte, aby byl konektor pevně usazen a aby k němu byl vodič pevně připojen. Neutahujte šrouby příliš, abyste nepoškodili konektor. Maximální utahovací moment pro šroub konektoru je 0,56 N-m (5 palců-libra).

Pro zabránění nežádoucích proudů v instalaci má S7-200 na určitých bodech napěťová oddělení. Limity těchto oddělení byste měli brát v úvahu při plánování zapojení vašeho systému. V příloze A je uvedena hodnota a umístění oddělení. Limity s hodnotou menší než 1500 VAC nesmí být brány za bezpečnostní limity.



Tip

Pro komunikační síť je maximální délka komunikačního kabelu 50 m bez použití zesilovače. Komunikační port na S7-200 není napěťově oddělen. Další informace jsou v kapitole 7.

Návod pro indukční zátěže

Indukční zátěže byste měli vybavit ochrannými obvody pro omezení špiček napětí při vypnutí výstupu. Ochranné obvody chrání výstupy před předčasnou poruchou, způsobenou velkými indukčními vypínacími proudy. Navíc omezují elektrické rušení, tvořené při vypnutí indukční zátěže.



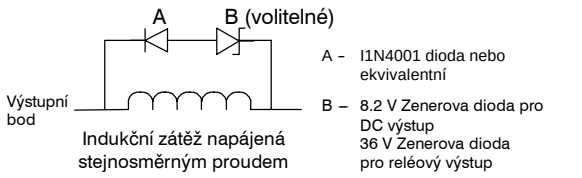
Tip

Účinnost ochranného obvodu závisí na použitém zařízení a musíte jej ověřit pro konkrétní použití. Vždy zajistěte, aby byly všechny komponenty použité v ochranném obvodu vypočítány pro použití v příslušném zařízení.

Stejnosměrné a reléové výstupy, které napájejí induktivní zátěž stejnosměrným proudem

Stejnosměrné výstupy mají vnitřní ochranu, která je dostačující pro většinu aplikací. Protože reléové výstupy mohou být použity pro stejnosměrný i střídavý proud, není u nich vnitřní ochrana implementována.

Na obrázku 3-3 je vzor přepěťové ochrany pro induktivní zátěž napájenou stejnosměrným proudem. U většiny aplikací je vhodné použít diodu (A) přes indukční zátěž, ale pokud vyžaduje vaše aplikace rychlejší vypínací časy, doporučuje se použít Zenerovu diodu (B). Zenerovu diodu správně dimenzujte pro proud ve vašem výstupním obvodu.

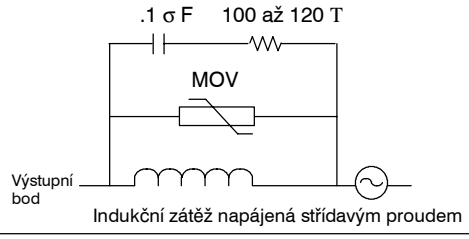


Obr. 3-3 Potlačovací obvod pro zátěž stejnosměrným proudem

Střídavé a reléové výstupy, které napájejí induktivní zátěž střídavým proudem.

Střídavé výstupy mají vnitřní ochranu, která je dostačující pro většinu aplikací. Protože reléové výstupy mohou být použity pro stejnosměrný i střídavý proud, není u nich vnitřní ochrana implementována.

Na obrázku 3-4 je vzor přepěťové ochrany pro induktivní zátěž napájenou střídavým proudem. Jestliže použijete reléový nebo střídavý výstup pro spínání zátěží 115 V/230 V AC, instalujte network s odporem/kondenzátorem přes AC zátěž, jak ukazuje tento obrázek. Pro omezení špičkového napětí můžete také použít varistor z oxidu kovu (MOV). Zajistěte, aby pracovní napětí MOV bylo alespoň o 20 % vyšší než nominální napětí vedení.



Obr. 3-4 Potlačovací obvod pro zátěž střídavým proudem

Poznámka

Jestliže se pro spínání indukčních zátěží 230 V AC použijí reléové rozšiřovací moduly, musí být přes AC zátěž instalován externí obvod pro potlačení elektrického rušení s odporem/kondenzátorem, jak je ukázáno na obrázku 3-4.

Návod pro žárovkové zátěže

Žárovkové zátěže poškozují kvůli vysokému nárazovému proudu při zapnutí reléové kontakty. Tento nárazový proud je nominálně 10 až 15násobek ustáleného proudu wolframové žárovky. Pro žárovkové zátěže, které prodělají velké množství zapnutí během životnosti aplikace, se doporučuje vyměnitelné přechodové relé nebo omezovač nárazu.

Základní principy

4

Základní funkcí programovatelného automatu S7-200 je monitorovat vstupy a na základě řídicího programu zapínat nebo vypínat výstupní zařízení. Tato kapitola vysvětluje principy používané pro provádění programu, různé typy použitých pamětí a to, jak je program v dané paměti uchováván.

Přehled kapitoly

Jak S7-200 vykonává řídicí program 24
Přístup k datům 26
Ukládání a obnova dat 36
Výběr pracovního režimu CPU S7-200 40
Použití Průvodníka S7-200 40
Vlastnosti programovatelného automatu S7-200 41

Jak S7-200 vykonává řídicí program

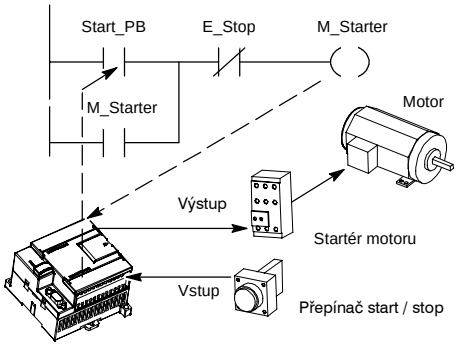
Programovatelný automat cyklicky provádí uložený řídicí program, čte a zapisuje data.

Jak S7-200 spojuje uložený program s fyzickými vstupy a výstupy

Základní provoz S7-200 je velmi jednoduchý:

- ❑ S7-200 přečte stav vstupů.
- ❑ Program uložený v automatu S7-200 použije tyto vstupy při zpracování řídicího programu. Při běhu programu S7-200 aktualizuje svoje data.
- ❑ Automat S7-200 zapíše data na výstup.

Obrázek 4-1 zobrazuje jednoduchý diagram vztahu kontaktního schématu a S7-200. V tomto příkladu je stav přepínače pro start motoru kombinován se stavem ostatních vstupů. Výpočtem těchto stavů pak S7-200 určí stav výstupu, který je přenesen na akční člen, jenž spustí motor.

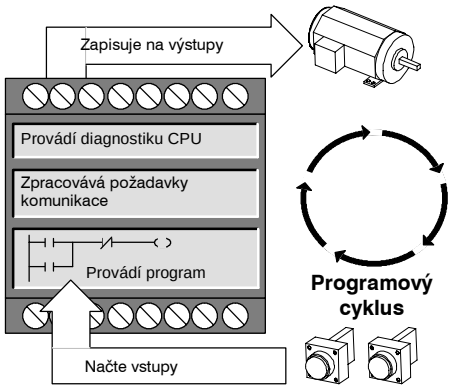


Obr. 4-1 Řízení vstupů a výstupů

S7-200 provádí své úkony v programovém cyklu

S7-200 cyklicky zpracovává řadu úloh. Toto cyklické provádění jednotlivých úloh se nazývá programový cyklus. Jak je znázorněno na obrázku 4-2, S7-200 provádí během programového cyklu všechny následující úlohy nebo jejich větší část:

- ❑ Čtení vstupů: S7-200 kopíruje stav fyzických vstupů do registru obrazu vstupů.
- ❑ Provádění řídicí logiky programu: S7-200 provede instrukce programu a hodnoty uloží do různých oblastí paměti.
- ❑ Zpracování požadavků komunikace: S7-200 provede všechny úlohy požadované pro komunikaci.
- ❑ Provádění autodiagnostiky CPU: S7-200 kontroluje, zda firmware, paměť pro program a všechny rozšiřovací moduly pracují správně.
- ❑ Zapisování na výstupy: hodnoty uložené v registru obrazu výstupů jsou zapsány na fyzické výstupy.



Obr. 4-2 Programový cyklus automatu S7-200

Provádění uživatelského programu závisí na tom, zda se S7-200 nalézá v režimu STOP nebo RUN. V režimu RUN program běží, v režimu STOP nikoliv.

Čtení vstupů:

Digitální vstupy: Každý programový cyklus začíná přečtením aktuálních hodnot na digitálních vstupech a zápisem do registru obrazu vstupů.

Analogové vstupy: Pokud není povoleno filtrování analogových vstupů, automat S7-200 neaktualizuje analogové vstupy z rozšiřovacích modulů při běžném programovém cyklu. Analogový filtr je instalován kvůli získání stabilnějšího signálu. Analogový filtr můžete nastavit pro každý analogový vstupní bod.

Pokud je filtrování analogového vstupu povoleno, S7-200 aktualizuje analogový vstup jednou za programový cyklus, provede funkci filtrování a interně uloží filtrovanou hodnotu. Tato hodnota je pak k dispozici pokaždé, když program přistupuje na tento analogový vstup.

Pokud není povoleno analogové filtrování, S7-200 přečte hodnotu analogového vstupu z rozšiřovacích modulů pokaždé, když program přistupuje na tento analogový vstup.

Analogové vstupy AIW0 a AIW2 na CPU 224XP se aktualizují při každém programovém cyklu nejnovějším výsledkem z analogově-číslicového převodníku. Tento převodník je průměrový (sigma-delta) a hodnoty obvykle nebudou potřebovat softwarové filtrování.



Tip

Filtrování analogového vstupu je instalováno pro získání stabilnější analogové hodnoty. Filtr použijte pro aplikace, kde se pomalu mění hodnota vstupního signálu. Pokud se signál mění rychle, analogový filtr by neměl být použit.

Analogový filtr nepoužívejte pro moduly, které předávají digitální informace nebo zprávy o poruše analogovými signály. Analogovou filtraci vždy zablokujte pro RTD moduly, termočlánkové moduly a moduly AS-Interface Master.

Provádění programu

Během programového cyklu S7-200 začíná program vykonávat instrukce od první až po poslední. Instrukce bezprostředního vstupu/výstupu poskytují v době běhu programu nebo během přerušení bezprostřední přístup na vstupy a výstupy.

Pokud v programu používáte přerušení, jsou podprogramy přerušení uložené jako součást programu. Podprogramy přerušení se neprovádějí jako součást běžného programového cyklu, ale provedou se tehdy, je–li splněna podmínka pro přerušení (což může nastat v libovolném okamžiku programového cyklu).

Zpracování požadavků komunikace

Během této fáze programového cyklu zpracuje S7-200 všechny zprávy, které byly přijaty z komunikačního rozhraní nebo inteligentních vstupních/výstupních modulů.

Provádění autodiagnostiky CPU

Během této fáze programového cyklu zkontroluje S7-200 správný provoz CPU a stav všech rozšiřovacích modulů.

Zapisování na digitální výstupy

Na konci každého programového cyklu zapisuje S7-200 hodnoty uložené v registru obrazu výstupů na digitální výstupy. (Analogové výstupy jsou aktualizovány okamžitě, nezávisle na programovém cyklu.)

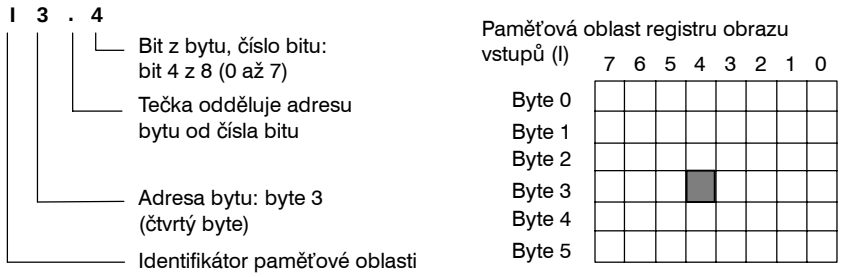
Přístup k datům

Programovatelný automat S7-200 uchovává informace v různých paměťových oblastech, které mají jedinečné adresy. Můžete explicitně určit konkrétní adresu paměti. To programu umožní přímý přístup k informacím. Tabulka 4-1 ukazuje rozsahy, které mohou být reprezentovány daty různých velikostí.

Tabulka 4-1 Desítkové a šestnáctkové rozsahy pro různé velikosti dat

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	*Nelze*	*Nelze*	+1.175495E–38 až +3.402823E+38 (kladné) –1.175495E–38 až –3.402823E+38 (záporné)

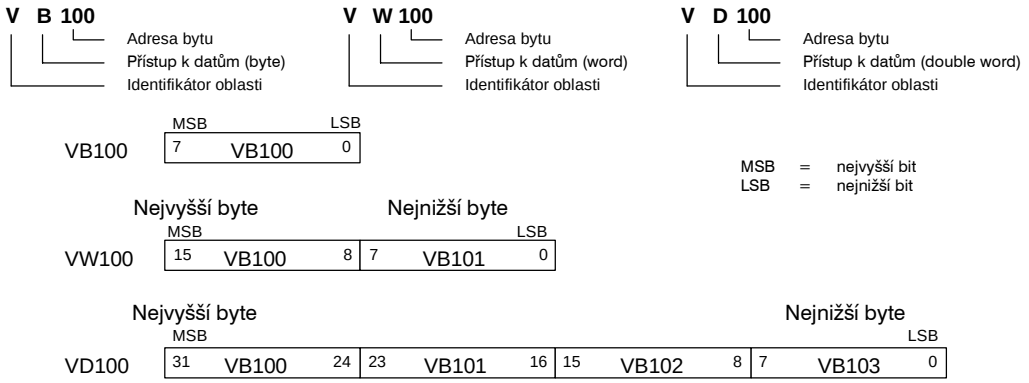
Když chcete přistoupit k nějakému bitu v paměťové oblasti, specifikujete adresu, která se skládá z identifikátoru paměťové oblasti, adresy bytu a čísla bitu. Obrázek 4-3 ukazuje příklad přístupu k bitu (často se nazývá adresování "byte.bit"). V tomto příkladu za paměťovou oblastí a adresou bytu (I = vstup a 3 = byte 3) následuje tečka ("."), která odděluje bitovou adresu (bit 4).



Obr. 4-3 Adresování byte.bit

K datům ve většině paměťových oblastí (V, I, Q, M, S, L, a SM) typu byte, word nebo double word můžete 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, jak to ukazuje obrázek 4-4.

K datům v jiných paměťových oblastech (jako jsou T, C, HC a v akumulátorech) se přistupuje s použitím formátu adresy s identifikátorem oblasti a číslem zařízení.



Obr. 4-4 Porovnání přístupu do stejné oblasti paměti k byte, word a double word

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. Přistupovat k registru obrazu vstupů můžete 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 můžete 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. Můžete ji také používat pro ukládání ostatních dat souvisejících s procesem nebo úlohou. Do paměťové oblasti V můžete 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

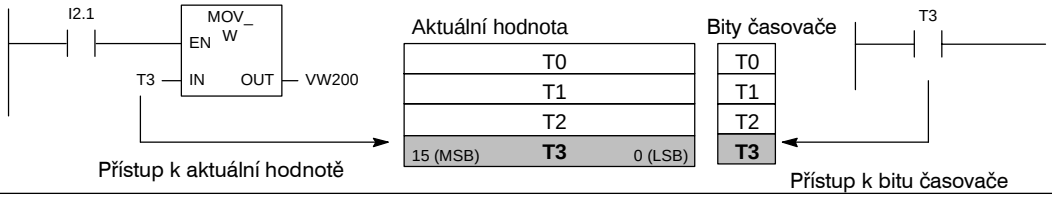
Oblast paměti časovačů: T

Časovače mají časové rozlišení 1 ms, 10 ms nebo 100 ms. K časovači patří dvě proměnné:

- Aktuální hodnotu: 16bitové číslo, které reprezentuje aktuální čas napočítaný časovačem.
- Bit časovače: tento bit má hodnotu 0 nebo 1 podle výsledku porovnání aktuální a nastavené hodnoty. Nastavená hodnota je součástí instrukce časovače.

Přístup k proměnným získáte použitím adresy časovače (T + číslo časovače). Přístup k bitu časovače nebo k aktuální hodnotě závisí na použité instrukci: instrukcí s bitovými operandy přistupují k bitu časovače, instrukce s operandy word přistupují k aktuální hodnotě. Jak je ukázáno na obrázku 4-5, instrukce kontaktu "V klidu otevřen" vyhodnotí bit časovače, instrukce "Přesunout word" umožní přístup k aktuální hodnotě časovače.

Formát: *T[číslo časovače]* T24



Obr. 4-5 Přístup k bitu časovače nebo k jeho aktuální hodnotě

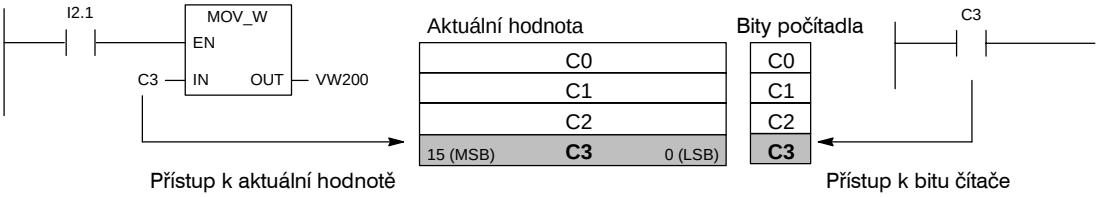
Oblast paměti čítačů: C

Programovatelný automat S7-200 má tři typy čítačů, která počítají každou událost s přechodem nahoru nebo dolů na vstupu (vstupech) čítačů: jeden typ počítá pouze nahoru, druhý pouze dolů a poslední nahoru i dolů. Každý čítač má dvě proměnné:

- Aktuální hodnotu: 16bitové číslo, kde je uložen aktuální počet událostí.
- Bit čítače: tento bit má hodnotu 0 nebo 1 podle výsledku porovnání aktuální a nastavené hodnoty. Nastavená hodnota je součástí instrukce čítače.

Přístup k proměnným získáte použitím adresy čítače (C + číslo čítače). Přístup k bitu čítače nebo k aktuální hodnotě závisí na použité instrukci: instrukce s bitovými operandy přistupují k bitu čítače, instrukce s operandy word přistupují k aktuální hodnotě. Jak je vidět na obrázku 4-6, instrukce kontaktu "V klidu otevřen" vyhodnotí bit čítače, instrukce "Přesunout word" umožní přístup k aktuální hodnotě čítače.

Formát: *C[číslo čítače]* C24



Obr. 4-6 Přístup k bitu čítače nebo k jeho aktuální hodnotě

Vysokorychlostní čítače: HC

Vysokorychlostní čítače počítají události s vysokou frekvencí nezávisle na programovém cyklu. Hodnota čítače je vyjádřena 32bitovým celým číslem. Pro přístup k hodnotě vysokorychlostního čítače je třeba zadat adresu čítače tak, že zadáme typ paměti (HC) a číslo čítače (např. HC0). Aktuální hodnotu čítače je možné pouze číst a je možné k ní přistupovat pouze jako k double word (32 bitů).

Formát: HC*[číslo vysokorychlostního čítače]* HC1

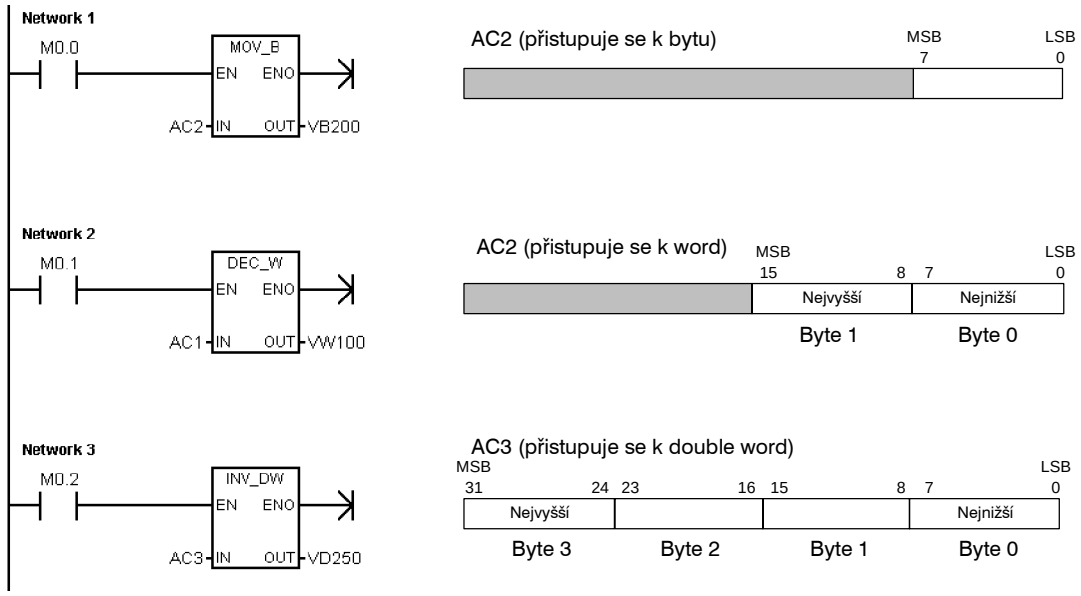
Akumulátory: AC

Akumulátory se používají na čtení a zápis podobně jako paměť. Akumulátory můžete například použít na předávání parametrů do podprogramů nebo pro uložení mezivýsledků výpočtů. Programovatelný automat S7-200 má čtyři 32bitové akumulátory (AC0, AC1, AC2 a AC3). K datům v akumulátorech můžete přistupovat po bytech, word nebo double word.

Velikost dat, kterou chcete číst nebo zapisovat, je určena typem instrukce pro práci s akumulátorem. Jak je patrné z obrázku 4-7, při přístupu k bytu přistupujete k 8 nejnižším bitům akumulátoru, při přístupu k word přistupujete k 16 nejnižším bitům akumulátoru. Při přístupu k double word využijete všech 32 bitů akumulátoru.

Informace o tom, jak používat akumulátory v přerušeních, najdete v kapitole 6 – Pokyny pro přerušení.

Formát: AC*[číslo akumulátoru]* AC0



Obr. 4-7 Přístup k akumulátorům

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. (Podrobné informace o SM bitech jsou uvedeny v příloze D.) K datům v této paměti můžete přistupovat po bitech, bytech, word nebo double word:

Bit:	SM*[adresa bytu].[adresa bitu]*	SM0.1
Byt, 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.



Tip

V programovacím jazyku LAD nebo FBD rezervuje STEP 7–Micro/WIN poslední čtyři byty lokální paměti pro vlastní použití.

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 určená 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. AIW0, 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 AIW0, AIW2 nebo AIW4). Hodnoty analogových výstupů se dají pouze zapisovat.

Formát:	AQW*[adresa počátečního bytu]*	AQW4

Paměťová oblast sekvenčních řídicích relé (SCR): S

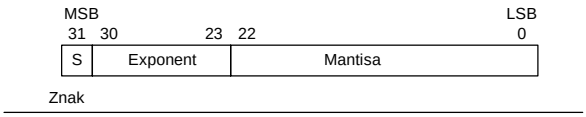
SCR neboli bity S se používají pro organizaci operací strojů nebo kroků v ekvivalentních segmentech programu. SCR umožňují logické členění řídicího programu. K datům v této paměti můžete přistupovat po bitech, bytech, word nebo double word.

Bit:	S*[adresa bytu].[adresa bitu]*	S3.1
Byte, word, double word:	S*[velikost][adresa počátecního bytu]*	SB4

Formát pro reálná čísla

Reálná čísla jsou reprezentována 32bitovými čísly s jednoduchou přesností, jejichž formát popisuje norma ANSI/IEEE 754–1985. Viz obrázek 4-8. Reálná čísla je možné číst jako data double word.

V automatu S7-200 jsou čísla s pohyblivou řádovou čárkou přesná na 6 desetinných míst. Proto je možné při vkládání konstanty s pohyblivou desetinou čárkou specifikovat maximálně 6 desetinných míst.



Obr. 4-8 Formát reálného čísla

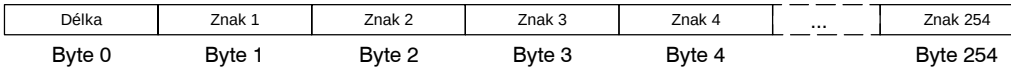
Přesnost při výpočtech s reálnými čísly

Výpočty, do kterých patří dlouhé řady hodnot, obsahující velmi velká a velmi malá čísla, mohou dát nepřesné výsledky. K tomu může dojít, pokud se čísla liší o 10 na *x–tou*, kde $x > 6$.

Například: $100\,000\,000 + 1 = 100\,000\,000$

Formát řetězce

Řetězec je posloupnost znaků, kde každý znak je uložen jako byte. První byte řetězce určuje jeho délku, to znamená počet znaků. Obrázek 4-9 ukazuje formát řetězce. Řetězec může mít délku 0 až 254 znaků plus byte udávající délku, takže maximální délka řetězce je 255 bytů. Řetězcová konstanta je omezena na 126 bytů.



Obr. 4-9 Formát řetězce

Specifikace konstanty pro instrukce S7-200

V mnoha instrukcích automatu S7-200 můžete použít konstantu. Konstanty mohou být typu byte, word nebo double word. Programovatelný automat S7-200 ukládá všechny konstanty jako binární čísla reprezentovaná ve formátu desítkovém, šestnáctkovém, ACSII formátu nebo jako reálná čísla. Viz tabulku 4-2.

Tabulka 4-2 Reprezentace konstant

Reprezentace	Formát	Příklad
Desítkové	[desítková hodnota]	20047
Šestnáctkové	16#[šestnáctková hodnota]	16#4E4F
Binární	2#[binární číslo]	2#1010_0101_1010_0101
ASCII	'[ASCII text]'	'ABCD'
reálný	ANSI/IEEE 754–1985	+1.175495E–38 (kladné) –1.175495E–38 (záporné)
Řetězec	"[řetězcový text]"	"ABCDE"



Tip

CPU S7-200 nepodporuje "kontrolu typu dat" (jako například specifikaci toho, že konstanta se ukládá jako integer, signed integer nebo jako double integer). Například instrukce "Sečíst" může použít hodnotu na adrese VW100 jako signed integer a instrukce XOR může tuto hodnotu použít jako binární hodnotu bez znaménka.

Adresování lokálních a rozšiřovacích I/O

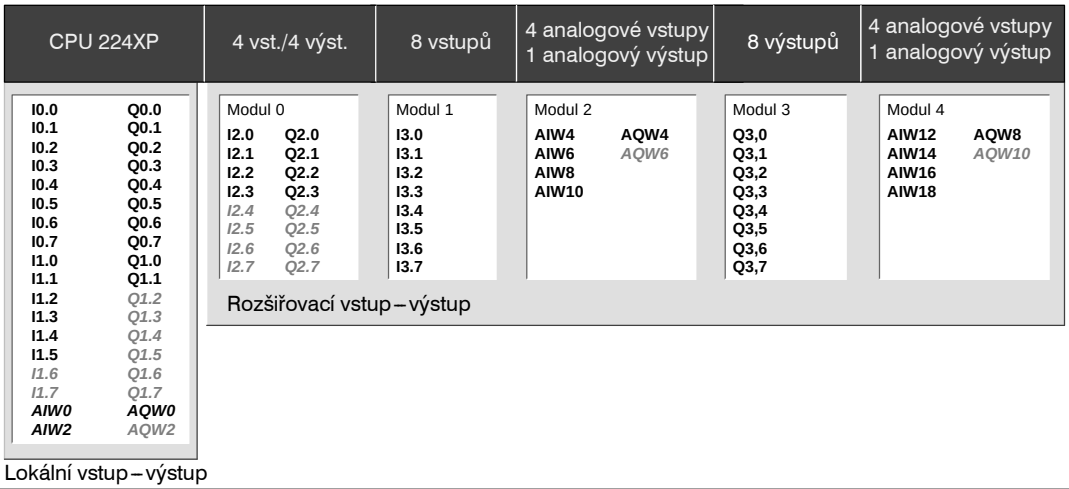
Lokální vstupy a výstupy, které jsou integrované přímo v CPU, mají pevně stanovené adresy. Je možné přidat I/O body CPU S7-200 připojením rozšiřovacích modulů k pravé straně CPU, a zvýšit tak počet vstupů nebo výstupů. Adresy bodů v rozšiřovacích modulech jsou určeny typem vstupu nebo výstupu a umístěním modulu v řetězci s přihlédnutím k předcházejícímu vstupnímu nebo výstupnímu modulu stejného typu. Například výstupní modul neovlivňuje adresy bodů ve vstupním modulu a naopak. Obdobně analogové moduly neovlivňují adresování digitálních modulů a naopak.

Tip

Pro digitální vstup/výstup je vždy rezervováno místo v registru obrazů v přírůstcích po osmi bitech (jeden byte). Pokud modul nevyužívá všechny bity z rezervovaného bytu, není možné je přiřadit dalším modulům v řetězci I/O modulů. V každém programovém cyklu jsou nevyužité bity vstupních modulů nastaveny na nulu.

Analogové vstupní/výstupní body mají vždy vymezená místa s adresami po dvou bodech. Pokud modul nevyužívá všechny bity z rezervovaného slova, není možné je využít ani je není možné přiřadit dalším modulům v řetězci I/O modulů.

Na obrázku 4-10 je příklad číslování vstupů a výstupů pro konkrétní konfiguraci hardwaru. Mezery v adresách (zvýrazněné šedou kurzívou) není možné programem využít.



Obr. 4-10 Příklad I/O adres pro lokální a rozšiřovací I/O (CPU 224XP)

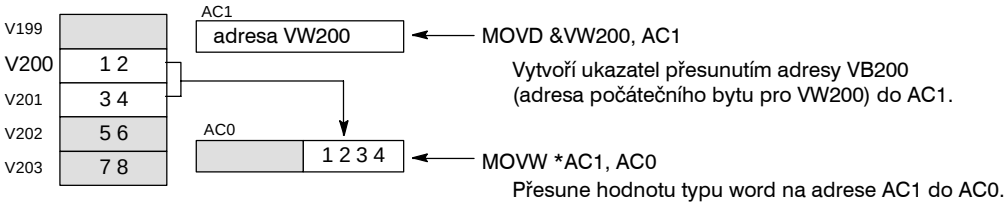
Použití ukazatelů pro nepřímé adresování paměťových oblastí S7-200

Nepřímé adresování používá pro přístup k datům v paměti ukazatel. Ukazatel je místo v paměti s velikostí double word, které obsahuje adresu jiného místa v paměti. Jako ukazatel je možné používat místa v paměti V , místa v paměti L nebo registry akumulátorů (AC1, AC2, AC3). Pro vytvoření ukazatele se používá instrukce ”Přesunout double word”, kterou se přesune adresa nepřímo adresovaného místa v paměti na adresu ukazatele. Ukazatel je také možné předat podprogramu jako parametr.

Automat S7-200 umožňuje ukazatelům přístup do následujících paměťových oblastí: I, Q, V, M, S, AI, AQ, SM, T (pouze aktuální hodnota) a C (pouze aktuální hodnota). Nepřímé adresování se nemůže použít pro přístup k jednotlivému bitu nebo pro přístup do paměťových oblastí HC nebo L.

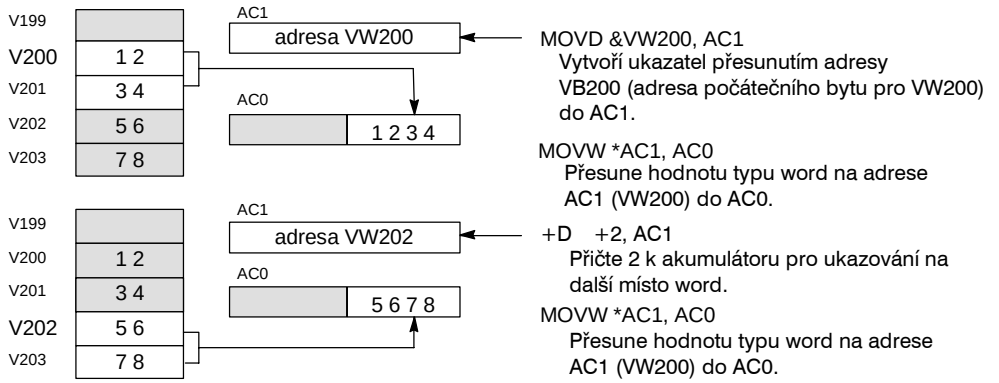
Pro nepřímý přístup k datům na dané paměťové adrese se vytvoří ukazatel na toto místo vložením ampersandu (&) a paměťového místa, které má být adresováno. Vstupní operand instrukce musí začínat ampersandem (&) označujícím, že se na místo určené výstupním operandem instrukce (ukazatel) má přesunout adresa paměťového místa a ne jeho obsah.

Vložení hvězdičky (*) před operand instrukce znamená, že operand je ukazatel. Podle příkladu na obrázku 4-11 vložení *AC1 určuje, že AC1 je ukazatel na hodnotu typu word, na kterou odkazuje instrukce ”Přesunout word”. V tomto příkladu se hodnoty uložené ve VB200 i ve VB201 přesunou do akumulátoru AC0.



Obr. 4-11 Vytvoření a užívání ukazatele

Jak je vidět na obrázku 4-12, hodnotu ukazatele je možné změnit. Protože ukazatele jsou 32bitové hodnoty, může být jeho hodnota změněná pomocí instrukcí pracujících s hodnotou double word. Pro změnu hodnot ukazatele se mohou použít jednoduché matematické operace, například přičtení nebo inkrementace hodnoty.



Obr. 4-12 Změna ukazatele

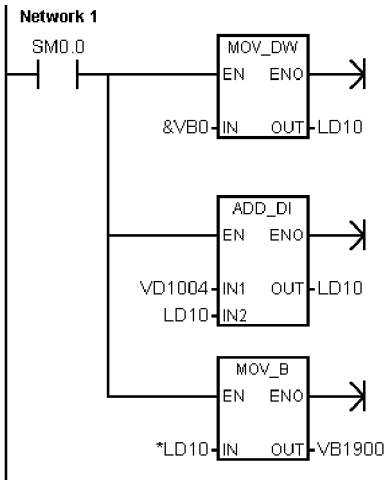


Tip

Je nutné provést přizpůsobení na typ dat, ke kterým se přistupuje: pro přístup k datům typu byte se hodnota ukazatele zvyšuje o 1; pro přístup k datům typu word nebo aktuální hodnotě časovače nebo čítače se hodnota ukazatele zvyšuje o 2 nebo se přičtou násobky dvou; pro přístup k datům typu double word se hodnota ukazatele zvyšuje o 4 nebo se přičtou násobky čtyř.

Ukázkový program použití offsetu pro přístup k datům v paměti V

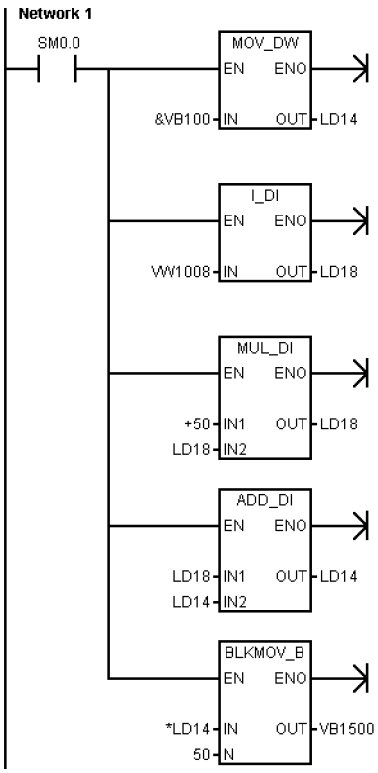
Tento příklad používá LD10 jako ukazatel na adresu VB0. Ukazatel se pak zvětší o offset uložené ve VD1004. Pak LD10 ukazuje na jinou adresu v paměti V (VB0 + offset). Hodnota uložená na adrese v paměti V, na kterou ukazuje LD10, je pak zkopírována do VB1900. Změnou hodnoty ve VD1004 se získá přístup na libovolné místo paměti V .




```
Network 1   //Použití offsetu pro čtení hodnoty
            //jakéhokoli místa VB:
            //
            //1. Načte se počáteční adresa
            //paměti V do ukazatele.
            //2. Hodnota offsetu se přičte k ukazateli.
            //3. Zkopíruje se hodnota z paměti V
            //(offset) do VB1900.
            //
LD      SM0.0
MOVD    &VB0, LD10
+D      VD1004, LD10
MOVB    *LD10, VB1900
```

Ukázkový program použití ukazatele pro přístup k datům v tabulce

V tomto příkladu se LD14 použije jako ukazatel na recepturu uloženou v tabulce receptur, která začíná na VB100. Ve VW1008 je uložen index konkrétní receptury z tabulky. Pokud má každá receptura v tabulce délku 50 bytů , index se vynásobí 50, aby se získal offset pro počáteční adresu konkrétní receptury. Přičtením offsetu k ukazateli se získá přístup k příslušné receptuře z tabulky. V tomto příkladu je receptura zkopírována do 50 bytů, které začínají na VB1500.




```
Network 1   //Jak přesunout recepturu z tabulky receptur:
            // – Každá receptura má délku 50 bytů.
            // – Index (VW1008) určuje
            //   recepturu, která má být zavedena.
            //
            //1. Vytvoří se ukazatel na počáteční adresu
            //   tabulky receptur.
            //2. Index receptury se převede na
            //   hodnotu double-word.
            //3. Offset se vynásobí, aby odpovídal
            //   velikosti receptury.
            //4. Hodnota upraveného offsetu se přičte k ukazateli.
            //5. Vybraná receptura se přesune na
            //   VB1500 přes VB1549.
LD      SM0.0
MOVD    &VB100, LD14
ITD     VW1008, LD18
*D      +50, LD18
+D      LD18, LD14
BMB     *LD14, VB1500, 50
```

Ukládání a obnova dat

Programovatelný automat S7-200 poskytuje více funkcí pro bezpečné uchování uživatelského programu a dat.

- Paměť remanentních dat – paměťové oblasti pro data, které uživatel volí jako neměnné od zapnutí po vypnutí napájení, pokud nebyl vybit velkokapacitní kondenzátor a doplňkový bateriový modul. Aktuální hodnoty V, M, časovače a hodnoty čítače jsou jediné paměťové oblasti pro data, které se mohou nastavit jako remanentní.
- Trvalá paměť – energeticky nezávislá paměť používaná pro uložení programového bloku, datového bloku, systémového bloku, vnucených hodnot, paměti M nastavené na uložení při výpadku proudu a vybraných hodnot zapsaných při běžném řízení uživatelského programu
- Paměťový modul – výměnná, energeticky nezávislá paměť, používaná pro uložení programového bloku, datového bloku, systémového bloku, receptur, datových záznamů a vnucených hodnot

K ukládání souborů dokumentace (doc, text, pdf atd.) do modulu můžete použít Průzkumník S7-200. Průzkumníka S7-200 můžete také použít k provádění celkové údržby souborů v paměťovém modulu (kopírování, mazání, vytváření adresářů a spouštění).

Při instalaci paměťového modulu je třeba nejdříve odstranit plastový kryt na CPU S7-200 a zasunout paměťový modul do konektoru. Tvar paměťového modulu je upraven tak, že může být zasunut pouze tehdy, je–li správně orientovaný.

Upozornění

Elektrostatický výboj může poškodit paměťový modul nebo svorkovnici na CPU S7-200.

Při manipulaci s modulem je nutné být kontaktu s uzemněnou vodivou podložkou nebo použít uzemněný vodivý náramek. Paměťový modul je třeba uchovávat ve vodivém obalu.

Download a upload projektu

Každý projekt se skládá z různých částí:

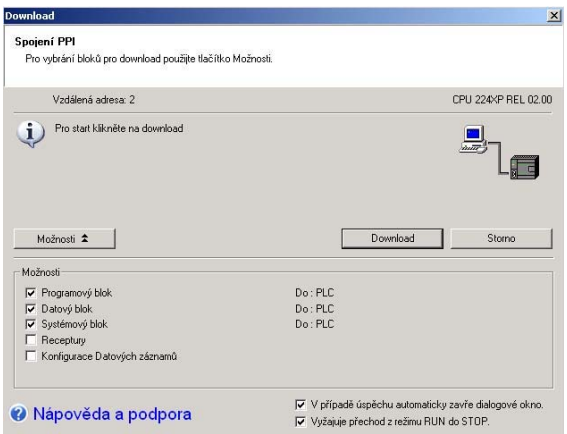
- Programový blok
- Datový blok (volitelný)
- Systémový blok (volitelný)
- Receptury (volitelné)
- Konfigurace datového záznamu (volitelné)

Když provádíte download projektu, programový blok, datový blok a systémový blok se ukládají bezpečně do trvalé paměti. Receptury a konfigurace datových záznamů se ukládají do paměťového modulu a nahrazují stávající receptury a datové záznamy. Programové prvky, které nejsou zahrnuty v operaci downloadu, jsou ponechány beze změny v trvalé paměti a paměťovém modulu.

Jestliže download projektu zahrnuje receptury nebo konfigurace datových záznamů, paměťový modul musí zůstat kvůli správné funkci programu nainstalován.

Postup downloadu projektu do CPU S7-200:

1. Vyberte příkaz menu **Soubor > Download**.
2. Klikněte na prvek projektu, který chcete načíst.
3. Klikněte na tlačítko “Download”.

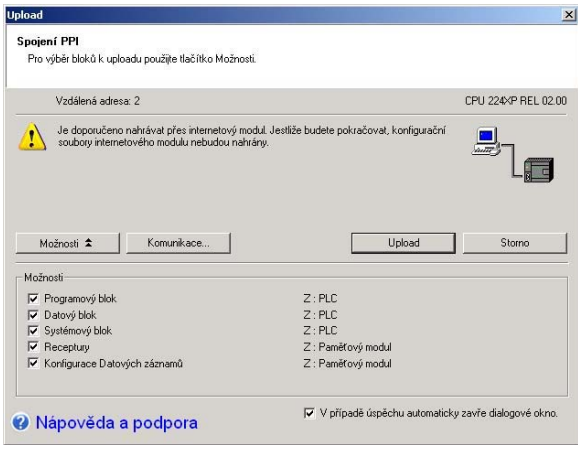


Obr. 4-13 Download projektu do CPU S7-200

Když provádíte upload projektu do počítače pomocí STEP 7–Micro/WIN, S7-200 nahraje programový blok, datový blok a systémový blok z trvalé paměti. Receptury a konfigurace datových záznamů se nahrávají z paměťového modulu. Data z datových záznamů se do počítače pomocí STEP7–Micro/WIN nenahrávají. K nahrávání dat z datových záznamů se používá Průzkumník S7-200 (viz kapitola 14).

Postup uploadu projektu z CPU S7-200:

1. Vyberte příkaz menu **Soubor > Upload**.
2. Klikněte na prvek projektu, který chcete nahrát.
3. Klikněte na tlačítko “Upload”.



Obr. 4-14 Upload projektu do S7-200

Uložení programu do paměťového modulu

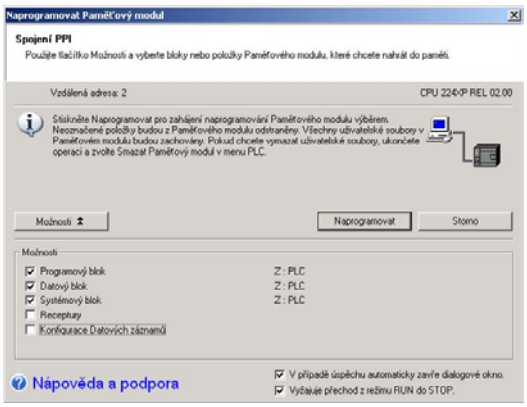
S7-200 vám umožňuje zkopírovat uživatelský program z jednoho CPU do jiného pomocí paměťového modulu. Můžete také rozšířit aktualizace pro kterýkoliv z těchto bloků ve vašem S7-200: programový blok, systémový blok nebo datový blok.

Před kopírování programových prvků do paměťového modulu vymaže STEP 7–Micro/WIN všechny programové prvky (včetně receptur a datových záznamů) kromě uživatelských souborů v paměťovém modulu. Jestliže nebude pro program dostatek místa z důvodu velikosti souborů, můžete provést jednu nebo dvě věci k vytvoření dostatečného úložného prostoru pro váš program. Můžete buď vymazat paměťový modul pomocí příkazu menu **PLC > Vymazat paměťový modul**, nebo můžete otevřít Průzkumníka S7-200 a odstranit uživatelské soubory, které už nebudete potřebovat.

Chcete–li programovat paměťový modul, PLC musí být v režimu STOP.

Postup uložení programu do paměťového modulu:

1. Zvolte příkaz menu **PLC > Programovat paměťový modul**.
2. Klikněte na projektový prvek, který chcete zkopírovat do paměťového modulu (implicitně jsou vybrány všechny programové prvky existující v projektu). Je–li vybrán systémový blok, vnucené hodnoty se rovněž zkopírují.
3. Klikněte na tlačítko “Program”.



Obr. 4-15 Uložit program do paměťového modulu

Programový blok, systémový blok, datový blok a všechny vnucené hodnoty se zkopírují z trvalé paměti v S7-200 do paměťového modulu. Receptury a konfigurace datovýh záznamů se zkopírují ze STEP 7–Micro/WIN do paměťového modulu.

Obnovení programu z paměťového modulu

Pro nahrání programu z paměťového modulu do S7-200 je nejdříve potřeba automat vypnout a s paměťovým modulem ho opět zapnout. Jestliže se některé bloky nebo vnucené hodnoty v paměťovém modulu liší od bloků nebo vnucených hodnot v S7-200, pak se všechny bloky v paměťovém modulu zkopírují do S7-200.

- ❏ Jestliže byl programový blok přenesen z paměťového modulu, v trvalé paměti bude nahrazen.
- ❏ Jestliže byl přenesen datový blok z paměťového modulu, datový blok v trvalé paměti se nahradí, celá paměť V se vymaže a inicializuje obsahem datového bloku.
- ❏ Jestliže byl systémový blok přenesen z paměťového modulu, systémový blok a vnucené hodnoty v trvalé paměti se nahradí a celá remanentní paměť se vymaže.

Jakmile je přenesený program uložen v trvalé paměti, můžete vyjmout paměťový modul. Jestliže jsou však receptury nebo datové záznamy stále v modulu, musíte nechat paměťový modul zasunutý. Zasunutý paměťový modul zpozdí režim RUN v následujících napájecích cyklech.

Poznámka

Při zapnutí CPU S7-200 s nainstalovaným paměťovým modulem naprogramovaným jiným typem CPU S7-200 může dojít k chybě. Paměťové moduly naprogramované v modelech CPU s nižším číslem se v modelech s číslem vyšším dají přečíst. Opačně to však není možné. Například paměťové moduly naprogramované v CPU 221 nebo CPU 222 lze přečíst v CPU 224, ale moduly naprogramované v CPU 224 nelze přečíst v CPU 221 nebo CPU 222.

Úplný seznam omezení použití paměťových modulů naleznete ve Volitelných zásuvných modelech (paměťový modul) v příloze A.

Ukládání dat z bitové paměti (M) při výpadku napájení

Pokud je prvních 14 bytů bitové paměti (MB0 až MB13) nakonfigurováno jako remanentní (uchovávají svoji hodnotu), jsou tyto byty v případě výpadku napájení S7-200 trvale uloženy v trvalé paměti. Implicitně je prvních 14 bytů paměti M zvoleno jako neremanentních.

Obnovení dat po zapnutí napájení

Při zapnutí napájení S7-200 obnoví programový blok a systémový blok z trvalé paměti. S7-200 potom ověří, zda velkokapacitní kondenzátor a volitelný bateriový modul, jsou – li nainstalovány, úspěšně uchovaly data uložená v paměti RAM. Pokud se data úspěšně zálohovala, ponechají se remanentní oblasti paměti uživatelských dat nezměněné. Neremanentní části paměti V se obnoví z obsahu datového bloku v trvalé paměti. Neremanentní části jiných oblastí paměti se vymažou.

Jestliže se obsah RAM neuchoval (např. po delším výpadku napájení), S7-200 vymaže všechny oblasti uživatelských dat, nastaví paměťový bit ztráty remanentních dat (SM0.2), obnoví paměť V z obsahu datového bloku v trvalé paměti a obnoví prvních 14 bytů paměti M z trvalé paměti, pokud tyto byty byly dříve nakonfigurovány jako remanentní.

Použití programu pro uložení paměti V do trvalé paměti

Hodnotu (byte, word nebo double word), uloženou kdekoliv v oblasti paměti V, můžete uložit do trvalé paměti. Proces uložení do trvalé paměti prodlouží čas programového cyklu maximálně o 5 ms. Nově zapsaná hodnota přepíše jakoukoliv předchozí hodnotu, uloženou v oblasti paměti V v trvalé paměti.

Operace uložení do trvalé paměti neaktualizuje data v paměťovém modulu.



Tip

Vzhledem k tomu, že je počet operací uložení do trvalé paměti (EEPROM) omezený (minimálně 100 000 a obvykle 1 000 000), je třeba zabezpečit, aby byly ukládány pouze nezbytné hodnoty. Jinak může dojít k opotřebení EEPROM a CPU může selhat. Operace zápisu by se měly provádět při výskytu specifických událostí, ke kterým nedochází často.

Například při jednom ukládání za programový cyklus S7-200, který trvá 50 ms, bude mít EEPROM životnost 5.000 sekund, což je méně než hodina a půl. Jestliže se naproti tomu hodnota ukládá jednou za hodinu, má EEPROM životnost minimálně 11 let.

Zkopírování paměti V do trvalé paměti

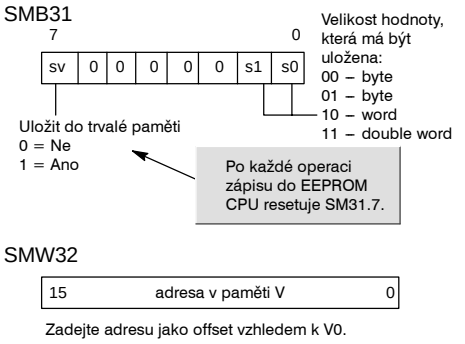
Byte 31 speciální paměti (SMB31) přikáže, aby automat zkopíroval hodnotu z paměti V do oblasti paměti V v trvalé paměti. V SMW32 je uložena adresa paměti, která má být zkopírována. Obrázek 4-16 ukazuje formát SMB31 a SMW32.

Pro naprogramování S7-200 pro uložení nebo zápis konkrétní hodnoty použijte následující kroky:

1. Adresu paměti V, která má být uložena, nahraje do SMW32.
2. Velikost dat uložte do SM31.0 a SM31.1, jak ukazuje obrázek 4-16.
3. SM31.7 se nastaví na 1.

Na konci každého programového cyklu S7-200 kontroluje SM31.7; pokud se SM31.7 rovná 1, uloží se daná hodnota do trvalé paměti. Operace se ukončí tím, že S7-200 resetuje SM31.7 na 0.

Hodnotu v paměti V neměňte, dokud operace ukládání neskončila.

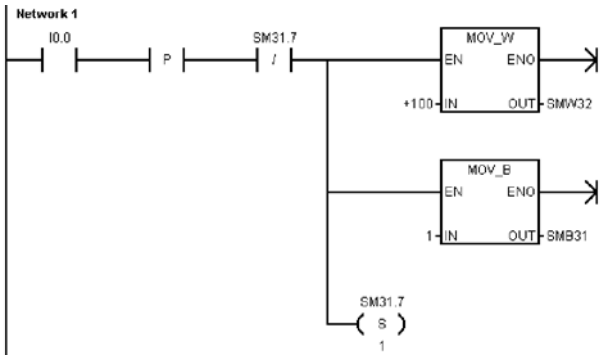


Obr. 4-16 SMB31 a SMW32

Vzorový program: Zkopírování paměti V do trvalé paměti

V tomto příkladu se do trvalé paměti zkopíruje VB100. Při náběhové hraně na I0.0 (pokud neprobíhá jiný přesun) se nahraje adresa místa v paměti V , které má být zkopírováno, do SMW32. Vybere se velikost paměti V, která bude zkopírována (1=byte; 2=word; 3=double word nebo real). Pak se SM31.7 nastaví tak, aby automat S7-200 na konci cyklu data přesunul.

S7-200 automaticky zresetuje SM31.7, když je přesun dokončen.




```
Network 1       //Přesun obsahu paměti V
                //(VB100) do
                //trvalé paměti
LD      I0.0
EU
AN      SM31.7
MOVW    +100, SMW32
MOVB    1, SMB31
S       SM31.7, 1
```

Výběr pracovního režimu CPU S7-200

S7-200 má dva pracovní režimy: režim STOP a režim RUN. Stavová LED dioda na předním panelu CPU signalizuje aktuální pracovní režim. V režimu STOP neprovádí S7-200 program a je možné nahrávat program (download) nebo konfigurovat CPU. V režimu RUN S7-200 program provádí.

- S7-200 je vybaven přepínačem pro změnu pracovního režimu. Přepínač (umístěný pod předními dvířky S7-200) je možné použít pro ruční výběr pracovního režimu: jeho přepnutí na režim STOP zastaví provádění programu, přepnutí na režim RUN spustí provádění programu. Přepnutím na režim TERM (terminál) se provozní režim nezmění.

  Jestliže je při výpadku napájení S7-200 ve stavu STOP nebo TERM, bude po obnovení napájení ve stavu STOP. Jestliže je při výpadku napájení S7-200 ve stavu RUN, bude po obnovení napájení ve stavu RUN.

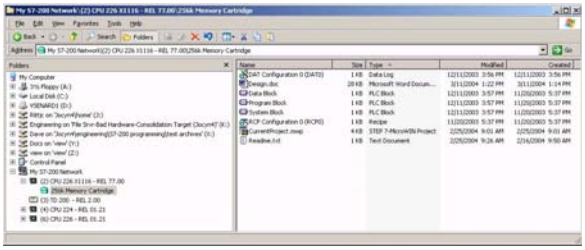
- Software STEP 7–Micro/WIN umožňuje měnit pracovní režim S7-200 online. Aby mohl software změnit pracovní režim, je nutné ručně nastavit přepínač na S7-200 buď na TERM, nebo na RUN. Pro změnu pracovního režimu je možné použít příkazy menu **PLC > STOP** nebo **PLC > RUN** nebo příslušná tlačítka na panelu nástrojů.

- Vložením instrukce STOP do programu se S7-200 přepne do stavu STOP. Tím se umožňuje zastavit provádění programu na základě uživatelského programu. Podrobné informace o instrukci STOP jsou uvedeny v kapitole 6.

Použití Průzkumníka S7-200

Průzkumník S7-200 je nadstavba k průzkumníku Windows, která poskytuje přístup k PLC automatu S7-200 a umožňuje prozkoumat obsah každého připojeného PLC. Mohou být určeny různé bloky, které se mohou nacházet buď v PLC, nebo v paměťovém modulu. Pro každý blok jsou k dispozici vlastnosti.

Protože Průzkumník S7-200 je nadstavbou k průzkumníku Windows, je podporována standardní navigace a chování jako u Windows.



Obr. 4-17 Průzkumník S7-200

Průzkumník S7-200 je mechanismus používaný ke čtení dat v datovém záznamu, uložených v paměťovém modulu. Další informace o datových záznamechviz kapitola 14.

Průzkumník S7-200 lze použít také ke čtení nebo zápisu uživatelských souborů do paměťového modulu. Mohou to být různé typy souborů, dokumenty Word, bitmapové soubory, soubory jpeg nebo projekty STEP 7–Micro/WIN.

Vlastnosti programovatelného automatu S7-200

Automat S7-200 má několik speciálních možností, které uživateli dovolují upravit funkce S7-200 tak, aby lépe vyhovovaly aplikaci.

Okamžité čtení a zápis vstupů a výstupů

Instrukční sada automatu S7-200 obsahuje instrukce pro okamžité čtení a zápis fyzických vstupů a výstupů. Tyto instrukce bezprostředního I/O umožňují přímý přístup na vstupní nebo výstupní bod, i když se normálně jako zdroj nebo cíl I/O operací používá registr obrazu vstupů a výstupů.

Při použití instrukcí okamžitého vstupu se odpovídající bity registru obrazu vstupů nezmění. Při použití instrukcí okamžitého výstupu se odpovídající bity registru obrazu výstupů přepíšou na aktuální hodnotu.



Tip
Pokud není povolené filtrování analogového vstupu, programovatelný automat S7-200 zachází s analogovými vstupy jako s bezprostředními daty. Když se zapíše hodnota na analogový výstup, je tento výstup bezprostředně aktualizován.

Obvykle bývá výhodné používat registr obrazu vstupů a výstupů místo přímého čtení hodnot vstupů a výstupů během provádění programu. Pro použití registrů obrazu existují tři důvody:

- Vzorkování všech vstupů na začátku programového cyklu synchronizuje a uchovává hodnoty vstupů pro celý programový cyklus. Po provedení programu se výstupy aktualizují podle registru obrazu výstupů, což má stabilizační efekt.
- Přístup do registru je daleko rychlejší než přístup přímo ke vstupům a výstupům; důsledkem toho je zrychlení celého programu.
- Vstupy a výstupy jsou bitové hodnoty, a proto se k nim musí přistupovat jako k bitům nebo k bytům; do registru obrazu je možné vstupovat po bitech, bytech, word nebo double word. Práce s registrem je proto efektivnější.

Přerušení programového cyklu

Pokud používáte přerušení, jsou podprogramy spojené s každou naprogramovanou událostí uloženy jako součást programu. Podprogramy přerušení se neprovádějí jako součást běžného programového cyklu, ale provedou se tehdy, je–li splněna podmínka pro přerušení (což může nastat v libovolném okamžiku programového cyklu).

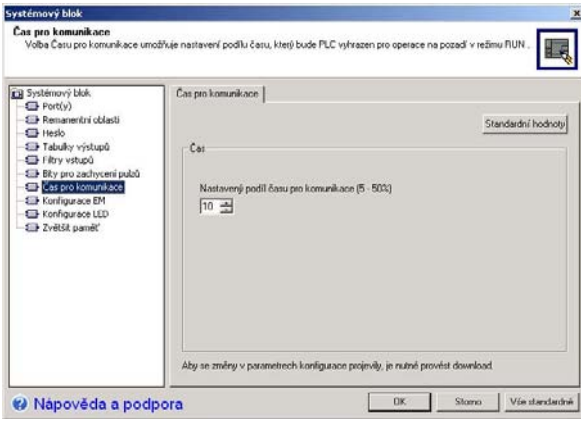
S7-200 obsluhuje přerušení na základě pravidla: kdo přijde první, je jako první obsloužen, ale s přihlédnutím k prioritě. Další informace poskytnou instrukce pro přerušení v kapitole 6.

Automat S7-200 vám umožňuje přidělit dobu zpracování pro editaci v režimu RUN a stav programu

Můžete konfigurovat procento programového cyklu, které má být určeno pro zpracování kompilace režimu RUN nebo pro stav programu. (Editace v režimu RUN a stav jsou funkce STEP 7–Micro/WIN, které usnadňují ladění programu.) Zvětšením procenta času určeného pro tyto dva úkoly se prodlouží doba programového cyklu a řídicí proces probíhá pomaleji.

Implicitně je procento programového cyklu, určené pro zpracování editací v režimu RUN a pro stav programu, nastaveno na 10 %. Toto nastavení bylo zvoleno jako rozumný kompromis mezi zpracováním kompilačních a stavových operací a minimalizuje dopad na řídicí proces. Tuto hodnotu můžete upravovat po 5 % až na maximálních 50 %. Nastavení časového úseku programového cyklu pro komunikaci na pozadí:

1. Vyberte příkaz menu **Zobrazit> Komponenty> Systémový blok** a zvolte “Čas na pozadí”.
2. Na záložce “Pozadí” použijte rozevírací okno a vyberte čas na pozadí pro komunikaci.
3. Kliknutím na OK výběr uložíte.
4. Proveďte download modifikovaného systémového bloku do S7-200.

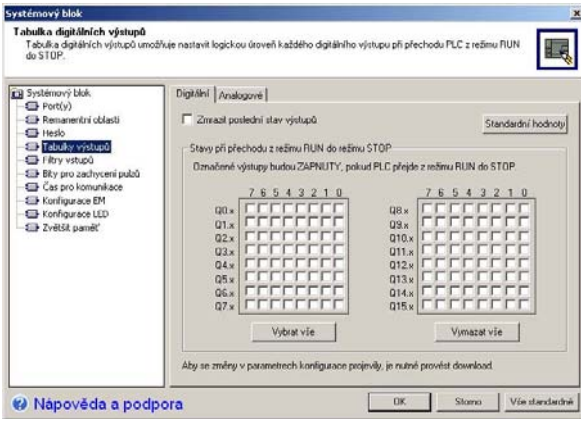


Obr. 4-18 Čas pro komunikace

Nastavení stavu digitálních výstupů v režimu STOP

Při přechodu do stavu STOP umožňuje tabulka výstupů automatu S7-200 nastavit stav digitálních výstupů na přednastavené hodnoty nebo výstupy ponechat ve stavu, ve kterém byly před tímto přechodem. Tabulka výstupu je součástí systémového bloku, který je stažen a uložen do S7-200.

1. Vyberte příkaz menu **Zobrazit> Komponenty > Systémový blok** a zvolte “Tabulka výstupů”. Klikněte na záložku “Digitální”.
2. Chcete–li výstupy ponechat v jejich posledním stavu, zaškrtněte volbu “Zmrazit výstupy”.
3. Chcete–li kopírovat hodnoty z tabulky na výstupy, vložte je kliknutím na zaškrtávací okénko pro každý výstupní bit, který chcete nastavit po přechodu RUN-STOP na log. 1. Všechny implicitní hodnoty v tabulce jsou log. 0.
4. Kliknutím na OK výběr uložíte.
5. Proveďte download modifikovaného systémového bloku do S7-200.

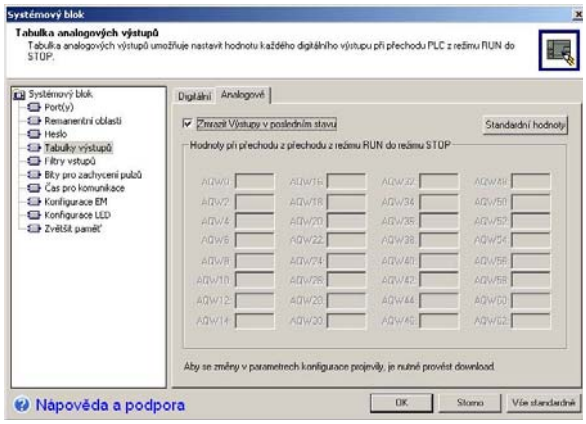


Obr. 4-19 Tabulka digitálních výstupů

S7-200 vám umožní konfigurovat hodnotu analogových výstupů

Tabulka analogových výstupů vám umožní nastavit analogové výstupy na přednastavené hodnoty po přechodu z režimu RUN na STOP nebo zachovat hodnoty výstupů, které existovaly před přechodem do režimu STOP. Tabulka výstupu je součástí systémového bloku, který je načten a uložen v CPU S7-200.

1. Vyberte příkaz menu **Zobrazit> Komponenty > Systémový blok** a zvolte “Tabulka výstupů”. Klikněte na záložku “Analogové”.
2. Chcete–li výstupy ponechat v jejich posledním stavu, zaškrtněte volbu “Zmrazit výstupy”.
3. Tabulka Zmražené hodnoty vám umožní nastavit analogové výstupy na známou hodnotu (–32768 až 37262) při přechodu z režimu RUN na STOP.
4. Kliknutím na OK výběr uložíte.
5. Proveďte download modifikovaného systémového bloku do S7-200.



Obr. 4-20 Tabulka analogových výstupů

Definování oblastí paměti, které udržují svoji hodnotu i při výpadku napájení

Můžete definovat až šest remanentních oblastí paměti, které udržují svoji hodnotu při výpadku napájení. Remanentní oblasti se dají definovat v těchto částech paměti: V, M, C a T. Hodnoty si mohou uchovávat pouze remanentní časovače (TONR). Implicitní nastavení prvních 14 bytů paměti M je neremanentní.

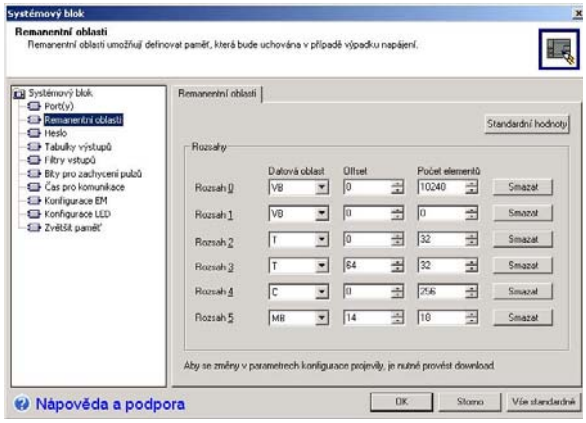
Časovače a čítače mohou mít remanentní pouze aktuální hodnoty: bity časovače a čítače nejsou remanentní.

Tip

Změna oblasti MB0 až MB13 na remanentní spustí speciální funkci, která při výpadku napájení automaticky ukládá tuto oblast do trvalé paměti.

Chcete–li definovat remanentní paměť:

1. Vyberte příkaz menu **Zobrazit> Komponenty > Systémový blok** a zvolte “Remanentní oblasti”.
2. Vyberte oblasti paměti, které si mají udržet svoji hodnotu při výpadku napájení, a klikněte na OK.
3. Proveďte download modifikovaného systémového bloku do S7-200.



Obr. 4-21 Remanentní paměť

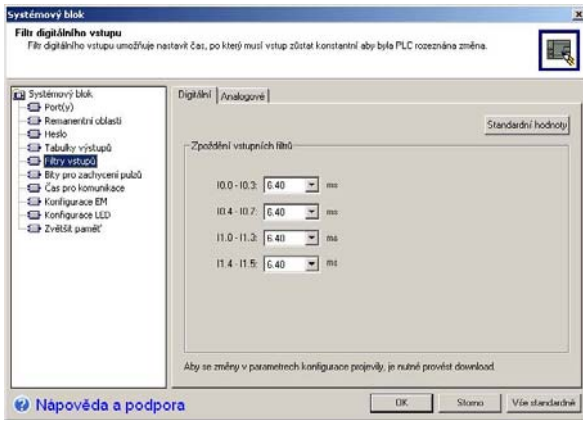
Filtrace digitálních vstupů

S7-200 umožňuje použít vstupní filtr, který definuje čas zpoždění (volitelný od 0,2 ms do 12,8 ms) pro některé nebo pro všechny lokální digitální vstupní body. Toto zpoždění pomáhá filtrovat šum z kabeláže, který by mohl způsobit nepředpokládanou změnu stavu na vstupu.

Vstupní filtr je částí systémového bloku, který je nahrán a uložen v S7-200. Implicitní čas filtru je 6,4 ms. Jak je vidět na obrázku 4-22, každá hodnota zpoždění platí pro skupinu vstupů.

Chcete–li konfigurovat dobu zpoždění pro vstupní filtr:

1. Vyberte příkaz menu **Zobrazit> Komponenty> Systémový blok** a zvolte “Filtry vstupů”. Klikněte na záložku “Digitální”.
2. Vložte dobu zpoždění pro každou skupinu vstupu a klikněte na OK.
3. Proveďte download modifikovaného systémového bloku do S7-200.



Obr. 4-22 Digitální vstupní filtr

Tip

Filtr digitálních vstupů ovlivňuje vstupní hodnotu z hlediska použití čtecí instrukce, přerušení od vstupu a zachycení pulzů. V závislosti na konfiguraci filtru by mohl program vynechat událost pro přerušení nebo zachycení pulzů. Vysokorychlostní čítače počítají události na nefiltrovaných vstupech.

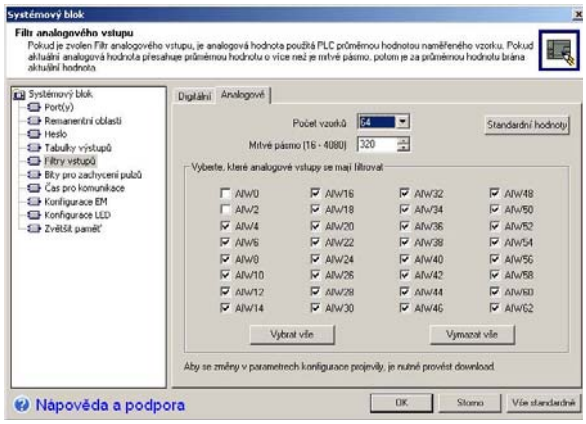
Filtrace analogových vstupů

Pro S7-200 je možné vybrat softwarový filtr pro jednotlivé analogové vstupy. Filtrovaná hodnota je průměr ze součtu zvoleného počtu vzorků analogového vstupu. Specifikace filtru (počet vzorků a mrtvé pásmo) je stejná pro všechny filtrované analogové vstupy.

Filtr má velmi rychlou odezvu a umožňuje, aby se rychle promítly i velké změny ve vstupním signálu. Filtr provádí skokovou změnu na poslední hodnotu analogového vstupu, když vstup překročí specifikovanou změnu aktuální hodnoty. Tato změna se nazývá mrtvé pásmo a je dána počtem jednotek digitální hodnoty analogového vstupu.

Implicitní konfigurace má umožnit filtrování pro všechny analogové vstupy kromě AIW0 a AIW2 na CPU 224XP.

1. Vyberte příkaz menu **Zobrazit> Komponenty> Systémový blok** a zvolte “Filtry vstupů”. Klikněte na záložku “Analogové”.
2. Vyberte analogové vstupy, které chcete filtrovat, počet vzorků a mrtvé pásmo.
3. Klikněte na OK.
4. Proveďte download modifikovaného systémového bloku do S7-200.



Obr. 4-23 Filtry analogových vstupů

Tip
Analogový filtr nepoužívejte pro moduly, které předávají digitální informace nebo zprávy o poruše analogovými signály. Analogovou filtraci vždy zablokujte pro RTD moduly, termočlánkové moduly a moduly AS-Interface Master.

Tip
AIW0 a AIW2 na CPU 224XP se filtrují analogově–číslicovým převodníkem a obvykle nepotřebují další softwarový filtr.

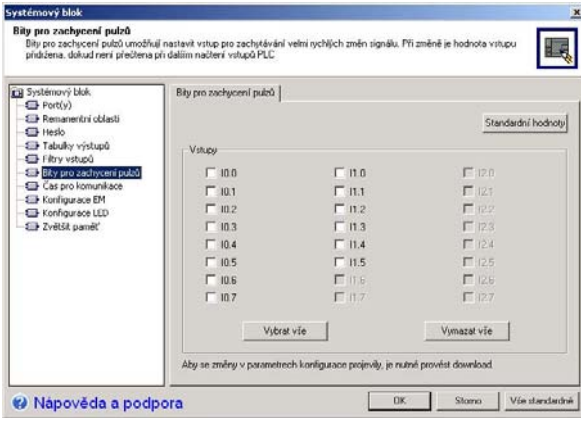
Zachycení krátkých pulzů

S7-200 nabízí pro některé nebo pro všechny lokální digitální vstupy funkci zachycení pulzů. Tato funkce umožňuje zachytit pulzy, které jsou natolik krátké, že je S7-200 na začátku programového cyklu při čtení digitálních vstupů nezaregistruje. Pokud je povolena funkce zachycení pulzů na vstupu, potom bude změna stavu na vstupu pozdržena až do příštího cyklu aktualizace. Tím je zajištěno, že pulz, který trvá pouze krátce, je zachycen a pozdržen, dokud S7-200 nepřečte vstupy.

Funkci zachycení pulzů je možné povolit pro každý lokální digitální vstup jednotlivě.

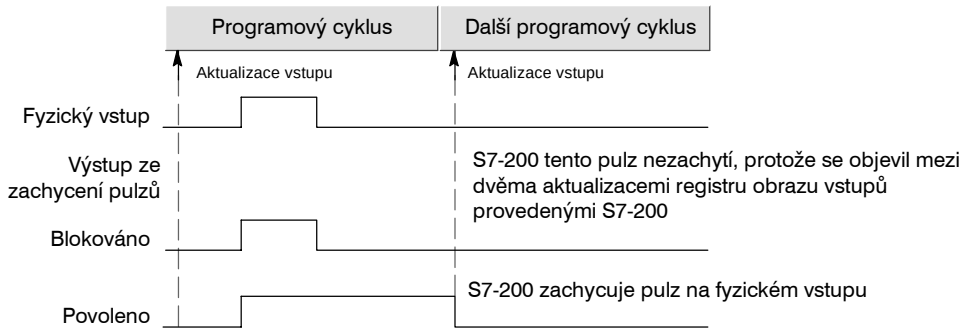
Pro zobrazení konfigurace zachycení pulzů:

1. Vyberte příkaz menu **Zobrazit> Komponenty> Systémový blok** a zvolte “Bity pro zachycení pulzů”.
2. Klikněte na odpovídající zaškrtávací okénko a potvrďte na OK.
3. Proveďte download modifikovaného systémového bloku do S7-200.



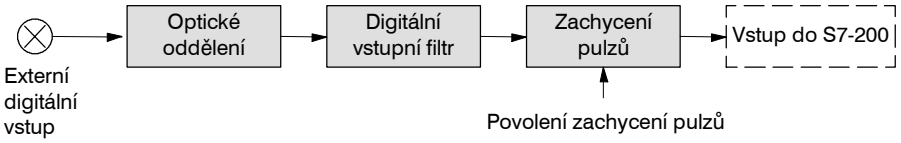
Obr. 4-24 Zachycení pulzů

Obrázek 4-25 ukazuje základní princip práce S7-200 s povolenou a blokovanou funkcí zachycení pulzů.



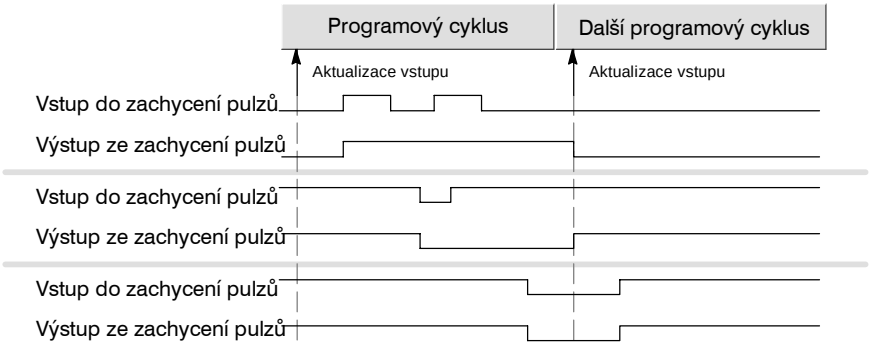
Obr. 4-25 Chod automatu S7-200 s povolenou a blokovanou funkcí zachycení pulzů

Protože funkce zachycování pulzů následuje až po vstupní filtraci, je třeba přizpůsobit čas vstupního filtru tak, aby pulz nebyl filtrem odstraněn. Obrázek 4-26 ukazuje blokové schéma vstupního digitálního obvodu.



Obr. 4-26 Digitální vstupní obvod

Obrázek 4-27 znázorňuje odezvu povolené funkce zachycení pulzů na různé vstupní podmínky. Pokud se objeví v době programového cyklu více než jeden pulz, zachytí funkce zachycení pulzů jen pulz první. Pokud je v programovém cyklu více pulzů, je třeba naprogramovat obsluhu přerušení spouštěnou náběžnou/sestupnou hranou. (Seznam událostí přerušení je v tabulce 6-46.)



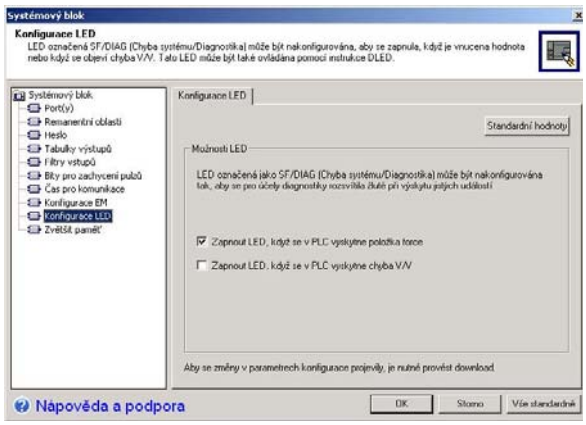
Obr. 4-27 Odezvy funkce zachycení pulzů na různé vstupní podmínky

S7-200 poskytuje uživatelsky řízenou kontrolku LED

S7-200 poskytuje kontrolku (SF/DIAG), která může svítit červeně (kontrolka poruchy systému) nebo žlutě (diagnostická kontrolka). Diagnostická kontrolka může svítit při řízení uživatelského programu nebo se může za určitých podmínek automaticky rozsvítit: když je vnucována hodnota vstup/výstup nebo data nebo když je na modulu chyba vstupu/výstupu.

Konfigurace automatických voleb pro diagnostickou kontrolku:

1. Vyberte příkaz menu **Zobrazit> Komponenty> Systémový blok** a zvolte “Konfigurovat kontrolku”.
2. Klikněte na každou položku a aktivujte nebo deaktivujte zapnutí kontrolky, když je vstup/výstup nebo hodnota dat vnucována, nebo když má modul chybu vstupu/výstupu.
3. Proveďte download modifikovaného systémového bloku do S7-200.



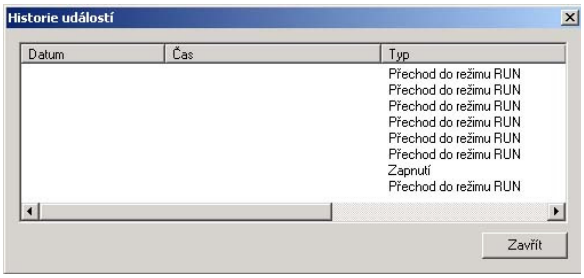
Obr. 4-28 Diagnostická kontrolka

K řízení stavu diagnostické kontrolky u vašeho uživatelského programu použijte instrukci diagnostické kontrolky v kapitole 6.

S7-200 udržuje záznam historie důležitých událostí CPU

S7-200 udržuje záznam, který obsahuje historii s údajem času o hlavních událostech CPU jako spuštění napájení, vstup CPU do režimu RUN a vznik kritických chyb. Musí být nastaveny hodiny reálného času, aby byl v zápisech v protokolu správný čas a datum.

Pro zobrazení záznamu historie událostí zvolte příkaz menu **PLC > Informace** a zvolte "Historie událostí".



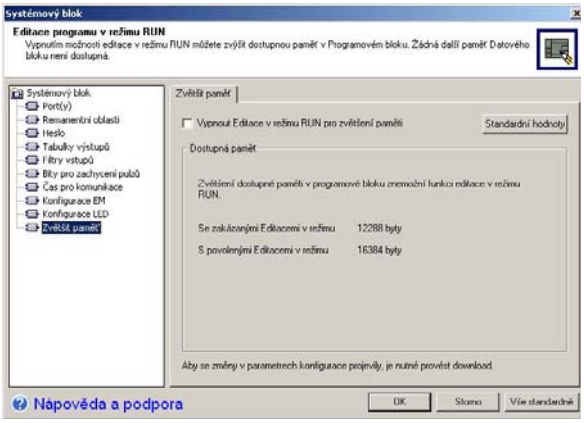
Obr. 4-29 Prohlížení záznamu historie událostí

S7-200 vám umožní zvýšit dostupnou paměť pro uživatelský program

S7-200 vám umožní vypnout funkci editace režimu RUN v CPU 224, CPU 224XP a CPU 226, aby se zvýšila velikost programové paměti, kterou můžete použít. V tabulce 1-2 je velikost paměti pro každý model CPU.

Postup vypnutí funkce editace režimu RUN

1. Zvolte příkaz menu **Zobrazit > Systémový blok** a zvolte "Zvýšit paměť".
2. Kliknutím na položku "Zvýšit paměť" vypnete funkci editace režimu RUN.
3. Proveďte download modifikovaného systémového bloku do S7-200.



Obr. 4-30 Vypnout editaci režimu RUN

Ochrana heslem

Všechny modely S7-200 mají ochranu heslem pro omezení přístupu ke konkrétním funkcím.

Heslo opravňuje k přístupu k funkcím a paměti: bez použití hesla je přístup do S7-200 neomezený. Když se použije ochrana heslem, jsou omezeny všechny operace podle konfigurace při programování hesla.

Heslo nerozlišuje malá a velká písmena.

Jak je patrné z tabulky 4-3, S7-200 má tři úrovně omezení přístupu. Každá úroveň umožňuje přístup k určitým funkcím i bez hesla. Pro všechny tři úrovně platí, že po zadání správného hesla je přístup neomezený. Implicitně je S7-200 nastaven na úroveň 1 (bez omezení).

Zadáním hesla se ochrana S7-200 heslem neodstraní.

Tabulka 4-3 Omezení přístupu do S7-200

Funkce CPU	Úroveň 1	Úroveň 2	Úroveň 3
Čtení a zápis uživatelských dat	Přístup povolen	Přístup povolen	Přístup povolen
Spuštění, zastavení a restart CPU			
Čtení a zápis hodin reálného času			
Aktualizace uživatelského programu, dat a konfigurace CPU	Přístup povolen	Přístup povolen	Požad. heslo
Nahrávání programu do CPU (download)	Přístup povolen	Požad. heslo	
Stav STL			
Vymazání programového, datového nebo systémového bloku			
Vnucení hodnoty (force) nebo provedení jediného či více programových cyklů			
Kopírování do paměťového modulu			
Zápis výstupů v režimu STOP			

Oprávnění přístupu daného uživatele k omezeným funkcím neopravňuje jiné uživatele přistupovat k těmto funkcím. V daném čase může mít neomezený přístup k S7-200 pouze jeden uživatel.

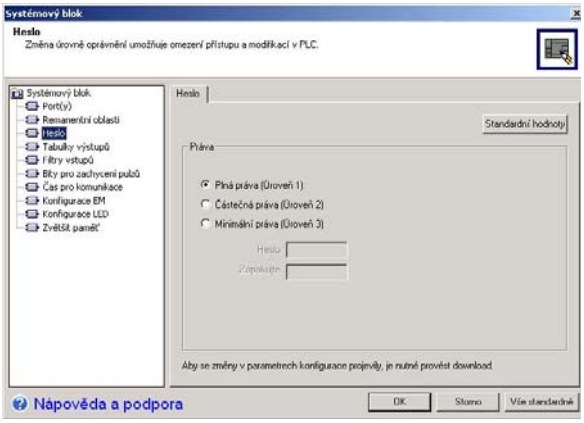
Tip

Po zadání hesla zůstává úroveň oprávnění pro toto heslo v platnosti až jednu minutu po odpojení počítače od S7-200. Vždy ukončete program STEP 7-Micro/WIN před odpojením kabelu, abyste zabránili jinému uživateli v přístupu k programu.

Naprogramování hesla pro S7-200

Dialogové okno “Systémový blok” (obrázek 4-31) vám umožní konfigurovat heslo pro S7-200. Implicitní stav pro S7-200 je nastaven na “Plný” (stupeň 1), bez omezení.

1. Vyberte příkaz menu **Zobrazit > Komponenty > Systémový blok**; zobrazí se dialogové okno “Systémový blok”. Pak klikněte na “Heslo”.
2. Vyberte příslušnou úroveň přístupu do S7-200.
3. Zadejte a ověřte heslo pro Částečný (stupeň 2) nebo Minimální (stupeň 3) přístup.
4. Klikněte na OK.
5. Proveďte download modifikovaného systémového bloku do S7-200.



Obr. 4-31 Vytvoření hesla

Obnovení ztraceného hesla

Pokud zapomenete heslo, je třeba smazat paměť S7-200 a opět do ní nahrát program. Smazáním paměti se S7-200 dostane do režimu STOP, je nastaveno na své implicitní hodnoty, nastavené ve výrobě, s výjimkou síťové adresy, přenosové rychlosti a hodin reálného času. Chcete–li vymazat program v automatu S7-200:

1. Vyberte příkaz menu **PLC > Smazat** a objeví se dialogové okno “Smazat”.
2. Vyberte všechny tři bloky a svoji činnost potvrďte kliknutím na OK.
3. Pokud bylo naprogramováno heslo, STEP 7–Micro/WIN zobrazí dialogové okno hesla pro oprávnění k přístupu. Pro vymazání hesla zadejte CLEARPLC v dialogovém okně pro heslo a operace smazání pokračuje. (Heslo CLEARPLC nerozlišuje malá a velká písmena.)

Operace smazání neodstraní program z paměťového modulu. Protože se program do paměťového modulu ukládá spolu s heslem, je nutné přeprogramovat i paměťový modul, a odstranit tak heslo i z něj.



Varování

Po vymazání paměti S7-200 se výstupy nastaví na hodnotu log. 0 (nebo v případě analogových výstupů zůstanou zmrazené na zadané hodnotě).

Pokud je automat S7-200 připojený k řízenému zařízení v době mazání paměti, změny stavu výstupů mohou být přeneseny na řízené zařízení. Pokud jste nakonfigurovali ”bezpečný stav” výstupů jinak, než bylo zadáno ve výrobě, změny na výstupech mohou zapříčinit nepředvídatelný stav zařízení, což může způsobit smrt nebo vážné zranění osob a poškození zařízení.

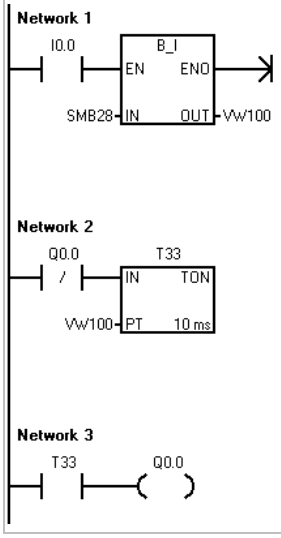
Vždy dodržujte příslušná bezpečnostní opatření. Před vymazáním paměti S7-200 zajistěte, aby byl celý proces v bezpečném stavu.

Analogové potenciometry

Analogové potenciometry jsou umístěny pod předním krytem modulu. Těmito potenciometry můžete zvýšit nebo snížit hodnoty uložené v bytech speciální paměti (SMB). Hodnoty se dají pouze číst, v programu mohou být použity pro různé funkce, např. pro nastavení hodnot časovače nebo čítače, pro zadávání nebo změnu přednastavených hodnot nebo pro nastavování limitů. Pro nastavení potenciometru použijte malý šroubovák: pro zvýšení hodnoty otáčejte potenciometrem ve směru hodinových ručiček (doprava), pro snížení hodnoty proti směru hodinových ručiček (doleva).

SMB28 obsahuje číselnou hodnotu, která reprezentuje polohu analogového potenciometru 0. SMB29 obsahuje číselnou hodnotu, která reprezentuje polohu analogového potenciometru 1. Analogové hodnota má nominální rozsah 0 až 255 a opakovatelnost ∂2.

Ukázkový program použití hodnoty vložené analogovými potenciometry




```
Network 1   //Načte hodnotu analogového potenciometru 0
            //(SMB28).
            //Uloží tuto hodnotu jako celé číslo do VW100.
LD    I0.0
BTI   SMB28, VW100

Network 2   //Použije celočíselnou hodnotu (VW100) jako
            //přednastavení pro časovač.
LDN   Q0.0
TON   T33, VW100

Network 3   //Nastaví Q0.0, když T33 dosáhne
            //nastavené hodnoty.
LD    T33
=     Q0.0
```

Vysokorychlostní vstupy a výstupy

Vysokorychlostní čítače

Automat S7-200 má integrované vysokorychlostní čítače, které dokáží počítat externí události s vysokou rychlostí bez omezení činnosti S7-200. Rychlosti čítačů pro daný typ CPU najdete v příloze A. Každý čítač má vstupy vyhrazené pro hodinové pulzy, řízení směru, nulování a start, pokud tyto funkce podporuje. Pro změnu rychlosti čítače je možné vybrat různé kvadraturní režimy. Více informací o vysokorychlostních čítačích najdete v kapitole 6.

Vysokorychlostní pulzní výstup

Programovatelný automat S7-200 podporuje vysokorychlostní pulzní výstupy Q0.0 a Q0.1, které mohou generovat sérii rychlých výstupních pulzů (PTO) nebo pulzní šířkovou modulaci (PWM).

Funkce PTO generuje výstup obdélníkové vlny (50% střídou) pro určitý počet pulzů (od 1 do 4.294.967.295 pulzů) a určitou dobu cyklu (v mikrosekundových nebo milisekundových přírůstcích). Funkce PTO může být naprogramována tak, aby generovala buď jednu sérii pulzů, nebo pulzní profil s více sériemi pulzů. Pulzní profil je například možné použít pro řízení krokového motoru prostřednictvím jednoduché posloupnosti rozběh, chod, doběh nebo prostřednictvím složitějších posloupností.

Funkce PWM má pevnou frekvenci pulzů s různou šířkou pulzu a mezery; perioda a šířka pulzu se zadává v mikrosekundách nebo v milisekundách. Když je šířka pulzu rovná době cyklu, střída je 100 % a výstup je trvale zapnut. Jestliže je šířka pulzu nulová, tvoří pulz 0 % periody a výstup je stále vypnutý.

Více informací o vysokorychlostních výstupech najdete v kapitole 6. Další informace o použití PTO v řízení pohybu v otevřené smyčce viz kapitola 9.

Pojmy, konvence a možnosti programování

5

Programovatelný automat S7-200 nepřetržitě vykonává uživatelský program, aby řídil úlohu nebo proces. Pro vytvoření programu a jeho downloadu do S7-200 se používá STEP 7–Micro/WIN, který obsahuje mnoho různých nástrojů a funkcí pro návrh, implementaci a ladění programu.

Přehled kapitoly

Návod pro návrh systému 52
Základní prvky programu 53
Použití STEP 7–Micro/WIN pro vytvoření programu 55
Volba mezi instrukčním souborem SIMATIC a IEC 1131–3 57
Vysvětlení konvencí použitých v programových editorech 58
Použití průvodců jako pomůcky pro tvorbu řídícího programu 60
Ošetření chyb S7-200 60
Přiřazení adres a počátečních hodnot v editoru datového bloku 62
Použití tabulky symbolů pro symbolické adresování proměnných 62
Použití lokálních proměnných 63
Použití stavové diagramu pro monitorování uživatelského programu 63
Vytvoření knihovny instrukcí 64
Funkce pro adaptaci uživatelského programu 64

Návod pro návrh systému

Existuje mnoho způsobů, jak navrhnout mikro – PLC systém. Následující obecné pokyny se dají aplikovat na mnoho konstrukčních projektů. Musíte se pochopitelně řídit směrnicemi vaší společnosti a zavedenou praxí, danou vašimi zkušenostmi a působištěm.

Rozdělení procesu nebo stroje

Rozdělte proces nebo stroj, který chcete řídit, na části s jistou úrovní vzájemné nezávislosti. Toto rozdělení určí hranice mezi řídicími systémy a ovlivní funkčního specifikaci a přiřazení prostředků.

Vytvoření funkční specifikace

Popište operace každé části procesu nebo stroje. Zahrňte následující témata: vstupní a výstupní body; funkční popis provozu; stavy, kterých musí být dosaženo, aby byla povolena činnost každého akčního členu (jako jsou solenoidy, motory a pohony); popis operátorského rozhraní a dalších rozhraní s ostatními částmi procesu nebo stroje.

Návrh bezpečnostních obvodů

Specifikujte zařízení, které kvůli bezpečnosti vyžaduje pevně zapojenou logiku. Řídicí zařízení mohou selhat a způsobit nečekané spuštění nebo změnu stavu řízených strojů. Tam, kde by neočekávaný nebo nesprávný provoz strojů mohl mít za následek zranění obsluhy nebo významné majetkové škody, by se měly použít elektromechanické ochrany pracující nezávisle na S7-200, zabrání se tak nebezpečným operacím. Do návrhu bezpečnostních obvodů by měly být zahrnuty následující úlohy:

- Specifikujte nesprávné nebo neočekávané operace akčních členů, které by mohly být nebezpečné.
- Specifikujte podmínky, které vás ujistí o tom, že provoz není nebezpečný, a určete, jak tyto podmínky detekovat nezávisle na S7-200.
- Specifikujte, jak při připojení nebo odpojení napájení a při zjištění chyb ovlivní CPU S7-200 a jeho vstupy a výstupy proces. Tyto informace by se měly použít pouze při projektování pro normální a předpokládaný mimořádný provoz; nemělo by se na ně spoléhat pro bezpečnostní účely.
- Navrhněte manuální nebo elektromechanické ochrany, které zablokují nebezpečné operace nezávisle na S7-200.
- Pro S7-200 zajistěte příslušné stavové informace z nezávislých obvodů tak, aby program a kterékoliv operátorské rozhraní byly dostatečně informovány.
- Specifikujte jakékoliv jiné požadavky, vztahující se k bezpečnosti procesu.

Specifikace operátorských stanic

Na základě požadavků funkční specifikace nakreslete výkresy operátorských stanic. Zahrňte následující položky:

- Přehled umístění každé operátorské stanice ve vztahu k procesu nebo stroji
- Mechanické uspořádání zařízení na operátorské stanici, jako jsou displeje, přepínače a signálky.
- Elektrotechnická schémata s příslušnými vstupy a výstupy CPU S7-200 a rozšiřovacími moduly.

Tvorba schémat konfigurace

Na základě požadavků funkční specifikace nakreslete schémata konfigurace řídícího zařízení. Zahrňte následující položky:

- ❑ Přehled umístění každého automatu ve vztahu k procesu nebo stroji
- ❑ Mechanické uspořádání jednotek S7-200 a rozšiřujících I/O modulů (včetně rozvaděčů a dalšího zařízení)
- ❑ Elektrotechnická schémata pro každý automat a rozšiřující I/O modul (včetně čísla modelu přístroje, komunikační adresy a adres vstupů a výstupů)

Vytvoření seznamu symbolických názvů (volitelné)

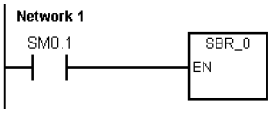
Pokud chcete používat pro adresování symbolická názvy, vypracujte jejich seznam pro absolutní adresy. Zahrňte nejen fyzické I/O signálky, ale i jiné prvky, které mají být použity v uživatelském programu.

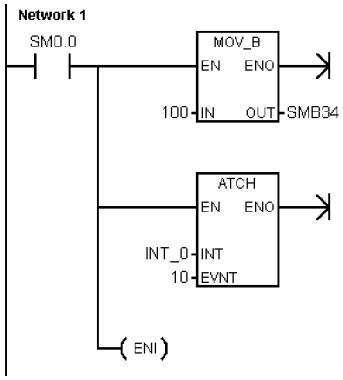
Základní prvky programu

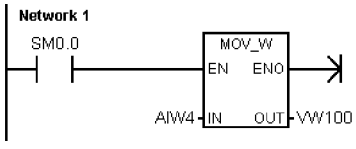
Programový blok se skládá ze spustitelného kódu a komentáře. Spustitelný kód se skládá z hlavního programu a dalších podprogramů nebo přerušení. Kód je kompilován a nahrán do S7-200; komentář k programu nikoliv. Pro zpřehlednění řídicího programu můžete použít organizační prvky (hlavní program, podprogramy a přerušení).

Následující příklad ukazuje program, který obsahuje hlavní program a podprogram přerušení. Tento vzorový program používá časové přerušení pro odečítání hodnoty analogového vstupu každých 100 ms.

Příklad: Základní prvky programu

MAIN		Network 1 //V prvním cyklu volá podprogram 0. LD SM0.1 CALL SBR_0

SBR0		Network 1 //Nastaví interval na 100 ms
		//pro časové přerušení.
		//Povolí přerušení 0.
		LD SM0.0
		MOVB 100, SMB34
		ATCH INT_0, 10
		ENI

INT0		Network 1 //Vzorkuje analogový vstup 4. LD SM0.0 MOVW AIW4,VW100

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.



Tip

Použití adres v paměti V může omezit přenositelnost podprogramu, protože je možné, že přiřazení adres paměti V jednoho programu může kolidovat s přiřazením v jiném programu. Naproti tomu podprogramy, které pro všechna přiřazení používají tabulku lokálních proměnných (L paměť), jsou snadno přenositelné, protože při použití lokálních proměnných nevzniknou konflikty adres mezi podprogramem a jinou částí programu.

Podprogramy přerušení

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

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



Tip

Protože není možné předvídat, kdy automat generuje přerušení, je žádoucí omezit počet proměnných, které se používají jak v podprogramu přerušení, tak kdekoliv v hlavním programu.

Abyste zajistili, že přerušení používá pouze dočasnou paměť a nepřepisuje data používaná jinde v programu, používejte v podprogramu přerušení tabulku lokálních proměnných.

Existuje mnoho programovacích technik, které můžete použít, abyste zajistili správné sdílení dat mezi hlavním programem a podprogramy přerušení. Tyto techniky jsou popsány v kapitole 6 spolu s instrukcemi přerušení.

Ostatní prvky programu

Ostatní bloky obsahují informace pro S7-200. Download těchto bloků můžete 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 můžete použít pro vložení počátečních hodnot dat.

Použití STEP 7–Micro/WIN pro vytvoření programu

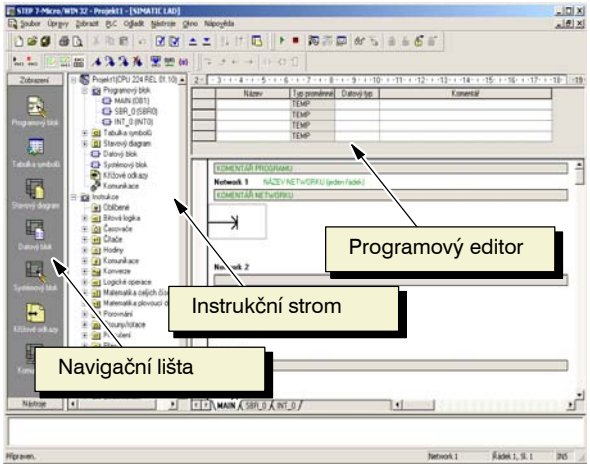
Chcete–li otevřít STEP 7–Micro/WIN, dvakrát klikněte na ikonu STEP 7–Micro/WIN nebo vyberte příkaz menu **Start > SIMATIC > STEP 7 MicroWIN 32 V4.0**. Jak je vidět na obrázku 5-1, okno projektu STEP 7–Micro/WIN nabízí přehledné pracovní prostředí pro vytváření řídicího programu.

Nástrojové lišty obsahují tlačítka pro zkratky k často používaným příkazům menu. Všechny nástrojové lišty můžete 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 můžete do programu vložit metodou drag&drop nebo můžete na instrukci dvakrát kliknout, čímž ji vložíte na současnou pozici kurzoru v programovém editoru.

Programový editor obsahuje program a tabulku lokálních proměnných, ve které můžete přiřadit symbolické názvy dočasným lokálním proměnným. Podprogramy a přerušení jsou zobrazeny jako záložky ve spodní části okna programového editoru. Chcete–li přecházet mezi podprogramy, přerušeními a hlavním programem, klikněte na příslušnou záložku.



Obr. 5-1 STEP 7–Micro/WIN


Programový editor

STEP 7–Micro/WIN obsahuje tři editory pro vytváření uživatelského programu: kontaktní schémata (LAD), výpis příkazů (STL) a 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 programujete v jazyku S7-200 a nikoli v jazyku grafického editoru, kde platí určitá omezení, aby byly diagramy správně nakresleny. Jak je vidět na obrázku 5-2, je tato znakově orientovaná koncepce velmi podobná programování ve strojovém kódu.

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.

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

Obr. 5-2 Příklad programu STL

Jestliže volíte editor STL, vezměte 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.

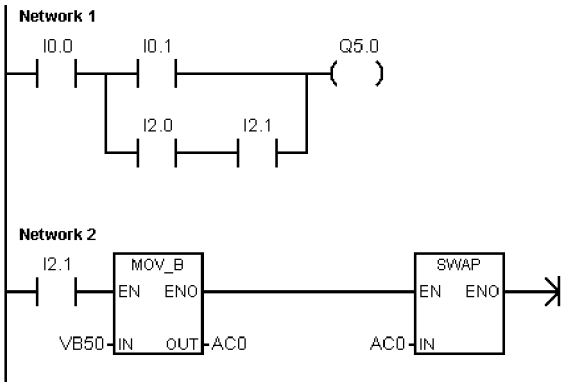
Vlastnosti editoru LAD

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 5-3 ukazuje 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.



Obr. 5-3 Příklad programu LAD

Bloky představují doplňující instrukce, jako jsou časovače, čítače nebo matematické instrukce.

Jestliže volíte editor LAD, vezměte v úvahu tyto hlavní body:

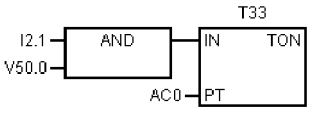
- Kontaktní schémata snadno používají i programátoři začátečníci.
- Grafické zobrazení se snadno chápe a je populární na celém světě.
- Editor LAD je možné použít s instrukčním souborem SIMATIC i IEC 1131–3.
- Vždy můžete použít editor STL pro zobrazení programu vytvořeného editorem SIMATIC LAD.

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 5-4 ukazuje 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.



Obr. 5-4 Příklad programu 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 zvolíte pro programování editor FBD, vezměte 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.

Volba mezi instrukčním souborem SIMATIC a IEC 1131–3

Většina programovatelných automatů nabízí podobné základní instrukce, obvykle jsou ale mezi jednotlivými výrobci drobné rozdíly ve vzhledu, provozu apod. V průběhu posledních několika let vypracovala Mezinárodní elektrotechnická asociace (IEC) globální normu, která se speciálně vztahuje na mnoho aspektů programování automatů. Tato norma doporučuje různým výrobcům programovatelných automatů, aby nabízeli instrukce, které jsou stejné jak svým vzhledem, tak funkcí.

Automat S7-200 obsahuje dva instrukční soubory, které umožní řešit velké množství různých automatizačních úloh. Instrukční soubor IEC odpovídá normě IEC 1131–3 pro programování PLC a instrukční soubor SIMATIC je navržen speciálně pro automat S7-200.



Tip

Pokud je program STEP 7–Micro/WIN nastaven do režimu IEC, zobrazí se v Instrukčním stromu vedle instrukcí, které nejsou definovány normou IEC 1131–3, červený kosočtverec ( ).

Mezi instrukčními soubory SIMATIC a IEC existuje jen několik klíčových rozdílů:

- Instrukční soubor IEC se omezuje na ty instrukce, které jsou mezi prodejci programovatelných automatů standardní. Některé instrukce, které jsou běžně zahrnuty do souboru SIMATIC, nejsou standardními instrukcemi ze specifikace IEC 1131–3. Jsou stále k dispozici pro užívání jako nestandardní instrukce; pokud je, ale použijete, není program nadále kompatibilní s IEC 1131–3.
- Některé blokové instrukce IEC akceptují více datových formátů. Této praxi se často říká vícenásobná definice (overloading). Například místo matematických bloků ADD_I (přičtení celého čísla) a ADD_R (přičtení reálného čísla) přezkoumá IEC instrukce ADD formát přičítaných dat a automaticky zvolí správnou instrukci S7-200. Tím se dá ušetřit cenný čas při navrhování programu.
- Když použijete instrukce IEC, jsou parametry instrukcí automaticky kontrolovány na správný datový formát, jako například celé číslo se znaménkem (signed integer) nebo celé číslo bez znaménka (unsigned integer). Jestliže se například pokusíte provést s hodnotou integer instrukci, která očekává bitovou hodnotu (zapnuto/vypnuto), je výsledkem chybové hlášení. Tato vlastnost pomáhá minimalizovat syntaktické chyby programu.

Jestliže volíte mezi instrukčním souborem SIMATIC nebo IEC, vezměte v úvahu následující body:

- Instrukce SIMATIC mají obvykle kratší dobu zpracování. Některé instrukce IEC mohou mít delší dobu zpracování.
- Některé instrukce IEC, například instrukce časovačů, čítačů, násobení a dělení se chovají jinak než jejich ekvivalenty SIMATIC.
- S instrukčním souborem SIMATIC můžete použít všechny tři programové editory (LAD, STL, FBD). Pro instrukce IEC můžete použít pouze programové editory LAD a FBD.
- Funkce instrukcí IEC je standardní pro různé značky programovatelných automatů a znalost tvorby programu odpovídajícího IEC je podporována platformami různých programovatelných automatů.
- Norma IEC sice definuje méně instrukcí, než nabízí instrukční soubor SIMATIC, do programu IEC je ale vždy možné zahrnout instrukce SIMATIC.
- Norma IEC 1131–3 vyžaduje přiřazení typu k proměnným a podporuje systémovou kontrolu datového typu.

Vysvětlení konvencí použitých v programových editorech

Ve všech programových editorech používá STEP 7–Micro/WIN následující konvence:

- # před jménem symbolu (#var1) znamená, že symbol má lokální platnost.
- V instrukcích IEC označuje symbol % přímou adresu.
- Symbol operandu "?.?" nebo "????" znamená, že se požaduje zadání operandu.

Programy LAD jsou rozděleny do segmentů nazývaných networky. Network je uspořádaná sestava kontaktů, cívek a bloků, které jsou propojeny tak, že se tvoří úplný obvod: neexistují v něm zkraty, přerušení obvodu ani podmínky zpětného signálového toku. STEP 7–Micro/WIN umožňuje tvorbu komentáře pro program LAD systémem jeden network za druhým. Při programování editorem FBD se pojem networku používá pro rozdělení programu na segmenty a vypracování komentáře.

Programy STL nepoužívají networky; můžete ale použít klíčové slovo NETWORK pro rozdělení programu na segmenty.

Konvence charakteristické pro editor LAD

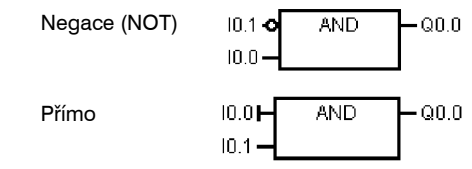
V editoru LAD můžete použít klávesy F4, F6 a F9 klávesnice pro přístup k instrukcím kontaktu, bloku a cívek. Editor LAD používá následující konvence:

- Symbol "– – – >>" znamená přerušený obvod nebo požadovaný přípoj signálového toku.
- Symbol "→|" znamená, že výstup je volitelná součást signálového toku pro instrukci, která může být uspořádána kaskádovitě nebo zapojena sériově.
- Symbol ">>" znamená, že můžete použít signálový tok.

Konvence charakteristické pro editor FBD

V editoru FBD můžete použít klávesy F4, F6 a F9 klávesnice pro přístup k instrukcím AND, OR a blokovým instrukcím. Editor FBD používá následující konvence:

- Symbol "– – – >>" na operandu EN je signálový tok nebo indikátor operandu. Může také znamenat přerušený obvod nebo požadovaný přípoj signálového toku.
- Symbol "→|" znamená, že výstup je volitelná součást signálového toku pro instrukci, která může být uspořádána kaskádovitě nebo zapojena sériově.
- Symboly "<<" a ">>" znamenají, že můžete použít buď hodnotu, nebo signálový tok.
- Negace: negace nebo inverze operandu nebo signálového toku se označují malým kroužkem na vstupu. Na obrázku 5-5 se Q0.0 rovná negaci logického součinu I0.0 a I0.1 (NOT I0.0 AND I0.1). Negace platí pouze pro booleovské signály, které mohou být specifikovány jako parametry nebo signálový tok.



Obr. 5-5 Konvence FBD

- Indikátory okamžitého stavu: Jak je vidět na obrázku 5-5, editor FBD zobrazuje okamžitý stav booleovského operandu svislou čarou na vstupu pro instrukci FBD. Indikátor okamžitého stavu způsobí okamžité načtení ze specifikovaného fyzického vstupu. Okamžité operátory platí pouze pro fyzické vstupy.
- Blok bez vstupu nebo výstupu: Blok bez vstupu označuje instrukci, která je nezávislá na signálovém toku.

Tip

Počet operandů pro instrukce AND a OR je možné rozšířit až na 32 vstupů. Pro zvětšení nebo zmenšení počtu operandů použijte klávesy "+" a "–" na klávesnici.

Obecné konvence pro programování automatu S7-200

Definování EN/ENO

EN (Enable IN) je booleovský vstup bloků v LAD a FBD. Aby byla bloková instrukce provedena, musí být na tomto vstupu přítomen signálový tok. V editoru STL nemají instrukce vstup EN; aby byla odpovídající instrukce provedena, musí být hodnota vrcholu zásobníku logická ”1”.

ENO (Enable Out) je booleovský výstup bloků v LAD a FBD. Pokud má blok signálový tok na vstupu EN a provede svou funkci bez chyby, výstup ENO předá signálový tok dalšímu prvku. Pokud je při provádění v bloku zjištěna chyba, je signálový tok přerušen na bloku, který chybu generoval.

V editoru STL výstup ENO neexistuje, ale instrukce v STL, které odpovídají instrukcím LAD a FBD s výstupy ENO, nastaví speciální bit ENO. K tomuto bitu se přistupuje pomocí instrukce AND ENO (AENO) a je možné ho použít stejně jako výstup bloků ENO.



Tip

Operandy a datové typy EN/ENO nejsou uvedeny v tabulce platných operandů u každé instrukce, protože tyto operandy pro všechny instrukce LAD a FBD stejné. Tabulka 5-1 uvádí tyto operandy a datové typy pro LAD a FBD. Tyto operandy platí pro všechny instrukce LAD a FBD, uvedené v této příručce.

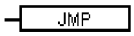
Tabulka 5-1 Operandy a datové typy EN/ENO pro LAD a FBD

Programový editor	Vstupy/Výstupy	Operandy	Typy dat
LAD	EN, ENO	Signálový tok	BOOL
FBD	EN, ENO	I, Q, V, M, SM, S, T, C, L	BOOL

Podmíněné/nepodmíněné vstupy

V LAD a FBD jsou bloky nebo cívky, které jsou závislé na signálovém toku, spojené s libovolným prvkem na levé straně. Cívky a bloky, které na signálovém toku nezávisí, jsou připojené přímo k levé napájecí liště. Tabulka 5-2 ukazuje příklad podmíněného i nepodmíněného vstupu.

Tabulka 5-2 Zobrazení podmíněných a nepodmíněných vstupů

Signálový tok	LAD	FBD
Instrukce závisející na signálovém toku (podmíněná)		
Instrukce nezávisející na signálovém toku (nepodmíněná)		

Instrukce bez výstupů

Bloky, které nemohou být řazeny do kaskády, se zobrazují bez booleovských výstupů. Patří mezi ně instrukce ”Volání podprogramu”, ”Skok” a ” Podmíněné ukončení programu”. Existují také instrukce, které je možné umístit pouze na levou napájecí lištu. Patří mezi ně instrukce ”Návěští”, ”Next”, ”Načíst SCR”, ”Podmíněný konec SCR” a ”Konec SCR”. Ty jsou v FBD zobrazeny jako bloky; poznají se podle neoznačených výkonových vstupů a podle toho, že nemají výstupy.

Instrukce porovnávání

Instrukce porovnávání se provádí bez ohledu na stav signálového toku. Pokud není signálový tok, má výstup hodnotu logická 0. Pokud je signálový tok přítomen, je výstup nastaven podle výsledku porovnání. Instrukce porovnání SIMATIC FBD, IEC Ladder a IEC FBD jsou zobrazovány jako bloky, přestože se operace provádějí jako kontakt.

Použití průvodců jako pomůcky pro tvorbu řídicího programu

STEP 7–Micro/WIN obsahuje průvodce, které usnadňují jednotlivé aspekty programování a automatizují je. V kapitole 6 jsou instrukce, které mají svého průvodce, označeny ikonou:



Průvodce

Ošetření chyb S7-200

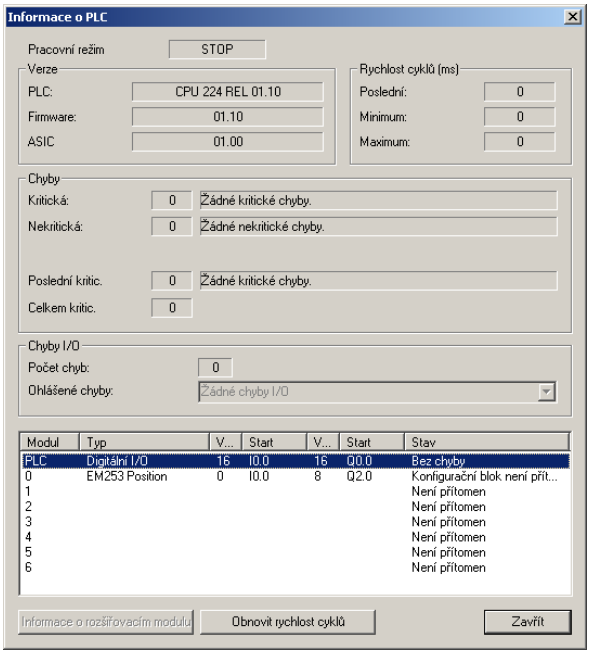
Automat S7-200 dělí chyby na chyby kritické a nekritické. Chybové kódy, generované vzniklou chybou, si můžete prohlédnout pomocí příkazu menu **PLC > Informace**.

Obrázek 5-6 ukazuje dialogové okno ”Informace o PLC”, kde je zobrazen chybový kód a popis chyby.

Políčko ”Poslední kritická” ukazuje poslední kód kritické chyby, generovaný S7-200. Tato hodnota se uchová i po výpadku napájení, pokud je uchována paměť RAM. Je však vymazána při smazání celé paměti S7-200 nebo když se RAM vymaže v důsledku dlouhodobého výpadku napájení.

Políčko ”Celkem kritických”obsahuje počet kritických chyb, generovaných od posledního smazání všech paměťových oblastí S7-200. Tato hodnota se uchovává při výpadku a obnovení napájení, pokud je uchována paměť RAM. Je však vymazána buď při smazání celé paměti S7-200, nebo když se RAM vymaže v důsledku dlouhodobého výpadku napájení.

V příloze C jsou uvedeny chybové kódy S7-200; v příloze D jsou popsány bity speciální paměti (SM), které je možné použít pro monitorování chyb.



Obr. 5-6 Dialogové okno ”Informace o PLC ”

Nekritické chyby

Nekritické chyby indikují problémy v konstrukci uživatelského programu, problémy s prováděním instrukcí v uživatelském programu a problémy s rozšiřovacími vstupními/výstupními moduly. Pro prohlédnutí chybových kódů, které byly generovány nekritickou chybou, můžete použít STEP 7–Micro/WIN. Existují tři základní kategorie nekritických chyb.

Chyby vzniklé při překladu programu

Automat S7-200 překládá program při jeho downloadu. Pokud S7-200 zjistí, že program porušuje pravidla překladu, je download přerušen a generuje se chybový kód. (Program, který už byl načten do S7-200, stále existuje v trvalé paměti a není ztracen.) Program nejprve opravte; pak můžete opět provést jeho download. V příloze C najdete seznam porušení pravidel překladu.

Chyby vstupů a výstupů

Při prvním spuštění načte S7-200 z každého modulu konfiguraci vstupů a výstupů. Během normálního provozu S7-200 periodicky kontroluje stav každého modulu a porovnává ho s konfigurací získanou při prvním spuštění. Pokud automat S7-200 zjistí rozdíl, nastaví chybový bit konfigurace v chybovém registru modulu. S7-200 z tohoto modulu nenačítá vstupní data ani do něj nezapisuje výstupní data, dokud konfigurace modulu opět neodpovídá konfiguraci získané při prvním spuštění.

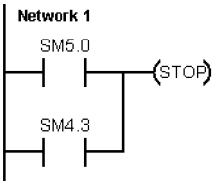
Informace o stavu modulu jsou uloženy ve speciálních paměťových bitech (SM). Uživatelský program dokáže tyto bity monitorovat a vyhodnocovat. Více informací o bitech SM, používaných pro hlášení chyb vstupů a výstupů, najdete v příloze D. SM5.0 je globální bit pro chyby vstupů a výstupů a zůstává nastaven, dokud chybová podmínka v rozšiřovacím modulu trvá.

Chyby provádění programu

Uživatelský program dokáže během svého provádění vytvořit chybové podmínky. Takové chyby mohou být důsledkem nesprávného použití instrukce nebo toho, že instrukce zpracovává neplatná data. Například ukazatel na nepřímou adresu, který byl platný v době překladu programu, může být během provádění programu modifikován, takže ukazuje na adresu mimo rozsah. To je příklad problému programování při běhu programu. SM4.3 je nastaven při výskytu problému programování v době běhu programu a zůstává nastaven, dokud je S7-200 v režimu RUN. (Seznam problémů programování při běhu programu najdete v příloze C.) Informace o chybách provádění programu jsou uloženy ve speciálních paměťových bitech (SM). Uživatelský program dokáže tyto bity monitorovat a vyhodnocovat. Více informací o bitech SM, používaných pro hlášení chyb provádění programu, najdete v příloze D.

Jestliže automat S7-200 zjistí nekritickou chybu, nepřejde do režimu STOP. Pouze tuto událost zaznamená do paměti SM a pokračuje v provádění uživatelského programu. Uživatelský program můžete navrhnout ale tak, aby byl S7-200 nucen přejít do režimu STOP, když je zjištěna nekritická chyba. Následující vzorový program ukazuje network programu, který monitoruje dva globální nekritické chybové bity a uvede S7-200 do režimu STOP, kdykoliv bude mít některý z těchto bitů hodnotu logická 1.

Vzorový program: Logika pro zjištění nekritické chybové podmínky




```
Network 1      //Jestliže dojde k chybě vstupu/výstupu nebo k chybě
               //v době běhu programu, přejde do režimu STOP
LD    SM5.0
O     SM4.3
STOP
```

Kritické chyby

Kritické chyby způsobí, že automat zastaví provádění uživatelského programu. Kritická chyba může podle své závažnosti způsobit, že S7-200 nebude moci provádět některé nebo všechny své funkce. Cílem ošetření kritických chyb je přivést automat S7-200 do bezpečného stavu, ve kterém může odpovídat na dotazy o stávajících chybových podmínkách. Při zjištění kritické chyby S7-200 přejde do režimu STOP, rozsvítí kontrolku SF/DIAG (červená) a STOP, zablokuje tabulku výstupů a vypne výstupy. V tomto stavu S7-200 zůstane, dokud není odstraněna kritická chyba.

Po provedení změn, nutných pro nápravu kritické chyby, použijte pro restart S7-200 některý z následujících způsobů:

- ❏ Vypněte a znovu zapněte napájení.
- ❏ Přepněte přepínač režimu z polohy RUN nebo TERM do polohy STOP.
- ❏ Vyberte příkaz menu **PLC > Softwarový reset** v STEP 7–Micro/WIN, který provede restart S7-200 a vymaže všechny kritické chyby.

Restartem S7-200 se vymaže kritická chyba a provede diagnostický test, který ověří, že byla kritická chyba odstraněna. Pokud by byla nalezena jiná kritická chyba, S7-200 opět rozsvítí chybovou LED diodu, která indikuje, že chyba nadále přetrvává. S7-200 jinak zahájí normální provoz.

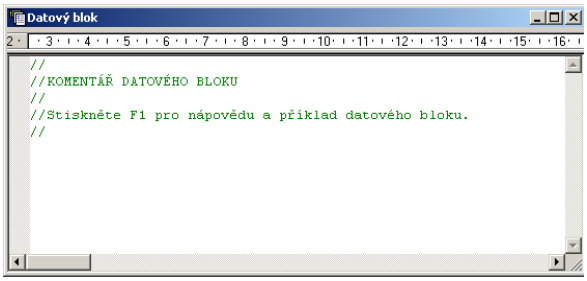
Některé chyby mohou způsobit nefunkčnost komunikace s automatem S7-200. V těchto případech si S7-200 nemůžete prohlédnout chybový kód. Takové typy chyby indikují poruchy hardwaru, které vyžadují opravu S7-200; není možné je opravit změnou programu nebo vymazáním paměti S7-200.

Přiřazení adres a počátečních hodnot v editoru datového bloku


Datový blok

Editor datového bloku vám umožňuje provést přiřazení počátečních hodnot pouze pro paměť V. Přiřazení můžete provést k hodnotám byte, word nebo double word paměti V. Komentáře jsou nepovinné.

Editor datového bloku je textový editor volného formátu; to znamená, že pro konkrétní informace nejsou definována žádná specifická pole. Když dokončíte psaní řádku a stisknete klávesu Enter, editor datového bloku řádek zformátuje (zarovná sloupce adres, dat a komentářů, napíše adresy paměti V velkými písmeny) a znovu ho zobrazí. Stisknutím CTRL–ENTER po dokončení řádku přiřazení se adresa automaticky přesune na nejbližší dostupnou adresu.



Obr. 5-7 Editor datového bloku

Editor datového bloku přiřadí příslušně velkou část paměti V na základě vašich předcházejících přiřazení adresy a velikosti (byte, word nebo double word) hodnoty/hodnot dat.

První řádek datového bloku musí mít přiřazenu absolutní adresu. Následující řádky mohou mít přiřazeny absolutní nebo implicitní adresu. Přiřazení implicitní adresy editor provede, když po přiřazení jediné adresy napíšete více datových hodnot nebo napíšete řádek, který obsahuje pouze datové hodnoty.

Editor datového bloku akceptuje velká nebo malá písmena. Pro oddělení adres a datových hodnot jsou povoleny čárky, tabulátory nebo mezery.

Použití tabulky symbolů pro symbolické adresování proměnných

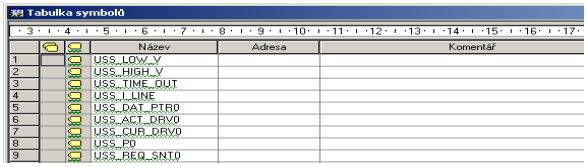

Tabulka symbolů

Tabulka symbolů umožňuje definovat a editovat symboly, ke kterým je odkudkoliv z programu možné přistupovat pomocí symbolických názvů. Tabulek symbolů můžete vytvořit několik. V tabulce symbolů je také záložka pro symboly definované systémem, které je možné použít v uživatelském programu. Tabulka symbolů se také označuje jako tabulka globálních proměnných.

Operandy instrukcí uživatelského programu můžete označit absolutně nebo symbolicky. Absolutní odkaz používá pro označení adresy paměťový bit nebo byte. Symbolický odkaz používá pro označení adresy kombinaci alfanumerických znaků.

V programech SIMATIC se provádí přiřazení globálních symbolů pomocí tabulky symbolů. V programech IEC se provádí přiřazení globálních symbolů pomocí tabulky globálních proměnných.

Chcete–li přiřadit symbol adrese:



Obr. 5-8 Tabulka symbolů

1. Klikněte na ikonu ”Tabulka symbolů” v navigační liště; tím otevřete tabulku symbolů.
2. Vložte jméno symbolu (například Input1) do sloupce ”Název”. Maximální délka symbolu je 23 znaků.
3. Vložte adresu (například I0.0) do sloupce ”Adresa”.
4. V tabulce globálních proměnných IEC vložte hodnotu do sloupce ”Typ dat” nebo ji vyberte ze seznamu.

Můžete vytvořit více tabulek symbolů; nemůžete však použít stejný řetězec pro přiřazení globálního symbolu více než jednou, a to ani v rámci jedné nebo více tabulek.

Použití lokálních proměnných

Tabulku lokálních proměnných vytvořenou v programovém editoru můžete použít pro přiřazení proměnných, které jsou jedinečné pro konkrétní podprogram nebo podprogram přerušení. Viz obrázek 5-9.

Lokální proměnné se mohou použít jako parametry, které jsou předávány podprogramu; zvyšují tak jeho přenositelnost nebo opakované použití.



	Název	Typ proměnné	Datový typ	Komentář
	EN	IN	BOOL	
L0.0	Prv_pruchod	IN	BOOL	příznak prvního průchodu
LB1	Addr	IN	BYTE	adresa zařízení
LW2	Data	IN	INT	data k zapsání do slave
LB4	Status	IN_OUT	BYTE	stav zápisu
L5.0	Hotovo	OUT	BOOL	příznak dokončení
		OUT		
		TEMP		

Obr. 5-9 Tabulka lokálních proměnných

Použití stavového diagramu pro monitorování uživatelského programu



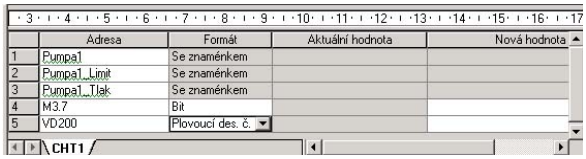
Stavová tabulka

Stavový diagram umožňuje monitorovat nebo modifikovat hodnoty procesních proměnných při běhu řídícího programu S7-200. Můžete sledovat stav programových vstupů, výstupů nebo proměnných tím, že si zobrazíte jejich aktuální hodnoty. Stavový diagram také umožňuje vnutit nebo změnit hodnoty procesních proměnných.

Můžete vytvořit více stavových diagramů, abyste mohli prohlížet různé části programu.

Chcete–li otevřít stavový diagram, vyberte příkaz menu **Zobrazit > Komponenty > Stavový diagram** nebo klikněte na ikonu ”Stavový diagram” na navigační liště.

Když vytváříte stavový diagram, zadáváte adresy procesních proměnných pro monitorování. Není možné zobrazit stav konstant, akumulátorů nebo lokálních proměnných. Hodnotu časovače nebo čítače můžete zobrazit buď jako bit, nebo jako word. Při zobrazení hodnoty jako bitu uvidíte stav časovače nebo čítače; zobrazení hodnoty jako word vám ukáže hodnotu časovače nebo čítače.



	Adresa	Formát	Aktuální hodnota	Nová hodnota
1	Pumpa1	Se znaménkem		
2	Pumpa1_Limit	Se znaménkem		
3	Pumpa1_Tlak	Se znaménkem		
4	M3.7	Bit		
5	VD200	Plovoucí des. č.		

CHT1

Obr. 5-10 Stavová tabulka

Chcete–li vytvořit stavový diagram a monitorovat proměnné:

1. Vložte adresy všech požadovaných hodnot do políčka ”Adresa”.
2. Vyberte typ dat ve sloupci ”Formát”.
3. Chcete–li zobrazit stav procesních proměnných v automatu S7-200, vyberte příkaz menu **Odladit > Stavový diagram**.
4. Chcete–li nepřetržitě vzorkovat hodnoty nebo jednorázově odečíst stav, klikněte na tlačítko na nástrojové liště. Stavový diagram také umožňuje modifikovat nebo vnutit hodnoty různých procesních proměnných.

Další řádky do stavového diagramu můžete vložit pomocí příkazu menu **Úpravy > Vložit > Řádek**.



Tip
Můžete vytvořit více stavových diagramů tak, aby byly proměnné rozděleny do logických skupin, z nichž lze každou prohlížet v kratší, samostatné formě.

Vytvoření knihovny instrukcí

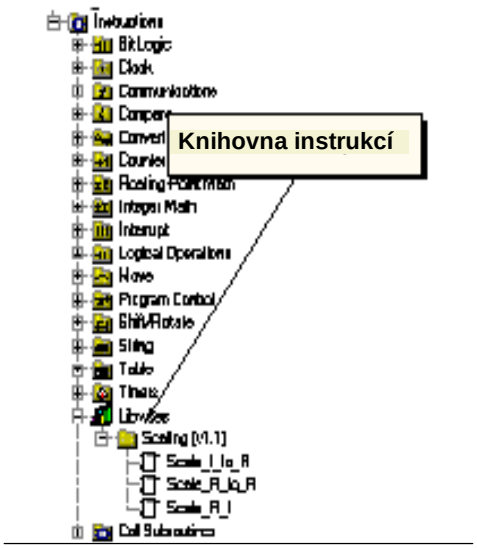
STEP 7–Micro/WIN umožňuje vytvoření své vlastní knihovny instrukcí nebo použití knihovny, vytvořené někým jiným. Viz obrázek 5-11.

Chcete–li vytvořit knihovnu instrukcí, vytvořte standardní podprogram STEP 7–Micro/WIN a podprogram přerušení a seskupte je. Abyste zabránili náhodným změnám nebo z důvodu ochrany know-how autora, můžete v těchto podprogramech skrýt kód.

Chcete–li vytvořit knihovnu instrukcí, proveďte následující kroky:

1. Napište program jako standardní projekt STEP 7–Micro/WIN a funkci, která má být zahrnuta do knihovny, vložte do podprogramů nebo přerušení.
2. Zkontrolujte, že všechna místa paměti V v podprogramech nebo přerušeních mají přiřazen symbolický název. Abyste minimalizovali velikost paměti V, kterou knihovna potřebuje, použijte za sebou jdoucí místa paměti V.
3. Přejmenujte podprogramy a přerušení tak, aby měly názvy, které chcete mít v knihovně instrukcí.
4. Pro kompilování nové knihovny instrukcí použijte příkaz menu **Soubor > Vytvořit knihovnu**.

Více informací o tvorbě knihoven naleznete v online nápovědě programu STEP 7–Micro/WIN.



Obr. 5-11 Instrukční strom s knihovnami

Pro přístup k instrukci v knihovně instrukcí použijte následující postup:

1. Pomocí příkazu menu **Soubor > Přidat knihovnu** přidejte adresář ”Knihovny” do instrukčního stromu.
2. Vyberte příslušnou instrukci a vložte ji do programu (stejně jako jakoukoliv standardní instrukci).

   Jestliže podprogram pro knihovnu potřebuje část paměti V, STEP 7–Micro/WIN vás po kompilaci projektu vyzve k přiřazení paměťového bloku. Pro přiřazení paměťových bloků použijte dialogové okno ”Přidělení paměti pro knihovnu”.

Funkce pro adaptaci uživatelského programu

STEP 7–Micro/WIN má následující funkce, které vám pomohou při ladění vašeho programu:

- ❏ Záložky, které u dlouhého programu usnadňují přechody dopředu a zpět mezi řádky.
- ❏ Tabulku křížových odkazů, která umožňuje kontrolu odkazů použitých v programu.
- ❏ Editování v režimu RUN umožňuje provádět malé změny programu s minimálním narušením řízeného procesu. Při editaci v režimu RUN také můžete provést download programového bloku.

Více informací o ladění vašeho programu najdete v kapitole 8.

Instrukční soubor S7-200

6

Tato kapitola popisuje soubor instrukcí SIMATIC a IEC 1131 pro mikro–PLC S7-200.

Přehled kapitoly

Konvence použité při popisu instrukcí 67
Rozsahy a vlastnosti paměti S7-200 68
Instrukce bitové logiky 70
Kontakty 70
Cívky 73
Práce se zásobníkem 75
Bistabilní klopný obvod 77
Instrukce hodin 78
Instrukce pro komunikaci 81
Čtení a zápis do sítě 81
Vysílání a příjem (Freeport) 86
Zjištění a nastavení adresy portu 95
Instrukce porovnávání 96
Porovnávání číselných hodnot 96
Porovnávání řetězce 98
Instrukce pro konverzi 99
Standardní konverze 99
Konverze na ASCII 103
Konverze řetězců 107
Zakódování a dekódování 112
Instrukce čítače 113
Čítač SIMATIC 113
Čítač IEC 116
Instrukce vysokorychlostního čítače 118
Instrukce pulzního výstupu 133
Matematické instrukce 140
Sčítání, odčítání, násobení a dělení 140
Násobení celých čísel s 32bitovým výsledkem a dělení celých čísel se zbytkem 142
Numerické funkce 143
Inkrementace a dekrementace 144
Instrukce PID algoritmu 145
Instrukce přerušení 153
Instrukce logické operace 161
Inverze 161
AND, OR a X OR 162
Instrukce přesunu 164
Přesun byte, word, double word nebo real 164
Okamžitý přesun bytu (čtení a zápis) 165
Přesun bloku 166

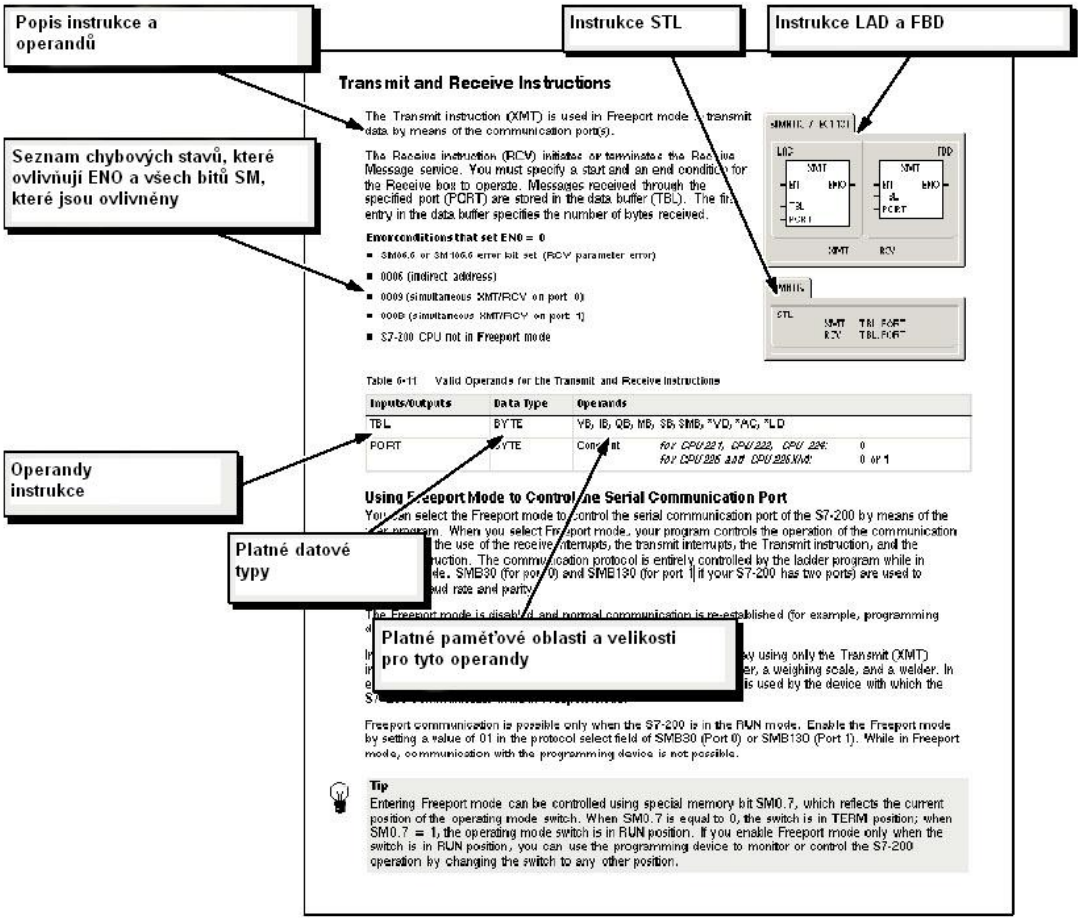
Řízení programu 167
Podmíněný konec programu 167
Stop 167
Reset Watchdogu 167
Smyčka For – Next 169
Skokové instrukce 171
Sekvenční řídící relé (SCR) 172
Instrukce diagnostických kontrolek 178
Instrukce posunu a rotace 179
Posunu vpravo a vlevo 179
Rotace vpravo a vlevo 179
Posunutí bitů registru 181
Výměna bytů 183
Instrukce pro řetězce 184
Instrukce pro práci s tabulkou 189
Přidávání do tabulky 189
Instrukce FIFO a LIFO 190
Vyplnění paměti 192
Prohledávání tabulky 193
Instrukce časovače 196
Časovač SIMATIC 196
Časovač IEC 201
Intervalové hodiny 203
Instrukce podprogramů 204

Konvence použité při popisu instrukcí

Obrázek 6-1 znázorňuje typický popis instrukce a ukazuje na různé oblasti používané pro popis instrukce a její vykonání. Instrukce jsou ve formátu LAD, FBD a STL. Tabulka operandů uvádí seznam operandů instrukce a ukazuje platné datové typy paměťové oblasti a velikost každého operandu.

Operandy a datové typy EN/ENO nejsou uvedeny v tabulce operandu, protože tyto operandy jsou pro všechny instrukce LAD a FBD stejné.

- *Pro LAD:* EN (Enable IN) a ENO (Enable Output) představují signálový tok; jde o booleovský typ dat.
- *Pro FBD:* EN a ENO představují I, Q, V, M, SM, S, T, C, L nebo signálový tok; jde o booleovský typ dat.



Obr. 6-1 Popisy instrukcí

Rozsahy a vlastnosti paměti S7-200

Tabulka 6-1 Rozsahy pamětí

Popis	CPU 221	CPU 222	CPU 224	CPU 224XP	CPU 226
Velikost uživatelského programu					
s editací v režimu RUN	4096 bytů	4096 bytů	8192 bytů	12288 bytů	16384 bytů
bez editace v režimu RUN	4096 bytů	4096 bytů	12288 bytů	16384 bytů	24576 bytů
Velikost uživatelských dat	2048 bytů	2048 bytů	8192 bytů	10240 bytů	10240 bytů
Registr obrazu vstupů	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7
Registr obrazu výstupů	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7
Analogové vstupy (čtení)	AIW0 až AIW30	AIW0 až AIW30	AIW0 až AIW62	AIW0 až AIW62	AIW0 až AIW62
Analogové výstupy (zápis)	AQW0 až AQW30	AQW0 až AQW30	AQW0 až AQW62	AQW0 až AQW62	AQW0 až AQW62
Paměť pro proměnné (V)	VB0 až VB2047	VB0 až VB2047	VB0 až VB8191	VB0 až VB10239	VB0 až VB10239
Lokální paměť (L)[1]	LB0 až LB63	LB0 až LB63	LB0 až LB63	LB0 až LB63	LB0 až LB63
Bitová paměť (M)	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7
Speciální paměť (SM)	SM0.0 až SM179.7	SM0.0 až SM299.7	SM0.0 až SM549.7	SM0.0 až SM549.7	SM0.0 až SM549.7
Pouze pro čtení	SM0.0 až SM29.7	SM0.0 až SM29.7	SM0.0 až SM29.7	SM0.0 až SM29.7	SM0.0 až SM29.7
Časovače	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)
Remanentní zpožděné zapnutí 1 ms	T0, T64	T0, T64	T0, T64	T0, T64	T0, T64
10 ms	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68
100 ms	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95
Zpožděné zapnutí/vypnutí 1 ms	T32, T96	T32, T96	T32, T96	T32, T96	T32, T96
10 ms	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100
100 ms	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255
Čítače	C0 až C255	C0 až C255	C0 až C255	C0 až C255	C0 až C255
Vysokorychlostní čítače	HC0 až HC5	HC0 až HC5	HC0 až HC5	HC0 až HC5	HC0 až HC5
Sekvenční řídicí relé (S)	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7
Registry akumulátoru	AC0 až AC3	AC0 až AC3	AC0 až AC3	AC0 až AC3	AC0 až AC3
Skoky/návěští	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
Volání/podprogramy	0 až 63	0 až 63	0 až 63	0 až 63	0 až 127
Přerušovací podprogramy	0 až 127	0 až 127	0 až 127	0 až 127	0 až 127
Náběžné/sestupné hrany	256	256	256	256	256
Smyčky PID	0 až 7	0 až 7	0 až 7	0 až 7	0 až 7
Porty	Port 0	Port 0	Port 0	Port 0, Port 1	Port 0, Port 1

[1] LB60 až LB63 jsou rezervovány pro STEP 7–Micro/WIN, verze 3.0 nebo pozdější.

Tabulka 6-2 Rozsahy operandů CPU S7-200

Způsob přístupu		CPU 221	CPU 222	CPU 224	CPU 224 XP	CPU 226
Přístup po bitech (byte.bit)						
	I	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7
	Q	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7	0,0 až 15,7
	V	0,0 až 2047,7	0,0 až 2047,7	0,0 až 8191,7	0,0 až 10239,7	0,0 až 10239,7
	M	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7
	SM	0,0 až 165,7	0,0 až 299,7	0,0 až 549,7	0,0 až 549,7	0,0 až 549,7
	S	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7	0,0 až 31,7
	T	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
	C	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
	L	0,0 až 63,7	0,0 až 63,7	0,0 až 63,7	0,0 až 63,7	0,0 až 63,7
Přístup po bytech	IB	0 až 15	0 až 15	0 až 15	0 až 15	0 až 15
	QB	0 až 15	0 až 15	0 až 15	0 až 15	0 až 15
	VB	0 až 2047	0 až 2047	0 až 8191	0 až 10239	0 až 10239
	MB	0 až 31	0 až 31	0 až 31	0 až 31	0 až 31
	SMB	0 až 165	0 až 299	0 až 549	0 až 549	0 až 549
	SB	0 až 31	0 až 31	0 až 31	0 až 31	0 až 31
	LB	0 až 63	0 až 63	0 až 63	0 až 63	0 až 63
	AC	0 až 3	0 až 3	0 až 3	0 až 255	0 až 255
	KB (konstanta)	KB (konstanta)	KB (konstanta)	KB (konstanta)	KB (konstanta)	KB (konstanta)
Přístup po word	IW	0 až 14	0 až 14	0 až 14	0 až 14	0 až 14
	QW	0 až 14	0 až 14	0 až 14	0 až 14	0 až 14
	VW	0 až 2046	0 až 2046	0 až 8190	0 až 10238	0 až 10238
	MW	0 až 30	0 až 30	0 až 30	0 až 30	0 až 30
	SMW	0 až 164	0 až 298	0 až 548	0 až 548	0 až 548
	SW	0 až 30	0 až 30	0 až 30	0 až 30	0 až 30
	T	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
	C	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
	LW	0 až 62	0 až 62	0 až 62	0 až 62	0 až 62
	AC	0 až 3	0 až 3	0 až 3	0 až 3	0 až 3
	AIW	0 až 30	0 až 30	0 až 62	0 až 62	0 až 62
	AQW	0 až 30	0 až 30	0 až 62	0 až 62	0 až 62
	KW (konstanta)	KW (konstanta)	KW (konstanta)	KW (konstanta)	KW (konstanta)	KW (konstanta)
Přístup po double word	ID	0 až 12	0 až 12	0 až 12	0 až 12	0 až 12
	QD	0 až 12	0 až 12	0 až 12	0 až 12	0 až 12
	VD	0 až 2044	0 až 2044	0 až 8188	0 až 10236	0 až 10236
	MD	0 až 28	0 až 28	0 až 28	0 až 28	0 až 28
	SMD	0 až 162	0 až 296	0 až 546	0 až 546	0 až 546
	SD	0 až 28	0 až 28	0 až 28	0 až 28	0 až 28
	LD	0 až 60	0 až 60	0 až 60	0 až 60	0 až 60
	AC	0 až 3	0 až 3	0 až 3	0 až 3	0 až 3
	HC	0 až 5	0 až 5	0 až 5	0 až 5	0 až 5
	KD (konstanta)	KD (konstanta)	KD (konstanta)	KD (konstanta)	KD (konstanta)	KD (konstanta)

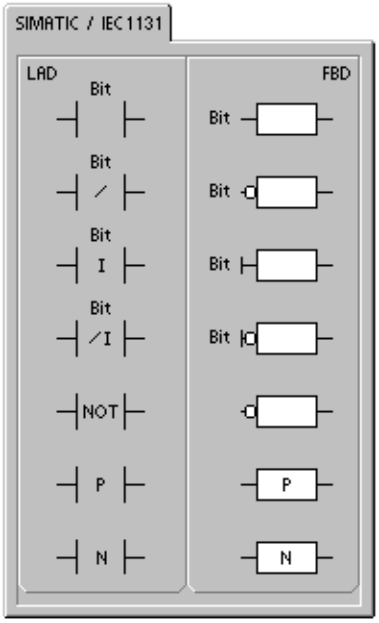
Instrukce bitové logiky

Kontakty

Standardní kontakty (Standard Contacts)

Instrukce kontaktu ”V klidu otevřen” (LD, A a O) a ”V klidu zavřen” (LDN, AN, ON) získávají hodnotu z paměti nebo z registru obrazu. Instrukce standardního kontaktu získávají hodnotu z paměti (nebo z registru obrazu procesu, pokud je typ dat I nebo Q).

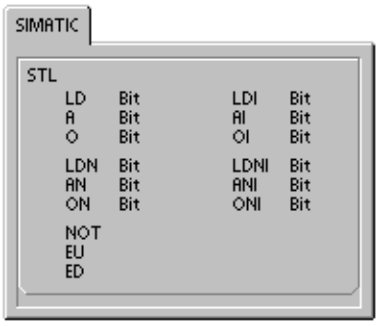
Kontakt ”V klidu otevřen” je sepnutý, když se bit rovná 1, kontakt ”V klidu zavřen” je sepnutý, když se bit rovná 0. V editoru FBD je možné vstupy do bloků AND a OR rozšířit na maximálně 32 vstupů. V editoru STL instrukce kontaktu ”V klidu otevřen”, ”Načíst”, ”AND” nebo ”OR” vykonávají funkce Načíst, AND nebo OR daného bitu s horním bitem zásobníku, instrukce kontaktu ”V klidu zavřen”, ”Načíst negované”, ”NAND” nebo ”NOR” vykonávají funkce negace Načíst, AND nebo OR daného bitu s horním bitem zásobníku.



Okamžité kontakty

Přímý kontakt se aktualizuje okamžitě, nezávisí na programovém cyklu. Instrukce kontaktu ”V klidu otevřen, okamžitý” (LDI, AI a OI) a instrukce ”V klidu zavřen, okamžitý”(LDNI, ANI a ONI) získají hodnotu z fyzického vstupu v době vykonávání instrukce; registr obrazu se ale neaktualizuje.

Kontakt ”V klidu otevřen, okamžitý” je sepnutý, když je na vstupu logická 1, kontakt ”V klidu zavřen, okamžitý” je sepnutý, když je na vstupu logická 0. Instrukce ”V klidu otevřen” okamžitě provedou operaci Načíst, AND nebo OR s hodnotou na vstupu a horní hodnotou zásobníku; instrukce ”V klidu zavřen” okamžitě provedou operaci Načíst negované, NAND nebo NOR negace hodnoty vstupu s horní hodnotou zásobníku.



Negace (NOT)

Instrukce ”Negace” (NOT) mění stav vstupu signálového toku (tzn. že mění horní hodnotu zásobníku z 0 na 1 nebo z 1 na 0).

Náběžná a sestupná hrana

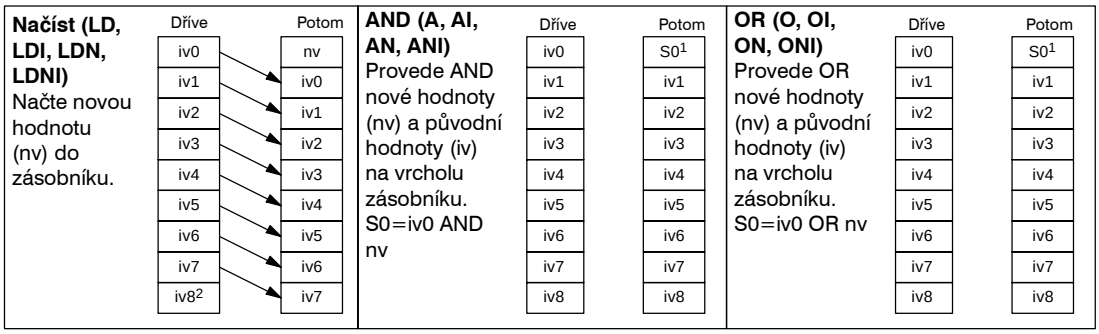
Instrukce ”Náběžná hrana” (EU) aktivuje signálový tok pouze pro jeden programový cyklus, pokud signál na vstupu změní stav z log. 0 na log.1. Instrukce ”Sestupná hrana” (ED) aktivuje signálový tok pouze pro jeden programový cyklus, pokud signál na vstupu změní stav z log. 1 na log. 0. U instrukce ”Náběžná hrana” nastaví detekce přechodu z 0 na 1 horní hodnotu zásobníku na 1; jinak je nastavena na 0. U instrukce ”Sestupná hrana”detekce přechodu z 1 na 0 nastaví horní hodnotu zásobníku na 1; jinak je nastavena na 0.

Při editaci programu v režimu RUN musíte vložit parametr pro instrukce náběžné a sestupné hrany. Více informací o editaci v režimu RUN najdete v kapitole 5.

Tabulka 6-3 Operandy platné pro vstupy instrukcí bitové logiky

Vstupy/Výstupy	Typ dat	Operandy
Bit	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Bit (okamžitý)	BOOL	I

Automat S7-200 používá při vykonávání řídicího programu zásobník – viz obrázek 6-2. V těchto příkladech označují ”iv0” až ”iv7” počáteční hodnoty zásobníku, ”nv” označuje novou hodnotu danou instrukcí a ”S0” označuje vypočítanou hodnotu, která je uložena do zásobníku.



1 S0 označuje vypočítanou hodnotu, která je uložena v logickém zásobníku.
2 Po provedení instrukce Načíst je hodnota iv8 ztracena.

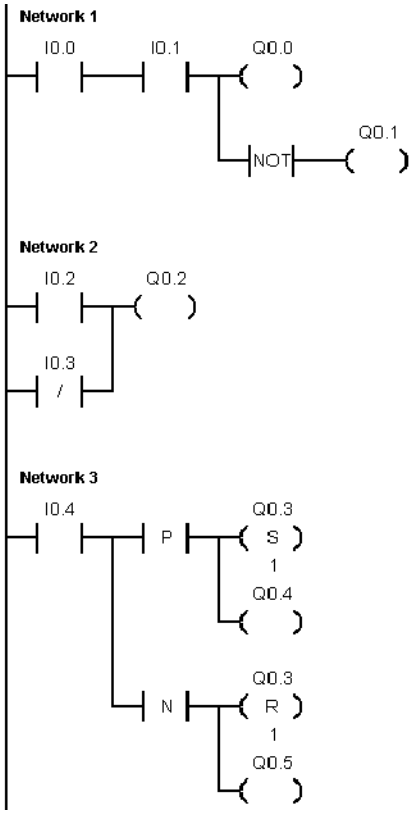
Obr. 6-2 Operace instrukcí kontaktů.



Tip

Protože instrukce náběžná a sestupná hrana vyžadují přechod ze zapnutého do vypnutého stavu nebo naopak, není možné zjistit náběžnou nebo sestupnou hranu v prvním programovém cyklu. Během prvního programového cyklu nastaví S7-200 stav bitu specifikovaného těmito instrukcemi. V příštích cyklech již mohou instrukce zjišťovat přechody u specifikovaného bitu.

Příklad: Instrukce kontaktů

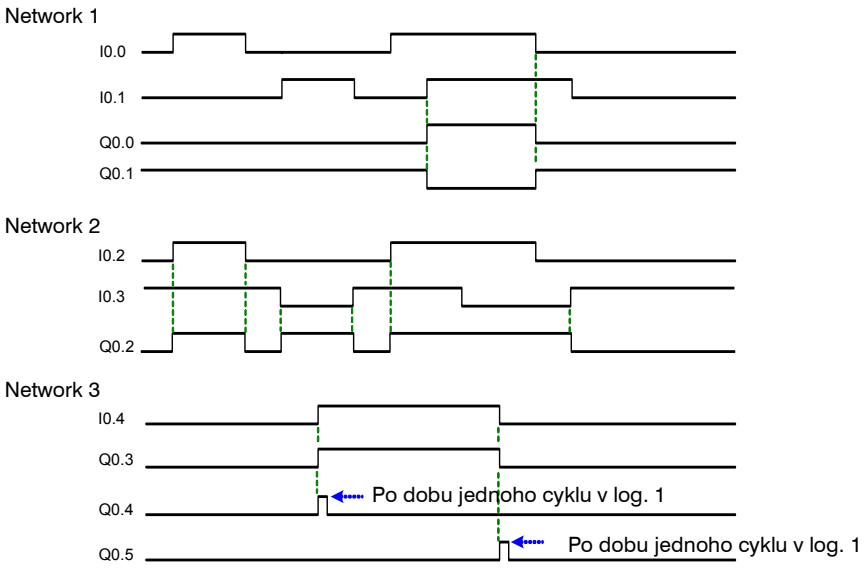



```
Network 1   //Pro aktivaci Q0.0 1 musí být sepnuté kontakty I0.0
            //a I0.1. Instrukce NOT
            //působí jako invertor. V režimu RUN
            // mají Q0.0 a Q0.1 opačné logické stavy.
LD    I0.0
A     I0.1
=     Q0.0
NOT
=     Q0.1

Network 2   //Pro aktivaci Q0.2 musí být kontakt I0.2 sepnutý nebo
            //kontakt I0.3 musí být rozepnutý.
            //Jedna nebo více paralelní větví LAD
            //(logické vstupy OR) musí být pravdivé, aby
            //byl výstup aktivní.
LD    I0.2
ON    I0.3
=     Q0.2

Network 3   //Náběžná hrana na kontaktu P nebo sestupná hrana
            //na kontaktu N
            //má na výstupu pulz v trvání 1 programového
            //cyklu. V režimu RUN jsou změny stavu
            //na Q0.4 a Q0.5 příliš rychlé na to,
            // aby byly viditelné ve stavovém zobrazení.
            //Výstupy nastavení (Set) a vynulování (Reset) zachytí
            // pulz v Q0.3 a zobrazí
            //změnu stavu ve stavovém zobrazení.
LD    I0.4
LPS
EU
S     Q0.3, 1
=     Q0.4
LPP
ED
R     Q0.3, 1
=     Q0.5
```

Časový diagram



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 specifického bitu zkopírována horní hodnota zásobníku.

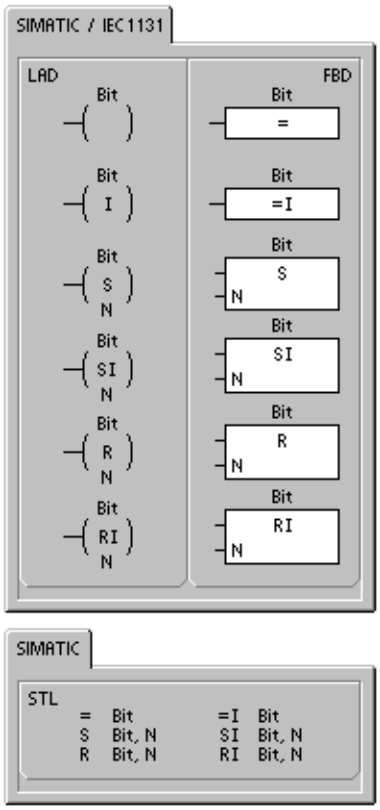
Okamžitý výstup

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). Můžete nastavovat nebo vynulovat 1 až 255 výstupních bodů.



Pokud se instrukce nulování vztahuje na bit časovače (T) nebo na bit čítače (C), vynuluje tato instrukce bit časovače nebo čítače a vymaže jejich aktuální hodnotu.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)

Set okamžité a Reset okamžité

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

”I” označuje okamžitou instrukci; při provádění instrukce je nová hodnota zapsána na fyzický výstup stejně jako na odpovídající místo registru obrazu procesu. Tím se liší od nepřímých instrukcí, které zapisují novou hodnotu pouze do registru obrazu procesu.

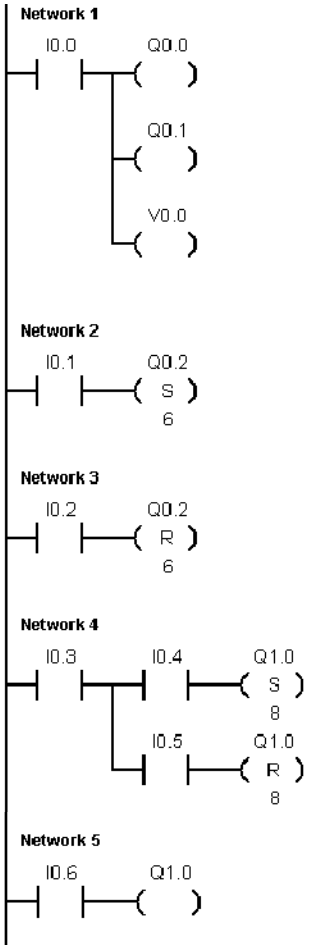
Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)

Tabulka 6-4 Operandy platné pro výstupy instrukcí bitové logiky

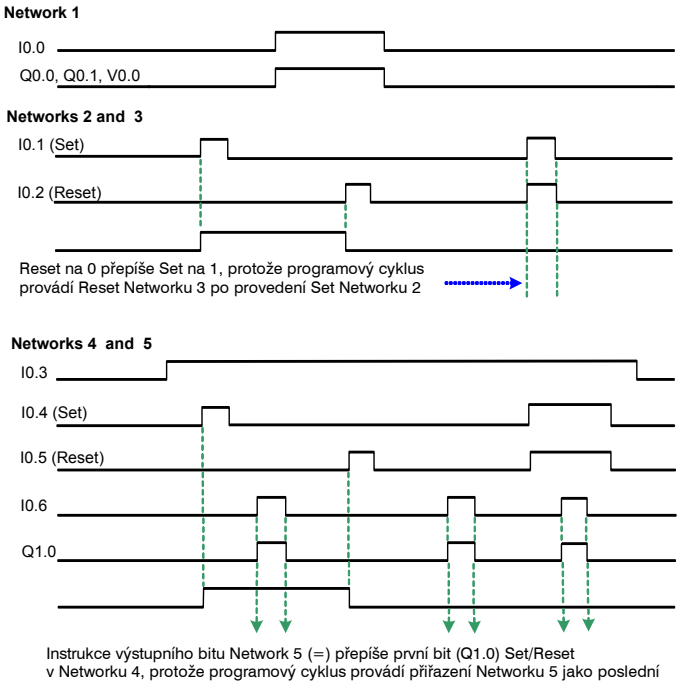
Vstupy/Výstupy	Typ dat	Operandy
Bit	BOOL	I, Q, V, M, SM, S, T, C, L
Bit (okamžitý)	BOOL	Q
N	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta

Příklad: Instrukce cívky




```
Network 1   //Instrukce výstupu přidělí bitové hodnoty vstupům a výstupům
            //(I, Q) a interní paměti (M, SM, T, C, V, S, L).
LD    I0.0
=     Q0.0
=     Q0.1
=     V0.0
Network 2   //Nastaví 6 následujících bitů na hodnotu log. 1. Specifikuje
            //adresu počátečního bitu a počet nastavovaných bitů.
            //Programový indikátor stavu pro Set je zapnut, když je hodnota
            //prvního bitu (Q0.2) 1.
LD    I0.1
S     Q0.2, 6
Network 3   //Nastaví 6 nasledujících bitů na hodnotu log. 0.
            //Specifikuje adresu počátečního bitu a počet nastavovaných
            //bitů. Indikátor stavu programu pro Reset je ON (zapnutý),
            //když hodnota prvního bitu (Q0.2) je log. 0.
LD    I0.2
R     Q0.2, 6
Network 4   //Nastaví a vynuluje 8 výstupních bitů (Q1.0 až Q1.7) jako
            //skupinu.
LD    I0.3
LPS
A     I0.4
S     Q1.0, 8
LPP
A     I0.5
R     Q1.0, 8
Network 5   //Instrukce Set a Reset provádějí funkci západkového
            //relé. Pro oddělení bitů Set/Reset se ujistěte, že nejsou
            //přepsány jinou instrukcí. V tomto příkladu
            //Network 4 nastavuje a obnovuje osm výstupních bitů (Q1.0 až
            //Q1.7) jako skupinu. V režimu RUN může Network 5 přepsat
            //hodnotu bitu Q1.0 a řídit stavové indikátory Set/Reset
            //v Networku 4.
LD    I0.6
=     Q1.0
```

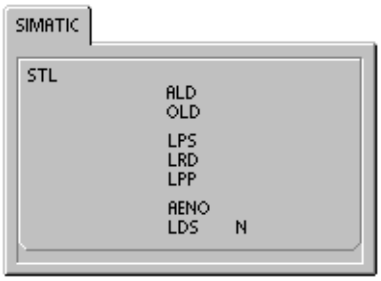
Časový diagram



Práce se zásobníkem

Načíst AND

Instrukce ”Načíst AND” (ALD) kombinuje hodnoty na první a druhé úrovni zásobníku pomocí logické operace AND. Výsledek se uloží na horní místo v zásobníku. Po vykonání instrukce ALD se hloubka zásobníku zmenší o jednu.



OR Load

Instrukce ”Načíst OR” (OLD) kombinuje hodnoty na první a druhé úrovni zásobníku pomocí logické operace OR. Výsledek se uloží na horní místo v zásobníku. Po vykonání instrukce OLD se velikost zásobníku zmenší o jedničku.

Logické Push

Instrukce ”Logické Push” (LPS) duplikuje horní hodnotu zásobníku. Spodní hodnota zásobníku je odsunuta a ztratí se.

Logické čtení

Instrukce ”Logické čtení” (LRD) zkopíruje druhou hodnotu v zásobníku na vrchol zásobníku. Hodnoty v zásobníku se žádným směrem neposouvají, ztratí se však stará hodnota vrcholu zásobníku.

Logické pop

Instrukce ”Logické pop” (LPP) odstraní jednu hodnotu ven ze zásobníku. Druhá hodnota v zásobníku se stane novou hodnotou vrcholu zásobníku.

AND ENO

Instrukce ”AND ENO” (AENO) provede logický AND bitu ENO s vrcholem zásobníku a vytvoří stejný efekt jako bit ENO bloku v LAD nebo FBD. Výsledkem operace AND je nová hodnota na vrcholu zásobníku.

ENO je booleovský výstup bloků v LAD a FBD. Pokud má blok signálový tok na vstupu EN a je proveden bez chyby, výstup ENO předá signálový tok dalšímu elementu. ENO můžete použít jako bit pro povolení, který ukazuje úspěšné dokončení instrukce. Bit ENO se používá s horním bitem zásobníku pro ovladání signálového toku při provádění dalších instrukcí. Instrukce STL nemají vstup EN, aby byly provedeny podmíněné instrukce. Hodnota nejvyššího bitu zásobníku musí být logická 1. Instrukce STL nemají ani výstup ENO. Nicméně instrukce STL, které odpovídají instrukcím LAD a FBD s výstupy ENO, nastavují speciální bit ENO. K tomuto bitu se dá přistupovat pomocí AENO.

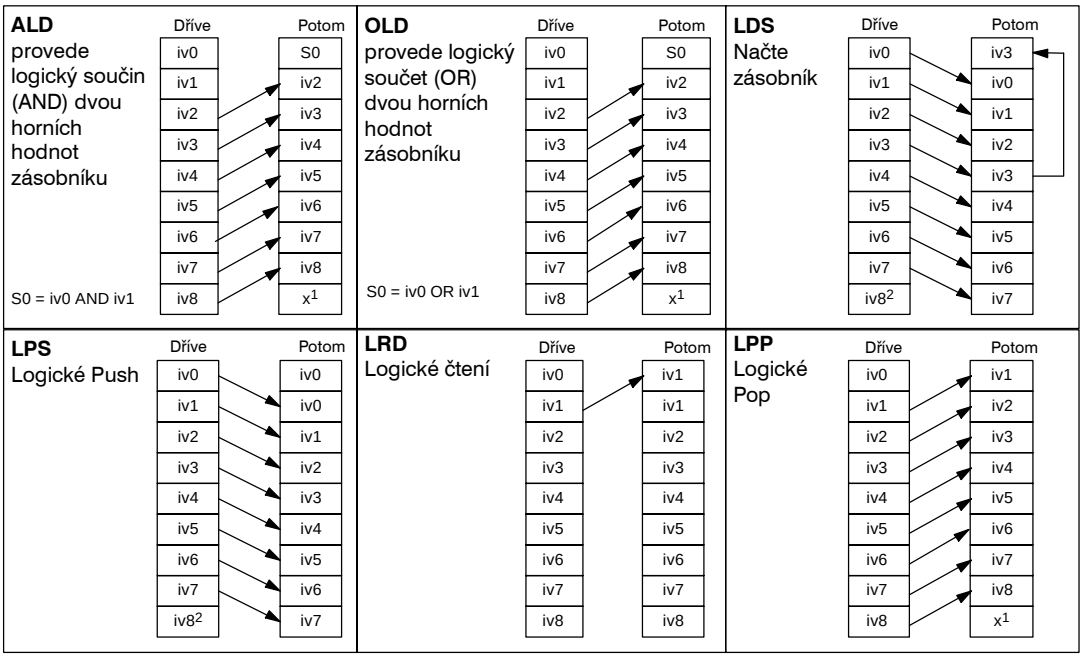
Načíst zásobník

Instrukce ”Načíst zásobník” (LDS) duplikuje bit zásobníku (N) a hodnotu nahraje na vrchol zásobníku. Spodní hodnota zásobníku je odsunuta a ztratí se.

Tabulka 6-5 Operandy platné pro instrukci ”Načíst zásobník”

Vstupy/Výstupy	Typ dat	Operandy
N	BYTE	Konstanta (0 až 8)

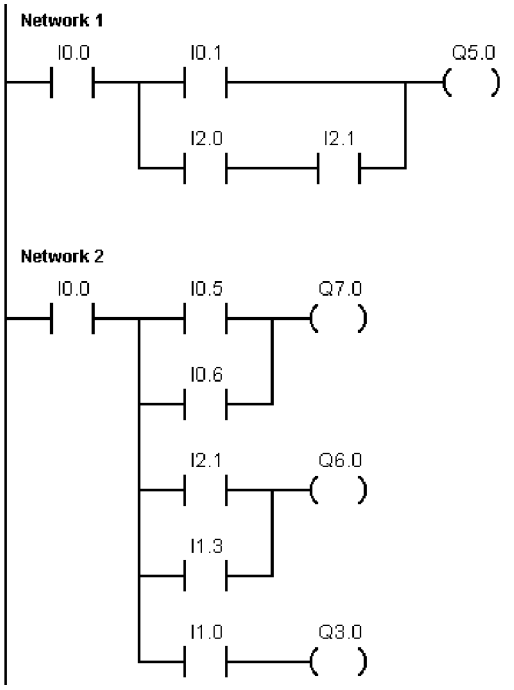
Automat S7-200 používá při vykonávání řídicího programu zásobník – viz obrázek 6-3. V těchto příkladech označují ”iv0” až ”iv7” počáteční hodnoty zásobníku, ”nv” označuje novou hodnotu danou instrukcí a ”S0” označuje vypočítanou hodnotu, která je uložena do zásobníku.



1 Hodnota je neznámá (může to být 0 nebo 1).
2 Po provedení instrukcí ”Logické Push” nebo ”Načíst zásobník” se hodnota iv8 ztratí.

Obr. 6-3 Operace instrukcí logického zásobníku

Příklad: Instrukce pro práci se zásobníkem




```
Network 1
LD    I0.0
LD    I0.1
LD    I2.0
A     I2.1
OLD
ALD
=     Q5.0
Network 2
LD    I0.0
LPS
LD    I0.5
O     I0.6
ALD
=     Q7.0
LRD
LD    I2.1
O     I1.3
ALD
=     Q6.0
LPP
A     I1.0
=     Q3.0
```

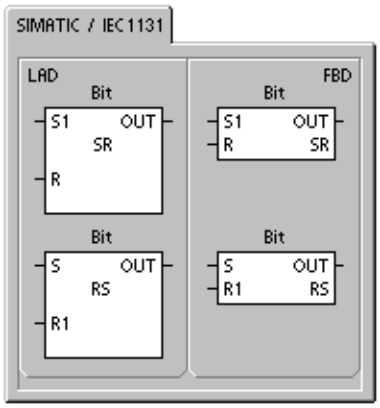
Bistabilní klopný obvod

Instrukce ”SR klopný obvod” je funkce RS klopného obvodu, ve které je dominantní Set. Pokud mají signály Set (S1) a Reset (R) současně hodnotu 1, bude mít hodnotu 1 i výstup (OUT).

Instrukce ”RS klopný obvod” je blokovací funkce RS klopného obvodu, ve které je dominantní Reset. Pokud mají signály Set (S) a Reset (R1) současně hodnotu 1, bude mít výstup (OUT) hodnotu 0.

Booleovský parametr definuje, který bit se nastavuje nebo nuluje. Volitelný výstup odráží stav signálu parametru Bit.

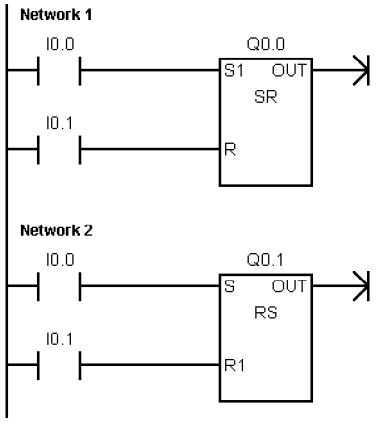
Tabulka 6-7 je pravdivostní tabulka pro vzorový program.



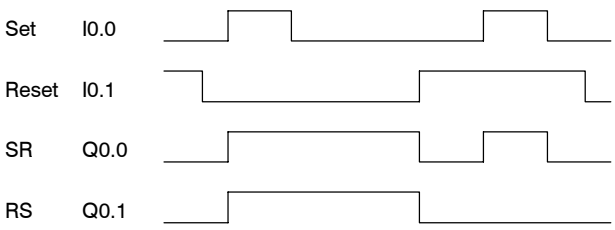
Tabulka 6-6 Operandy platné pro instrukce ”SR klopný obvod” a ”RS klopný obvod”

Vstupy/Výstupy	Typy dat	Operandy
S1, R	BOOL	I, Q, V, M, SM, S, T, C, signálový tok
S, R1, OUT	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Bit	BOOL	I, Q, V, M, S

Příklad: Instrukce Bistabilní KO Set/Reset



Časový diagram

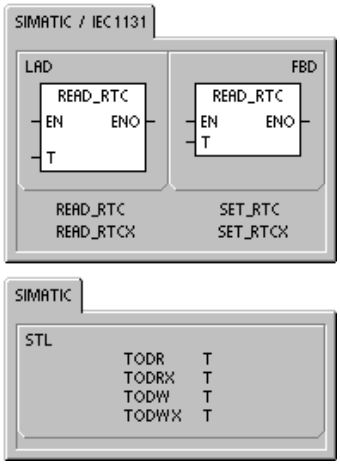


Tabulka 6-7 Pravdivostní tabulka pro instrukce ”SR a RS klopný obvod”

Instrukce	S1	R	Out (Bit)
Instrukce ”Bistabilní KO s převládajícím Set” (SR)	0	0	Předchozí stav
	0	1	0
	1	0	1
	1	1	1
Instrukce	**S**	**R1**	**Out (Bit)**
Instrukce ”Bistabilní KO s převládajícím Reset” (RS)	0	0	Předchozí stav
	0	1	0
	1	0	1
	1	1	0

Instrukce hodin

Čtení a nastavení hodin reálného času



Instrukce "Čtení hodin reálného času" (TODR) přečte aktuální čas a datum z hardwarových hodin a zapíše ho do 8bytového zásobníku se začátkem na adrese T. Instrukce "Nastavení hodin reálného času" (TODW) zapíše do hodin aktuální čas a datum se začátkem na 8bytové adrese zásobníku dané parametrem T.

Všechny hodnoty data a času musí být v BCD formátu (například 16#97 pro rok 1997). Obrázek 6-4 ukazuje formát 8bytového časového zásobníku (T).

Po déletrvajícím výpadku napájení nebo při ztrátě paměti inicializují hodiny reálného času (TOD) následující datum a čas:

Datum: 01–Jan–90
Čas: 00:00:00
Den v týdnu: neděle

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0007 (chyba dat TOD) *pouze "Čtení hodin reálného času"*
- 000C (hodiny nejsou připojeny)

Tabulka 6-8 Operandy platné pro instrukce hodin

Vstupy/Výstupy	Typy dat	Operandy
T	BYTE	IB, QB, VB, MB, SMB, SB, LB, *VD, *LD, *AC

T	T+1	T+2	T+3	T+4	T+5	T+6	T+7
Rok: 00 až 99	Měsíc: 01 až 12	Den: 01 až 31	Hodiny: 00 až 23	Minuty: 00 až 59	Sekundy: 00 až 59	0	Den v týdnu: 0 až 7*

*T+7 1=neděle, 7=sobota
0 znamená den v týdnu vypnutý.

Obr. 6-4 Formát 8bytového časového zásobníku (T)



Tip

CPU S7-200 nekontroluje správné nastavení dne v týdnu podle data. Mohou být přijata i nesprávná data, jako například 30. únor. Proto byste zadání správného data měli kontrolovat.

Nepoužívejte instrukci TODR/TODW současně v hlavním programu i v přerušení. Instrukce TODR/TODW v podprogramu přerušení, které se pokouší o činnost během činnosti jiné instrukce TODR/TODW, nemůže být provedena. SM4.3 se nastaví tak, že označuje pokus o dva současné přístupy na hodiny (nekritická chyba 0007).

Hodiny reálného času v S7-200 používají pro rok pouze dvě poslední číslice, takže např. rok 2000 je reprezentován jako 00. Automat S7-200 informaci o roku žádným způsobem nevyužívá. Ale uživatelské programy, které používají aritmetiku nebo porovnání hodnoty roků, musí počítat s reprezentací roku dvěma číslicemi a s přechodem století.

Přestupný rok je správně zpracovávaný až do roku 2096.

Rozšířené čtení hodin reálného času

Instrukce ”Rozšířené čtení hodin reálného času” (TODRX) čte aktuální čas, datum a nastavení letního času z PLC a načítá je do 19bytového zásobníku začínajícího na adrese specifikované T.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 000C (modul hodin není připojen)
- 0091 (chyba rozsahu)

Nastavení rozšířeného čtení hodin reálného času

Instrukce nastavení hodin reálného času (TODWX) zapíše aktuální čas, datum a nastavení letního času do PLC začínající na adrese 19bytového zásobníku specifikované T.

Všechny hodnoty data a času musí být v BCD formátu (například 16#02 pro rok 2002). Tabulka 6-9 ukazuje formát 19bytového časového zásobníku (T).

Po déletrvajícím výpadku napájení nebo při ztrátě paměti inicializují hodiny denního času následující datum a čas:

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0007 (chyba dat TOD)
- 000C (modul hodin není připojen)
- 0091 (chyba rozsahu)

Datum:	01–Jan–90
Čas	00:00:00
Den v týdnu:	neděle

Tabulka 6-9 Formát 19bytového časového zásobníku (T)

T byte	Popis	Data bytu
0	rok (0 – 99)	běžný rok (BCD value)
1	měsíc (1 – 12)	běžný měsíc (hodnota BCD)
2	den (1 – 31)	běžný den (hodnota BCD)
3	hodina (0 – 23)	běžná hodina (hodnota BCD)
4	minuta (0 – 59)	běžná minuta (hodnota BCD)
5	sekunda (0 – 59)	běžná sekunda (hodnota BCD)
6	00	rezervováno – vždy nastaveno na 00
7	den v týdnu (1 – 7)	běžný den v týdnu, 1=neděle (hodnota BCD)
8	režim (00H – 03H, 08H, 10H – 13H, FFH)	korekční režim: 00H = korekce vypnuta 01H = EU (odchylka časového pásma od UTC = 0 hodin) [1] 02H = EU (odchylka časového pásma od UTC = +1 hodina) [1] 03H = EU (odchylka časového pásma od UTC = +2 hodiny) [1] 04H – 07H = rezervováno 08H = EU (odchylka časového pásma od UTC = –1 hodina) [1] 09H – 0FH = rezervováno10H = US [2] 11H = Austrálie [3] 12H = Austrálie (Tasmánie) [4] 13H = Nový Zéland [5] 14H – FEH = rezervováno FFH = specifikováno uživatelem (pomocí hodnot v bytech 9 – 18)
9	korenční hodiny (0 – 23)	velikost korekce, hodiny (hodnota BCD)
10	korenční minuty (0 – 59)	velikost korekce, minuty (hodnota BCD)
11	počáteční měsíc (1 – 12)	počáteční měsíc letního času (hodnota BCD)
12	počáteční den (1 – 31)	počáteční den letního času (hodnota BCD)
13	počáteční hodina (0 – 23)	počáteční hodina letního času (hodnota BCD)
14	počáteční minuta (0 – 59)	počáteční minuta letního času (hodnota BCD)
15	konečný měsíc (1 – 12)	konečný měsíc letního času (hodnota BCD)
16	konečný den (1 – 31)	konečný den letního času (hodnota BCD)
17	konečná hodina (0 – 23)	konečná hodina letního času (hodnota BCD)
18	konečná minuta (0 – 59)	konečná minuta letního času (hodnota BCD)

[1] Konvence EU: Posunout čas o jednu hodinu dopředu poslední neděli v březnu v 1:00 UTC. Nastavit čas o jednu hodinu dozadu poslední neděli v říjnu ve 2:00 UTC. (Místní čas při provádění korekce závisí na odchylce časového pásma od UTC).

[2] Konvence USA: Posunout čas o jednu hodinu dopředu první neděli v dubnu ve 2:00 místního času. Posunout čas o jednu hodinu dozadu poslední neděli v říjnu ve 2:00 místního času.

[3] Konvence Austrálie: Posunout čas o jednu hodinu dopředu poslední neděli v říjnu ve 2:00 místního času. Posunout čas o jednu hodinu dozadu poslední neděli v březnu ve 3:00 místního času.

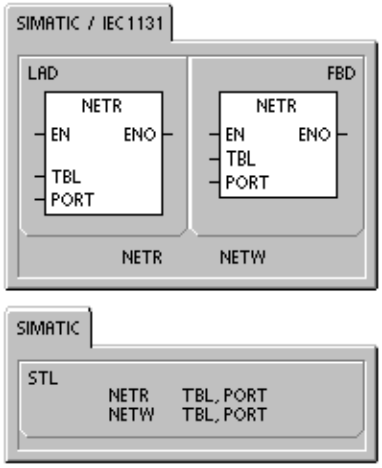
[4] Konvence Austrálie (Tasmánie): Posunout čas o jednu hodinu dopředu první neděli v říjnu ve 2:00 místního času. Posunout čas o jednu hodinu dozadu poslední neděli v březnu ve 3:00 místního času.

[5] Konvence Nového Zélandu: Posunout čas o jednu hodinu dopředu první neděli v říjnu ve 2:00 místního času. Posunout čas o jednu hodinu dozadu v neděli 15. března nebo první neděli po 15. březnu ve 3:00 místního času.

Instrukce pro komunikaci

Čtení a zápis do sítě

Instrukce "Čtení ze sítě" (NETR) inicializuje čtení dat ze vzdáleného zařízení přes daný port (PORT) podle tabulky (TBL). Instrukce "Zápis do sítě" (NETW) inicializuje zápis dat na vzdálené zařízení přes daný port (PORT) podle tabulky (TBL).



Chyby, které nastavují ENO = 0:

- 0006 (nepřímá adresa)
- Pokud má funkce za následek chybu, je nastaven bit E stavového bytu tabulky (viz obrázek 6-5)

Instrukce "Čtení ze sítě" může přečíst až 16 bytů ze vzdálené stanice, instrukce "Zápis do sítě" může zapsat až 16 bytů do vzdálené stanice.

V programu může být libovolný počet instrukcí "Čtení ze sítě" a "Zápis do sítě"; současně však může být aktivováno maximálně osm instrukcí "Čtení ze sítě" a "Zápis do sítě". Ve stejnou dobu mohou být v daném S7-200 aktivovány 4 instrukce "Čtení ze sítě" a 4 instrukce "Zápis do sítě" nebo 2 instrukce "Čtení ze sítě" a 6 instrukcí "Zápis do sítě".

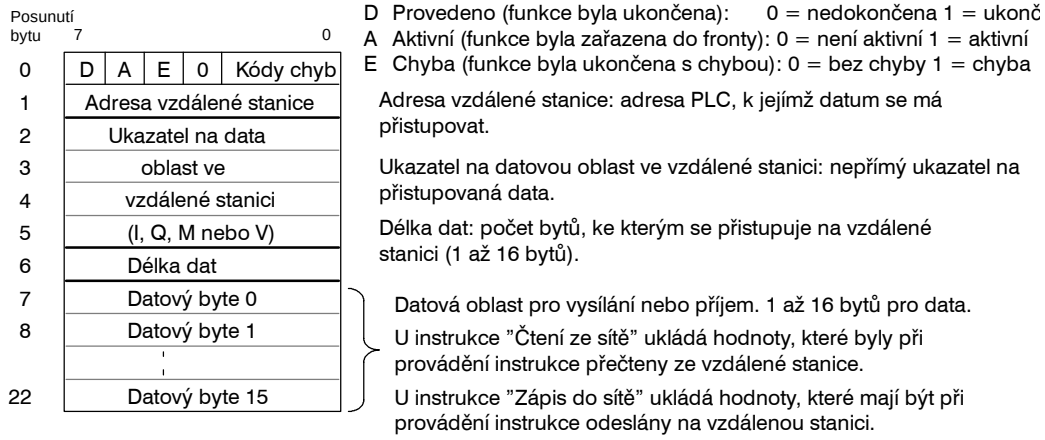

Průvodce

Pro konfiguraci komunikačních instrukcí můžete použít "Průvodce instrukcemi". Chcete–li spustit průvodce pro "Čtení ze sítě"/"Zápis do sítě", vyberte příkaz menu **Nástroje > Průvodce instrukcemi** a pak vyberte "NETR/NETW" z okna průvodce.

Tabulka 6-10 Operandy platné pro instrukce "Čtení ze sítě" a "Zápis do sítě"

Vstupy/Výstupy	Typ dat	Operandy		
TBL	BYTE	VB, MB, *VD, *LD, *AC		
PORT	BYTE	Konstanta	*pro CPU 221, CPU 222, CPU 224:* *pro CPU 224XP, CPU 226:*	0 0 nebo 1

Obrázek 6-5 ukazuje tabulku pro parametr TBL; v tabulce 6-11 je seznam chybových kodů.

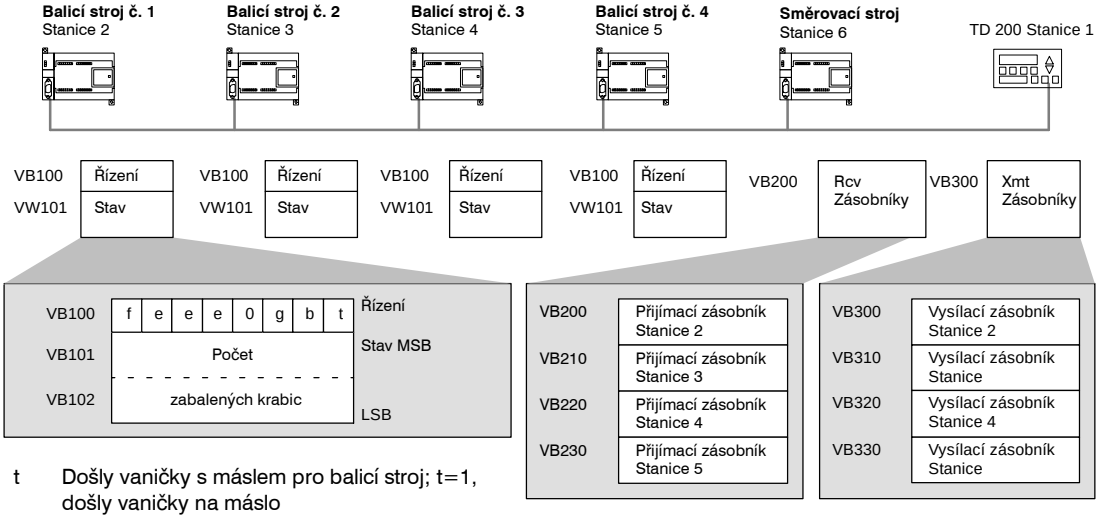


Obr. 6-5 Parametr TBL pro instrukce "Čtení ze sítě" a "Zápis do sítě"

Tabulka 6-11 Chybové kódy pro parametr TBL

Kód	Definice
0	Bez chyby.
1	Překročení časového limitu: Vzdálená stanice neodpovídá.
2	Chyba příjmu: Chyba parity, rámce nebo kontrolního součtu v odpovědi.
3	Chyba offline: Kolize způsobené zdvojenou adresou stanice nebo chybou hardwaru.
4	Chyba přeplnění fronty: Bylo aktivováno více než 8 instrukcí "Čtení ze sítě" nebo "Zápis do sítě".
5	Porušení protokolu: Pokus o provedení instrukce "Čtení ze sítě" nebo "Zápis do sítě" bez povolení PPI Master Mode v SMB30 nebo SMB130.
6	Nepřípustný parametr: Parametr TBL obsahuje nepřípustnou nebo neplatnou hodnotu.
7	Chybí zdroj: Vzdálená stanice je obsazená. (Probíhá upload nebo download.)
8	Chyba úrovně: Porušení protokolu aplikace.
9	Chyba zprávy: Chybná adresa dat nebo nesprávná délka dat.
A až F	Nepoužito. (Rezervováno)

Obrázek 6-6 ukazuje příklad použití instrukcí "Čtení ze sítě" a "Zápis do sítě". V tomto příkladu jde o výrobní linku, kde se plní vaničky máslem a posílají se do jednoho ze čtyř balicích strojů (do krabic). Balicí stroj zabalí osm vaniček másla do jedné lepenkové krabice. Směrovací stroj řídí přísun vaniček másla do každého balicího stroje. Balicí stroje řídí čtyři programovatelné automaty S7-200 a jeden S7-200 s textovým displejem TD 200 řídí směrovač.



t Došly vaničky s máslem pro balicí stroj; t=1, došly vaničky na máslo

b Docházejí krabice; b=1, nutné doplnit krabice v příštích 30 minutách

g Dochází lepidlo; g=1, nutné doplnit lepidlo v příštích 30 minutách

eee chybový kód, který určuje kód chyby, k níž došlo

f Indikátor chyby; f=1, balicí stroj zaznamenal chybu

Obr. 6-6 Příklad instrukcí "Čtení ze sítě" a "Zápis do sítě"

Obrázek 6-7 ukazuje přijímací zásobník (VB200) a vysílací zásobník (VB300) pro přístup k datům ve stanici 2. Automat S7-200 používá instrukci "Čtení ze sítě" pro trvalé čtení řídicích a stavových informací z každého balicího stroje. Směrovač zaznamená každé zabalení 100 krabic balicím strojem a pošle zprávu pro vyčištění stavového slova pomocí instrukce "Zápis do sítě".

Přijímací zásobník *pro čtení z balicího stroje č. 1*

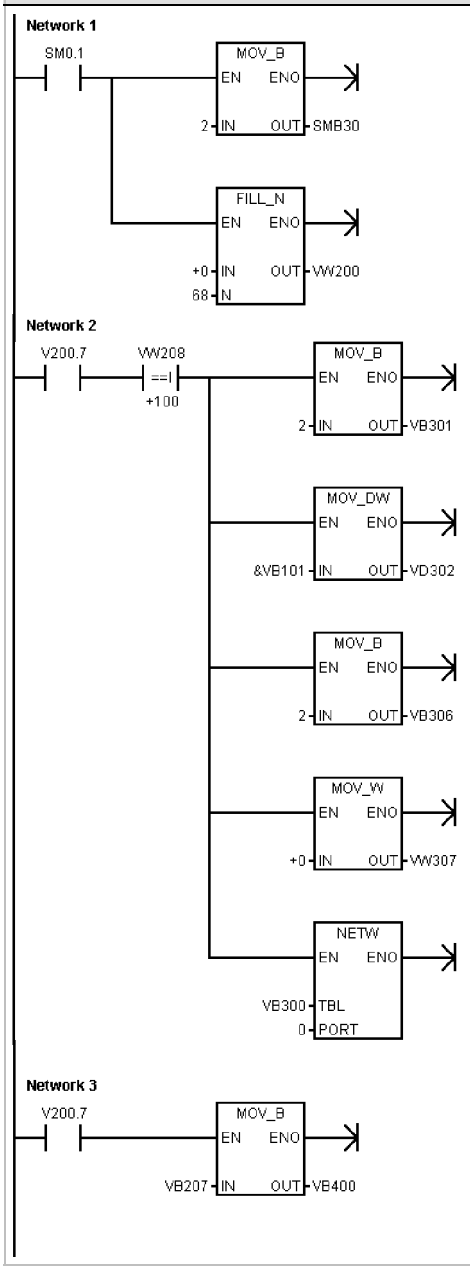
	7				0
VB200	D	A	E	0	Kódy chyb
VB201	Adresa vzdálené stanice = 2				
VB202	Ukazatel na				
VB203	datovovou oblast				
VB204	ve				
VB205	Vzdálená stanice = (&VB100)				
VB206	Délka dat = 3 byty				
VB207	Řízení				
VB208	Stav (MSB)				
VB209	Stav (LSB)				

Vysílací zásobník *pro vymazání počtu v balicím stroji č. 1*

	7				0
VB300	D	A	E	0	Kódy chyb
VB301	Adresa vzdálené stanice = 2				
VB302	Ukazatel na				
VB303	datovou oblast				
VB304	ve				
VB305	Vzdálená stanice = (&VB101)				
VB306	Délka dat = 2 byty				
VB307	0				
VB308	0				

Obr. 6-7 Vzorek dat TBL pro uvedený příklad "Čtení a zápis do sítě"

Příklad: Instrukce čtení a zápisu do sítě

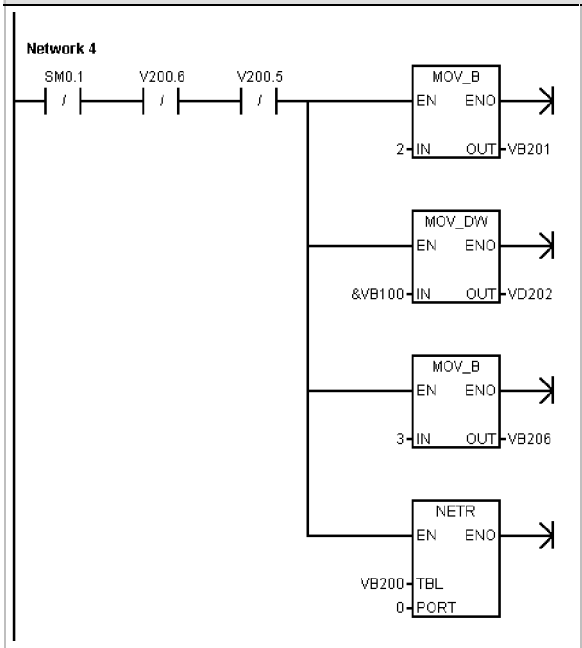



```
Network 1   //V prvním programovém cyklu zapne
            //režim PPI master a vymaže
            //všechny přijímací a vysílací zásobníky.
LD          SM0.1
MOVB        2, SMB30
FILL        +0, VW200, 68

Network 2   //Pokud je bit Provedeno NETR (V200.7)
            //nastaven a 100 krabic bylo zabaleno:
            //1.  Načte se adresa stanice
            //    balicího stroje č. 1
            //2.  Nahraje ukazatel na data ve
            //    vzdálené stanici.
            //3.  Nahraje délku
            //     vysílaných dat.
            //4.  Nahraje vysílaná data.
            //5.  Vynuluje počet krabic zabalených
            //    balicím strojem č. 1
LD          V200.7
AW=         VW208, +100
MOVB        2, VB301
MOVD        &VB101, VD302
MOVB        2, VB306
MOVW        +0, VW307
NETW        VB300, 0

Network 3   //Pokud je nastaven bit Provedeno NETR,
            //uloží řídicí data
            //z balicího stroje č. 1
LD          V200.7
MOVB        VB207, VB400
```

Příklad: Instrukce čtení a zápisu do sítě, pokračování

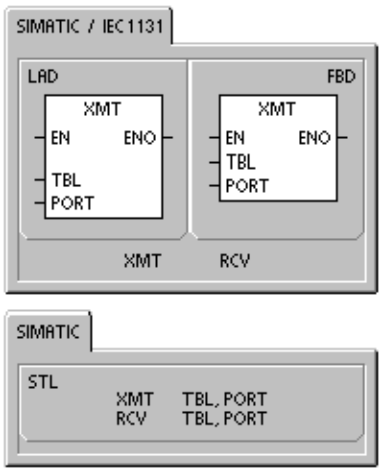



```
Network 4    //Pokud neběží první programový cyklus a
             //nevyskytly se žádné chyby:
             //1.  Načte se adresa stanice
             //    balicího stroje č. 1
             //2.  Nahraje ukazatel na data ve
             //    vzdálené stanici.
             //3.  Nahraje délku
             //    přijímaných dat.
             //4.  Čte řídicí a stavová data
             //    v balicím stroji č. 1
LDN          SM0.1
AN           V200.6
AN           V200.5
MOVB         2, VB201
MOVD         &VB100, VD202
MOVB         3, VB206
NETR         VB200, 0
```

Vysílání a příjem (Freeport)

Instrukce ”Transmit” (XMT) se používá v režimu Freeport pro přenos dat přes komunikační port/porty.

Instrukce ”Příjem” (RCV) vyvolá nebo ukončí funkci přijetí zprávy. Je nutné zadat počáteční a konečnou podmínku pro práci funkce ”Příjem”. Zprávy přijaté přes daný port jsou uloženy v datovém bufferu (TBL). První položka v datovém zásobníku udává počet přijatých bytů.



Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0009 (současné vysílání i příjem na portu 0)
- 000B (současné vysílání i příjem na portu 1)
- Chyba příjmu nastaví SM86.6 nebo SM186.6
- CPU S7-200 není v režimu Freeport

Tabulka 6-12 Operandy platné pro instrukce ”Vysílání” a ”Příjem”

Vstupy/Výstupy	Typ dat	Operandy		
TBL	BYTE	IB, QB, VB, MB, SMB, SB, *VD, *LD, *AC		
PORT	BYTE	Konstanta	*pro CPU 221, CPU 222, CPU 224:* *pro CPU 224XP, CPU 226:*	0 0 nebo 1

Více informací o používání režimu Freeport najdete v odstavci Vytváření protokolů definovaných uživatelem v režimu Freeport na stránce 227 v kapitole 7.

Použití režimu Freeport pro řízení sériových portu

Režim Freeport je možné využít pro řízení komunikace sériového portu automatu S7-200 pomocí uživatelského programu. Jestliže vyberete režim Freeport, řídí uživatelský program komunikační port pomocí přerušení instrukcí ”Vysílání” a ”Příjem”. V režimu Freeport je komunikační protokol zcela řízen uživatelským programem. Pro nastavení přenosové rychlosti a parity se používají SMB30 (pro port 0) a SMB130 (pro port 1, pokud má automat S7-200 dva porty).

Při přechodu S7-200 do režimu STOP se režim Freeport zastaví a obnoví se normální komunikace (např. pro spojení počítačem).

V nejjednodušším případě můžete pomocí instrukce ”Vysílání” (XMT) poslat zprávu tiskárně nebo displeji. Jiné využití je například pro spojení se snímačem čárového kódu, elektronickými vahami a svářečkou. V každém případě je nutné napsat program tak, aby podporoval protokol používaný zařízením, se kterým automat S7-200 komunikuje v režimu Freeport.

Komunikace Freeport je možná pouze tehdy, je–li S7-200 v režimu RUN. Režim Freeport se aktivuje nastavením hodnoty 01 v SMB30 (port 0) nebo SMB130 (port 1). V režimu Freeport není možná komunikace s počítačem.

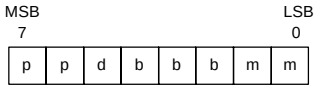


Tip

Režim Freeport je možné řídit pomocí speciálního paměťového bitu SM0.7, který odráží aktuální polohu přepínače pracovního režimu. Když je SM0.7 roven log. 0, je přepínač v poloze TERM; pokud SM0.7 = 1, je přepínač pracovního režimu v poloze RUN. Pokud povolíte režim Freeport pouze v případě, že je přepínač v poloze RUN, můžete použít programovací zařízení na monitorování nebo řízení činnosti S7-200 při přepnutí přepínače do jiné polohy.

Změna komunikace PPI na režim Freeport

SMB30 a SMB130 konfigurují komunikační porty 0 a 1 pro režim Freeport a nastavují přenosovou rychlost, paritu a počet datových bitů. Obrázek 6-8 ukazuje řídicí byte pro Freeport. Pro všechny konfigurace je přidán jeden stop bit.



SMB30 = Port 0
SMB130 = Port 1

pp: Výběr parity
- 00 = bez parity
- 01 = sudá parita
- 10 = bez parity
- 11 = lichá parita

d: Počet datových bitů na znak
- 0 = 8 bitů na znak
- 1 = 7 bitů na znak

bbb: přenosová rychlost pro Freeport
- 000 = 38 400 baud
- 001 = 19 200 baud
- 010 = 9 600 baud
- 011 = 4 800 baud
- 100 = 2 400 baud
- 101 = 1 200 baud
- 110 = 115,2 kbaud[1]
- 111 = 57,6 kbaud[1]

[1] Vyžaduje S7-200 CPU verzi 1.2 nebo vyšší

mm: Výběr protokolu
- 00 = režim PPI/slave
- 01 = protokol Freeport
- 10 = režim PPI/master
- 11 = Rezervováno (implicitně režim PPI/slave)

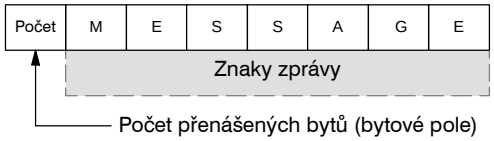
Obr. 6-8 Řídicí byte SM pro režim Freeport (SMB30 nebo SMB130)

Vysílání dat

Instrukce ”Vysílání” umožňuje přenos jednoho nebo více znaků (maximálně 255).

Obrázek 6-9 ukazuje formát přenosového zásobníku.

Pokud je ukončení přenosu řízeno podprogram přerušení, generuje S7-200 přerušení po vyslání posledního znaku ze zásobníku (přerušení 23 pro port 0 a přerušení 26 pro port 1).



Obr. 6-9 Formát vysílacího zásobníku

Přenos je možné uskutečnit bez použití přerušení (např. odeslání zprávy tiskárně) monitorováním SM4.5 nebo SM4.6 pro ukončení přenosu.

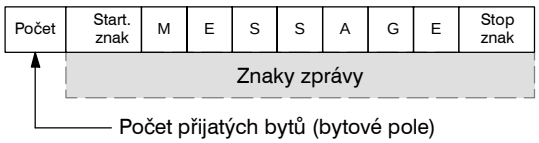
Instrukci ”Vysílání” je možné použít pro generování podmínky BREAK nastavením počtu znaků na nulu a následným provedením instrukce. To generuje podmínku BREAK po dobu 16 bitů při aktuální přenosové rychlosti. BREAK se vysílá stejně jako libovolná jiná zpráva, po ukončení se generuje přerušení vysílání a SM4.5 nebo SM4.6 signalizuje aktuální stav operace vysílání.

Příjem dat

Instrukce ”Příjem” umožňuje příjem zásobníku složeného z jednoho nebo více znaků až do 255.

Obrázek 6-10 ukazuje formát přijímacího zásobníku.

Pokud je ukončení přenosu řízeno podprogramem přerušení, generuje S7-200 přerušení po přijetí posledního znaku do zásobníku. (Přerušení 23 pro port 0 a přerušení 26 pro port 1).



Obr. 6-10 Formát pro přijímací buffer

Zprávy je možné přijímat bez použití přerušení monitorováním SMB86 (port 0) nebo SMB186 (port 1). Pokud je instrukce ”Příjem” neaktivní nebo byla ukončena, je tento byte roven log. 1. Nulový je v případě, že přijímání pokračuje.

Jak je vidět v tabulce 6-13, instrukce ”Příjem” umožňuje zvolit podmínku začátku zprávy a konce zprávy použitím SMB86 až SMB94 pro port 0 a SMB186 až SMB194 pro port 1.



Tip
Funkce příjmu zprávy se automaticky ukončí při přetečení nebo chybě parity. Aby byla funkční, je třeba definovat počáteční podmínku a ukončovací podmínku (maximální počet znaků).

Tabulka 6-13 Byty přijímacího zásobníku (SMB86 až SMB94 a SM1B86 až SMB194)

Port 0	Port 1	Popis
SMB86	SMB186	Příjem zprávy stavový byte MSB 7 … LSB 0 n r e 0 0 t c p n: 1 = Funkce příjmu zprávy ukončena: uživatel vydal příkaz k odpojení. r: 1 = Funkce příjmu zprávy ukončena: chyba ve vstupních parametrech nebo chybí startovací či ukončovací podmínka. e: 1 = Přijat ukončovací znak. t: 1 = Funkce příjmu zprávy ukončena: vypršel čas. c: 1 = Funkce příjmu zprávy ukončena: maximální počet znaků dosažen. p 1 = Funkce příjmu zprávy ukončena: chyba parity.
SMB87	SMB187	Příjem zprávy řídicí byte MSB 7 … LSB 0 en sc ec il c/m tmr bk 0 en: 0 =Funkce příjmu zpráv není povolená. 1 =Funkce príjmu zpráv je povolená. Povolovací bit příjmu zpráv je kontrolován při každém provedení instrukce RCV. sc: 0 = Ignorovat SMB88 nebo SMB188. 1 = Použít hodnotu SMB88 nebo SMB188 pro rozpoznání začátku zprávy. ec: 0 = Ignorovat SMB89 nebo SMB189. 1 = Použít hodnotu SMB89 nebo SMB189 pro rozpoznání konce zprávy. il: 0 = Ignorovat SMW90 nebo SMW190. 1 = Použít hodn. SMW90 nebo SMW190 pro rozpoznání prázdného úseku. c/m: 0 = Časovač je použit na časování mezi znaky. 1 = Časovač je použit na časování zpráv. tmr: 0 = Ignorovat SMW92 nebo SMW192. 1 = Ukončit příjem, pokud čas v SMW92 nebo SMW192 byl překročen. bk: 0 = Ignorovat BREAK (pauzu). 1 = Použít BREAK (pauzu) pro detekci začátku zprávy.
SMB88	SMB188	Znak začátku zprávy.
SMB89	SMB189	Znak konce zprávy.
SMW90	SMW190	Doba prázdného úseku uvedená v milisekundách. První znak přijatý po skončení prázdného úseku je startovací znak.
SMW92	SMW192	Hraniční hodnota pro časovač (v ms) odměřující čas mezi znaky/zprávami. Pokud je hodnota překročena, je funkce příjmu zpráv ukončena.
SMB94	SMB194	Maximální počet přijatých znaků (1 až 255 bytů). Tato hodnota musí být nastavena na předpokládanou maximální velikost zásobníku i v případě, že není použito ukončení příjmu zprávy při dosažení maximálního počtu znaků.

Podmínky začátku a ukončení pro instrukci ”Příjem”

Instrukce ”Příjem” používá pro definování podmínek začátku a ukončení zprávy bity řídicího bytu příjmu zprávy (SMB87 nebo SMB187).

Tip

Pokud na portu probíhá komunikace s jinými zařízeními, když se provádí instrukce ”Příjem”, může funkce příjmu zprávy začít přijímat znak odprostřed, což má za následek eventuální chybu parity a ukončení funkce příjmu zprávy. Pokud není povolena parita, může přijatá zpráva obsahovat nesprávné znaky. K této situaci může dojít, když je podmínka začátku zprávy určena jako konkrétní startovací znak nebo libovolný znak popsaný v bodu 2. a bodu 6. níže.

Instrukce ”Příjem” podporuje několik podmínek začátku zprávy. Problém specifikace začátku zprávy, který zahrnuje rozpoznání BREAK (přestávky) nebo prázdného úseku řešíme tak, že přinutíme synchronizovat začátek zprávy se začátkem znaku dříve, než se znaky umístí do zásobníku.

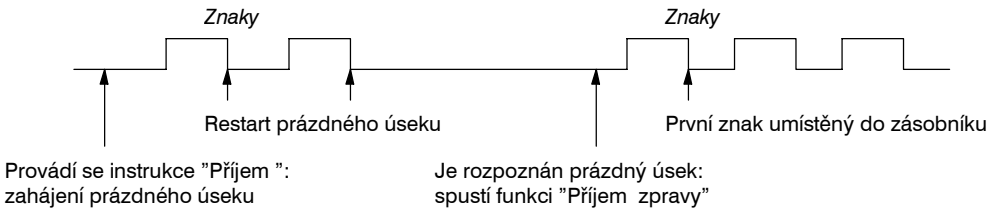
Instrukce ”Příjem” podporuje několik podmínek začátku zprávy:

1. *Rozpoznání prázdného úseku:* Prázdný úsek je definován jako doba ”ticha” nebo nečinnosti přenosového vedení. Příjem je zahájen v případě, že komunikační linka nebyla v činnosti určitý počet milisekund, specifikovaný v SMW90 nebo SMW190. Když se provádí instrukce ”Příjem” uživatelského programu, iniciuje funkce příjmu zprávy vyhledávání podmínky prázdného úseku. Pokud jsou přijaty nějaké znaky dříve, než vyprší doba prázdného úseku, funkce příjmu zprávy tyto znaky ignoruje a znovu spustí časovač prázdného úseku s časem z SMW90 nebo SMW190. Viz obrázek 6-11. Po vypršení času prázdného úseku uloží funkce příjmu zprávy všechny následující přijaté znaky do zásobníku.

   Doba prázdného úseku by vždy měla být delší než doba přenosu jednoho znaku (start bit, datové bity, parita a stop bity) při dané přenosové rychlosti. Běžná hodnota doby prázdného úseku je trojnásobek doby přenosu znaku při dané přenosové rychlosti.

   Rozeznání prázdného úseku se používá jako podmínka pro zahájení binárních protokolů, kde neexistuje konkrétní startovací znak, nebo tam, kde protokol specifikuje minimální čas mezi zprávami.

   Nastavení: il = 1, sc = 0, bk = 0, SMW90/SMW190 = časová prodleva prázdného úseku v milisekundách



Obr. 6-11 Použití rozeznání prázdného úseku pro spuštění instrukce ”Příjem”

2. *Rozpoznání startovacího znaku:* Startovací znak je libovolný znak použitý jako první znak zprávy. Zpráva začíná, když je přijat startovací znak specifikovaný v SMB88 nebo SMB188. Funkce příjmu zprávy uloží startovací znak v příjmovém zásobníku jako první znak zprávy. Tato funkce ignoruje jakékoliv znaky přijaté před startovacím znakem. Startovací znak a všechny znaky přijaté po něm se uloží do zásobníku.

   Rozpoznání startovacího znaku se běžně používá pro protokoly ASCII, ve kterých všechny zprávy začínají stejným znakem.

   Nastavení: il = 0, sc = 1, bk = 0, SMW90/SMW190 = není podstatný, SMB88/SMB188 = startovací znak

3. *Prázdný úsek a startovací znak:* Při instrukci "Příjem" může zpráva začínat kombinací prázdného úseku a startovacího znaku. Když se provádí instrukce "Příjem", vyhledává funkce příjmu zprávy podmínku prázdného úseku. Po nalezení prázdného úseku hledá funkce příjmu zprávy specifikovaný startovací znak. Jestliže je nalezen jakýkoliv jiný znak než startovací, znovu se zahájí vyhledávání podmínky prázdného úseku. Všechny znaky přijaté před tím, než je splněna podmínka prázdného úseku a než je přijat startovací znak, jsou ignorovány. Startovací znak je umístěn do zásobníku spolu se všemi následujícími znaky.

   Doba prázdného úseku by vždy měla být delší než doba přenosu jednoho znaku (start bit, datové bity, parita a stop bity) při dané přenosové rychlosti. Běžná hodnota doby prázdného úseku je trojnásobek doby přenosu znaku při dané přenosové rychlosti.

   Běžně se tento typ startovací podmínky užívá u protokolu, který specifikuje minimální čas mezi zprávami a první znak je adresa nebo něco, co specifikuje konkrétní zařízení. Nejužitečnější je to při implementaci protokolu, kdy je na komunikační spoj připojeno více zařízení. V tomto případě instrukce "Příjem" spustí přerušení pouze tehdy, je–li přijata zpráva pro konkrétní adresu nebo zařízení specifikované startovacím znakem.

   Nastavení: il = 1, sc = 1, bk = 0, SMW90/SMW190 > 0, SMB88/SMB188 = startovací znak

4. *Rozpoznání BREAK (přestávky):* Přestávka je indikována tehdy, když přijímaná data setrvají na nulové hodnotě po dobu delší, než je doba přenosu úplného znaku. Doba přenosu úplného znaku se definuje jako celková doba start bitu, datových bitu, parity a stop bitu. Pokud je instrukce "Příjem" konfigurována, aby zahájila zprávu při příjmu podmínky přestávky, jsou jakékoliv znaky přijaté po podmínce přestávky umístěny do zásobníku. Jakékoliv znaky přijaté před přestávkou jsou ignorovány.

   Běžně se rozpoznání přestávky používá jako podmínka pro začátek zprávy pouze tehdy, když to vyžaduje protokol.

   Nastavení: il = 0, sc = 0, bk = 1, SMW90/SMW190 = není podstatný, SMB88/SMB188 = není podstatný

5. *Znak BREAK (přestávky) a startovací znak:* Instrukci "Příjem" je možné konfigurovat tak, aby začala přijímat znaky po přijetí sledu přestávky a následujícího konkrétního startovacího znaku. Po podmínce přestávky se hledá specifikovaný startovací znak. Jestliže je nalezen jakýkoliv jiný znak než startovací, znovu se zahájí vyhledávání podmínky přestávky. Všechny znaky přijaté před splněním podmínky přestávky a před příjmem startovacího znaku jsou ignorovány. Startovací znak je umístěn do zásobníku spolu se všemi následujícími znaky.

   Nastavení: il = 0, sc = 1, bk = 1, SMW90/SMW190 = není podstatný, SMB88/SMB188 = startovací znak

6. *Libovolný znak:* Instrukci "Příjem" je možné nakonfigurovat tak, aby zahájila okamžitý příjem jakýchkoliv znaků a umístila je do zásobníku. Jde o speciální případ rozpoznání prázdného úseku. V tomto případě je čas prázdného úseku (SMW90 nebo SMW190) nastaven na nulu. To nutí instrukci "Příjem" zahájit příjem znaku okamžitě po provedení.

   Nastavení: il = 1, sc = 0, bk = 0, SMW90/SMW190 = 0, SMB88/SMB188 = není podstatný

   To, že zpráva začíná libovolným znakem, umožňuje použití časovače pro načasování příjmu zprávy. To je užitečné v případech, kdy se režim Freeport použije pro implementaci části protokolu "master" nebo "host" a je zapotřebí prodleva, pokud ze slave nepřijde ve specifikovaném časovém úseku odezva. Časovač zahájí činnost při provádění instrukce "Příjem", protože doba prázdného úseku byla nastavena na nulu. Časovač pozdrží a ukončí funkci příjmu, pokud není splněna žádná jiná podmínka pro její ukončení.

   Nastavení: il = 1, sc = 0, bk = 0, SMW90/SMW190 = 0, SMB88/SMB188 = není podstatný, c/m = 1, tmr = 1, SMW92 = prodleva (timeout) v milisekundách

Instrukce ”Příjem” podporuje několik způsobů ukončení zprávy. Zprávu může ukončit jedno nebo kombinace následujících kritérií:

1. *Rozpoznání ukončovacího znaku:* Ukončovací znak je libovolný znak použitý pro označení konce. Po nalezení podmínky pro začátek zprávy kontroluje instrukce ”Příjem” každý další přijatý znak, zda odpovídá ukončovacímu znaku. Když je přijat ukončovací znak, je uložen do zásobníku a příjem je ukončen.

   Běžně se rozpoznání ukončovacího znaku používá v protokolech ASCII, kde každá zpráva končí specifickým znakem. Pro ukončení můžete rozpoznání ukončovacího znaku použít v kombinaci s časovačem měřícím čas mezi znaky, časovačem zprávy nebo maximálním počtem přijatých znaků.

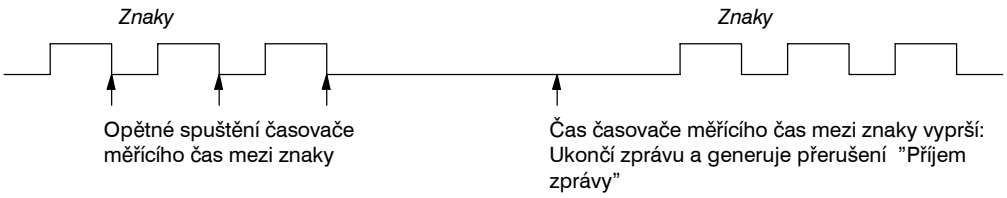
   Nastavení: ec = 1, SMB89/SMB189 = ukončovací znak

2. *Časovač měřící čas mezi znaky:* Čas mezi znaky je doba naměřená od konce jednoho znaku (stop bit) do konce následujícho znaku (stop bit). Pokud čas mezi znaky (včetně druhého znaku) překročí počet milisekund specifikovaných v SMW92 nebo SMW192, je funkce příjmu zprávy ukončena. Časovač měřící čas mezi znaky je znovu spuštěn po každém přijatém znaku. Viz obrázek 6-12.

   Časovač měřící čas mezi znaky je možné použít pro ukončení zprávy v protokolech, které nemají konkrétní ukončovací znak. Tento časovač musí být nastaven na hodnotu vyšší, než je doba jednoho znaku při zvolené přenosové rychlosti, protože tento časovač vždy zahrnuje čas pro příjem jednoho celého znaku (start bit, datové bity, paritní bit a stop bity).

   Pro ukončení zprávy můžete použít časovač měřící čas mezi znaky v kombinaci s rozpoznáním ukončovacího znaku a s maximálním počtem přijatých znaků.

   Nastavení: c/m = 0, tmr = 1, SMW92/SMW192 = prodleva (timeout) v milisekundách



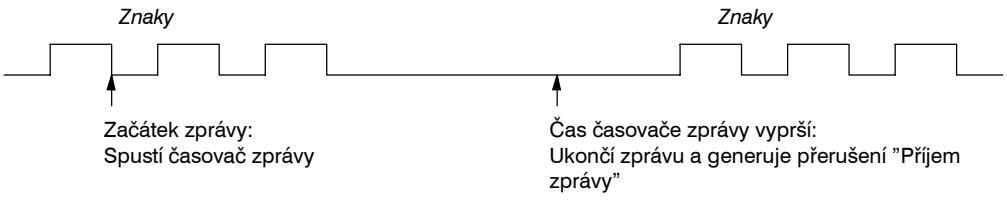
Obr. 6-12 Použití časovače měřícího čas mezi znaky pro ukončení instrukce ”Příjem”

3. *Časovač zprávy:* Časovač zprávy ukončí zprávu ve specifikovaný čas po začátku zprávy. Časovač zprávy zahájí činnost, jakmile byla splněna/byly splněny podmínka/podmínky pro začátek zprávy funkce příjmu zprávy. Doba časovače zprávy vyprší po uplynutí určitého počtu milisekund, specifikovaných v SMW92 nebo SMW192. Viz obrázek 6-13.

   Časovač zprávy se běžně používá tehdy, nemůže–li komunikační zařízení zaručit, že nenastanou časové mezery mezi znaky, nebo při použití modemů. U modemů můžete časovač zprávy využít pro zadání maximální doby povolené pro příjem po jejím zahájení. Typická hodnota časovače zprávy bude asi 1,5násobek doby potřebné pro příjem nejdelší možné zprávy při dané přenosové rychlosti.

   Pro ukončení zprávy můžete použít časovač zprávy v kombinaci s rozpoznáním ukončovacího znaku a s maximálním počtem přijatých znaků.

   Nastavení: c/m = 1, tmr = 1, SMW92/SMW192 = prodleva (timeout) v milisekundách



Obr. 6-13 Použití časovače zprávy pro ukončení instrukce "Příjem"

4. *Maximální počet přijatých znaků:* V instrukci "Příjem" musí být zadán maximální počet přijatých znaků (SMB94 nebo SMB194). Když je této hodnoty dosaženo nebo když je překročena, je funkce příjmu zprávy ukončena. Instrukce "Příjem" vyžaduje, aby uživatel specifikoval maximální počet přijatých znaku, i když to není přímo použito jako podmínka pro ukončení zprávy. Důvodem je to, že instrukce "Příjem" potřebuje znát maximální velikost přijímané zprávy, aby nebyla přepsána uživatelská data umístěná za zásobníkem.

   Maximální počet přijatých znaků se používá pro ukončení zpráv v protokolech, kde je délka zprávy známá a vždy stejná. Maximální počet přijatých znaků se vždy používá v kombinaci s rozpoznáním ukončovacího znaku, časovačem měřícím čas mezi znaky nebo s časovačem zprávy.

5. *Chyby parity:* Instrukce "Příjem" je automaticky ukončena, když hardware signalizuje chybu parity u přijatého znaku. Chyby parity jsou možné pouze tehdy, jestliže je parita povolena v SMB30 nebo SMB130. Neexistuje způsob, jak tuto funkci zablokovat.

6. *Ukončení uživatelem:* Uživatelský program může ukončit funkci příjmu zprávy provedením jiné instrukce "Příjem" s povolovacím bitem (EN) v SMB87 nebo SMB187 nastaveným na nulu. Tím je příjem zprávy okamžitě ukončen.

Použití přerušení pro řízení příjmu dat

Pro umožnění úplné flexibility v podporovaných protokolech je možné přijímat data pomocí přerušení. Každý přijatý znak generuje přerušení. Přijatý znak se uloží do SMB2 a parita (pokud je povolená) se uloží do SM3.0 bezprostředně před provedením podprogramu přerušení přiřazeného k události přijetí znaku. SMB2 je zásobník pro znak přijatý z Freeportu. Každý znak přijatý během režimu Freeport je umístěn na tuto adresu kvůli jednoduchému přístupu z uživatelského programu. SMB3 se používá pro režim Freeport a obsahuje bit chyby parity, který se aktivuje při detekci chyby parity na přijatém znaku. Všechny ostatní bity tohoto bytu jsou rezervované. Paritní bit se používá pro vyřazení zprávy nebo pro generování negativního potvrzení zprávy.

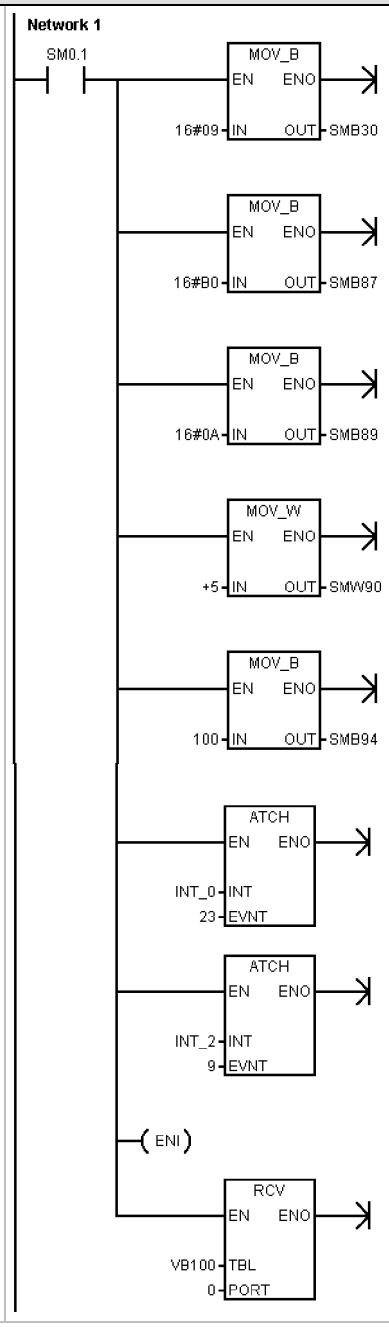
Jestliže se přerušení použije při vysokých přenosových rychlostech (38,4 kbaud až 115,2 kbaud), je doba mezi přerušeními velmi krátká. Například přerušení pro 38,4 kbaud je 260 mikrosekund, pro 57,6 kbaud je 173 mikrosekund a pro 115,2 kbaud je 86 mikrosekund. Abyste zabránili vynechání znaků, zajistěte, aby byly podprogramy přerušení vždy velmi krátké nebo použijte instrukci "Příjem".



Tip

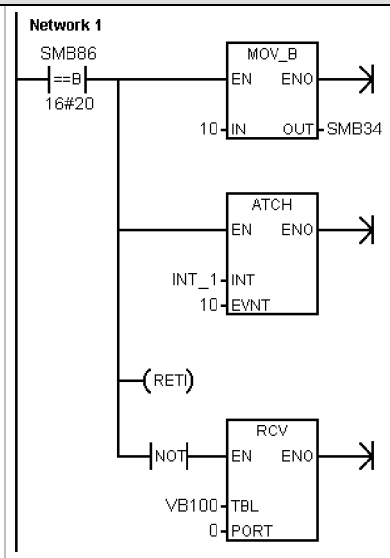
SMB2 a SMB3 jsou sdíleny mezi portem 0 a portem 1. Po přijetí znaku se na portu 0 provede podprogram přerušení přiřazený k této události (událost 8), SMB2 obsahuje znak přijatý na portu 0 a SMB3 obsahuje stav parity přijatého znaku. Pokud po přijetí znaku na portu 1 provede podprogram přerušení přiřazený k této události (událost 25), SMB2 obsahuje znak přijatý na portu 1 a SMB3 obsahuje stav parity přijatého znaku.

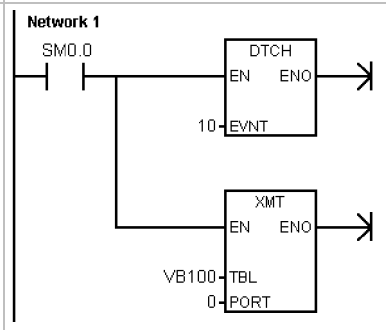
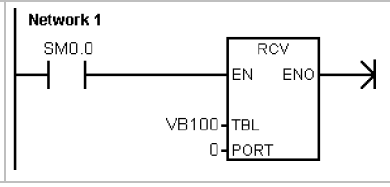
Příklad: Instrukce vysílání a příjmu




```
Network 1       //Tento program přijímá řetězec znaků
                //až do příjmu znaku LF (line feed).
                //Zpráva je pak vyslána zpět
                //vysílači.
LD      SM0.1           //V prvním programovém cyklu:
MOVB    16#09, SMB30    //1.  Inicializace Freeport:
                        //    – Rychlost 9600 baud.
                        //    – 8 datových bitů.
                        //    – Bez parity.
MOVB    16#B0, SMB87    //2.  Inicializace řídicího byte pro RCV:
                        //    – Povolení RCV.
                        //    – Nastavení rozpoznání znaku pro
                        //    ukončení.
                        //    – Nastavit rozpoznání prázdného
                        //    úseku
                        //    jako podmínky začátku zprávy.
MOVB    16#0A, SMB89    //3.  Nastavení ukončovacího znaku
                        //    na OA hexa (line feed).
MOVW    +5, SMW90       //4.  Nastavení prodlevy prázdného
                        //    úseku
                        //    na 5 ms.
MOVB    100, SMB94      //5.  Nastavení maximálního počtu znaků
                        //    na 100.
ATCH    INT_0, 23       //6.  Přiřazení přerušení 0
                        //    k události "Příjem dokončen".
ATCH    INT_2, 9        //7.  Přiřazení přerušení 2
                        //    k události "Vysílání dokončeno".
ENI                     //8.  Povolení uživatelských přerušení.
RCV     VB100, 0        //9.  Povolení příjmu krabice se
                        //    zásobníkem na VB100.
```

Příklad: Instrukce vysílání a příjmu, pokračování

INT 0		Network 1 //Podprogram přerušení "Příjem dokončen":
		//1. Pokud stav příjmu signalizuje příjem
		// ukončovacího znaku, pak přiřadí
		// 10ms časovač na spuštění vysílání a návratu.
		//2. Pokud byl příjem ukončen z jakéhokoliv jiného
		// důvodu, spustí nový příjem.
		LDB= SMB86, 16#20
		MOVB 10, SMB34
		ATCH INT_1, 10
		CRETI
		NOT
		RCV VB100, 0

INT 1		Network 1 //Přerušení 10ms časovače: //1. Odpojí časové přerušení. //2. Vyšle zprávu zpět uživateli na portu. LD SM0.0 DTCH 10 XMT VB100, 0
INT 2		Network 1 //Přerušení "Vysílání dokončeno": //Povolí další příjem. LD SM0.0 RCV VB100, 0

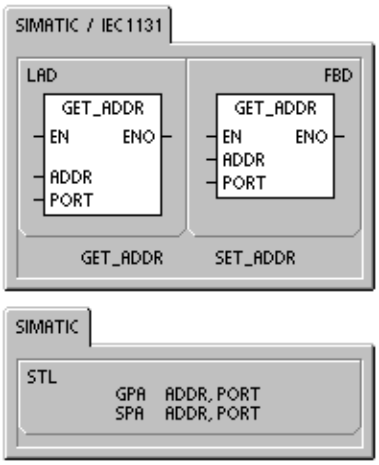
Zjištění a nastavení adresy portu

Instrukce ”Zjistit adresu portu” (GPA) přečte adresu stanice z portu CPU S7-200 daného v PORT a hodnotu zapíše na adresu ADDR.

Instrukce ”Nastavit adresu portu” (SPA) nastavuje adresu portu stanice (PORT) na hodnotu danou v ADDR. Nová adresa není uložena trvale. Po výpadku a obnovení napájení se daný port vrátí k poslední adrese (adrese, která byla nahrána společně se systémovým blokem).

Chyby, které nastavují ENO = 0:

- 0006 (nepřímá adresa)
- 0004 (pokus provést instrukci ”Nastavit adresu portu” v přerušovacím podprogramu)



Tabulka 6-14 Operandy platné pro instrukce ”Zjistit adresu portu” a ”Nastavit adresu portu”

Vstupy/Výstupy	Typ dat	Operandy		
ADDR	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta *(Konstantní hodnota platí pouze pro instrukci ”Nastavit adresu portu”.)*		
PORT	BYTE	Konstanta	*pro CPU 221, CPU 222, CPU 224:* *pro CPU 224XP, CPU 226:*	0 0 nebo 1

Instrukce porovnávání

Porovnávání číselných hodnot

Instrukce porovnávání se používají pro porovnání dvou hodnot:

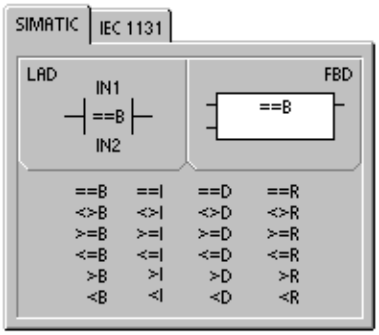
IN1 = IN2 IN1 >= IN2 IN1 <= IN2
IN1 > IN2 IN1 < IN2 IN1 <> IN2

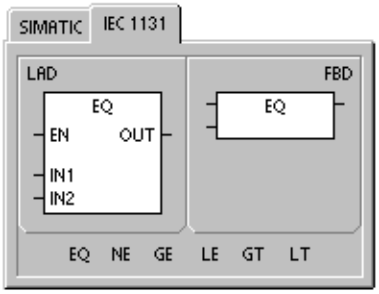
Operace porovnání bytu jsou bez znaménka.
Operace porovnání celých čísel (integer) jsou se znaménkem.
Operace porovnání double word jsou se znaménkem.
Operace porovnání reálných čísel jsou se znaménkem.

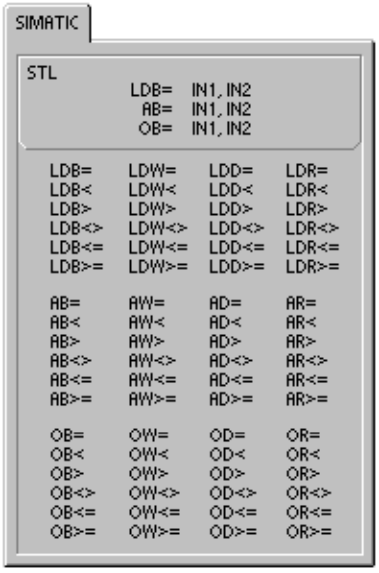
V LAD a FBD: Je–li výsledek porovnání pravdivý, instrukce nastaví kontakt (LAD) nebo výstup (FBD) na hodnotu log. 1.

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

Použijete–li instrukce porovnávání IEC, můžete pro vstupy použít různé typy dat. Obě vstupní hodnoty však musí být stejného typu.







Poznámka

Následující podmínky jsou kritické chyby, které způsobí, že programovatelný automat S7-200 okamžitě zastaví provádění programu:

- Nepřípustná nepřímá adresa (kterákoliv instrukce porovnávání)
- Nepřípustné reálné číslo (například NAN) (instrukce porovnání reálných čísel)

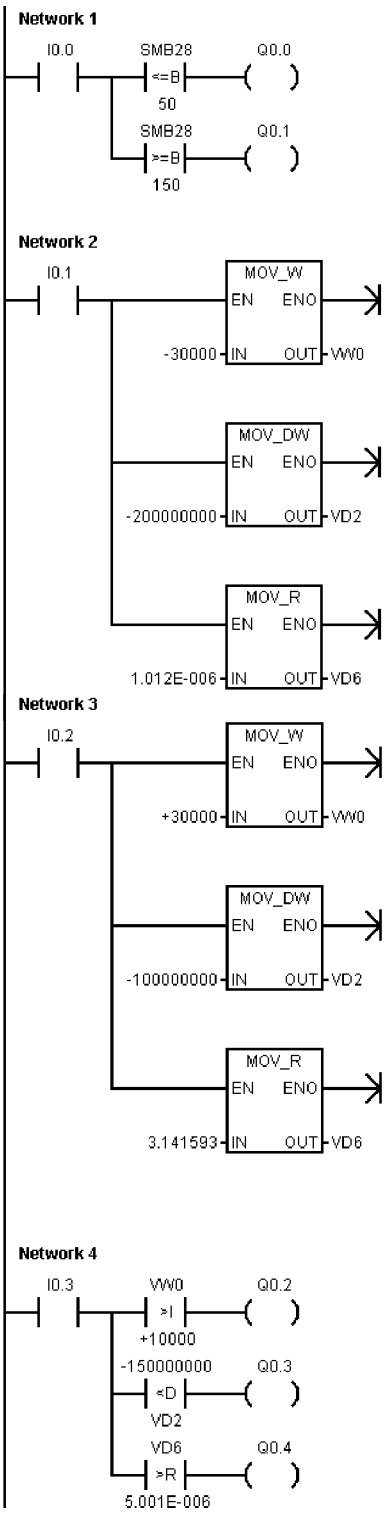
Abyste zabránili vzniku těchto chyb, zjistěte správnou inicializaci ukazatele a hodnoty, které obsahují reálná čísla, dříve než vykoná instrukce porovnání.

Instrukce porovnávání se provádějí bez ohledu na stav signálového toku.

Tabulka 6-15 Operandy platné pro instrukce porovnávání

Vstupy/Výstupy	Typ	Operandy
IN1, IN2	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, HC, *VD, *LD, *AC, konstanta
	REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC, konstanta
Výstup (neboli OUT)	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok

Příklad: Porovnávání



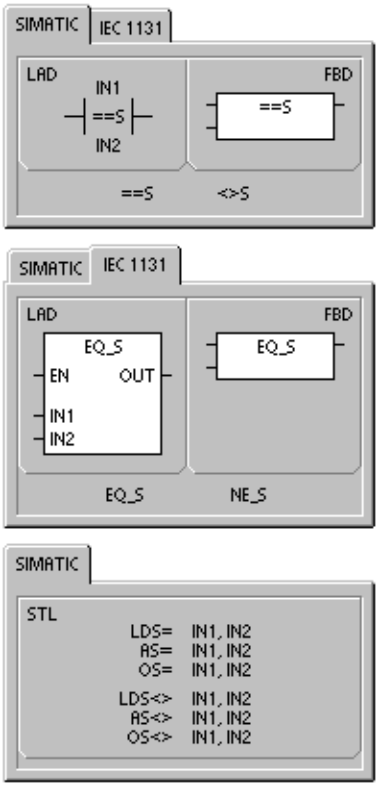

```
Network 1       //Otáčejte potenciometrem 0,
                //abyste změnili hodnotu bytu SMB28.
                //Q0.0 je aktivní, když je hodnota SMB28
                //menší nebo rovná 50.
                //Q0.1 je aktivní, když je hodnota SMB28
                //větší nebo rovná 150.
                //Stavový indikátor je zapnutý, pokud
                //platí výsledek porovnání.
LD      I0.0
LPS
AB<=    SMB28, 50
=       Q0.0
LPP
AB>=    SMB28, 150
=       Q0.1

Network 2       //Nahraje adresy paměti V
                //s nízkými hodnotami, které činí porovnání
                //neplatnými a které
                //vypínají stavové indikátory.
LD      I0.1
MOVW    –30000, VW0
MOVD    –200000000, VD2
MOVR    1.012E–006, VD6

Network 3       //Nahraje adresy paměti V
                //s vysokými hodnotami, které činí
                //porovnání platnými a které
                //zapínají stavové indikátory.
LD      I0.2
MOVW    +30000, VW0
MOVD    –100000000, VD2
MOVR    3,141593, VD6

Network 4       //Porovnání "Integer Word" zjistí,
                //zda VW0 > +10000 platí.
                //Užívá konstanty programu pro zobrazení
                //různých typů dat. Můžete také
                //porovnat dvě hodnoty
                //uložené v programovatelné paměti
                //např.: VW0 > VW100
LD      I0.3
LPS
AW>     VW0, +10000
=       Q0.2
LRD
AD<     –150000000, VD2
=       Q0.3
LPP
AR>     VD6, 5.001E–006
=       Q0.4
```

Porovnání řetězců



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

IN1 = IN2 IN1 <> IN2

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

Poznámka

Následující podmínky jsou kritické chyby, které způsobí, že programovatelný automat S7-200 okamžite zastaví provádění programu:

- Nepřípustná nepřímá adresa (kterákoliv instrukce porovnávání)
- Řetězec delší než 254 znaků (Instrukce ”Porovnání řetězců”)
- Řetězec s takovou počáteční adresou a délkou, které se nevejdou do dané paměťové oblasti (Instrukce ”Porovnání řetězců”)

Aby k tomu nedošlo, zkontrolujte před prováděním instrukcí porovnání, jaké hodnoty používají a zda jste správně inicializovali ukazatele a místa paměti, které jsou určeny pro řetězce ASCII. Zkontrolujte, zda se zásobník rezervovaný pro řetězec ASCII celý vejde do dané paměťové oblasti.

Instrukce porovnávání se provádějí bez ohledu na stav signálového toku.

Tabulka 6-16 Operandy platné pro instrukce porovnávání řetězce

Vstupy/Výstupy	Typ	Operandy
IN1	ŘETĚZEC	VB, LB, *VD, *LD, *AC, konstanta
IN2	ŘETĚZEC	VB, LB, *VD, *LD, *AC
Výstup (OUT)	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok

Instrukce pro konverzi

Standardní konverze

Číselné konverze

Instrukce ”Byte na integer” (BTI), ”Integer na byte” (ITB), ”Integer na double integer” (ITD), ”Double integer na integer” (DTI), ”Double integer na real” (DTR), ”BCD na integer” (BCDI) a ”Integer na BCD” (IBCD) konvertují vstupní hodnotu IN na specifikovaný formát a ukládají výstupní hodnotu v místě paměti specifikovaném v OUT. Můžete například konvertovat double integer na reálné číslo. Můžete také provádět konverzi mezi formáty integer a BCD.

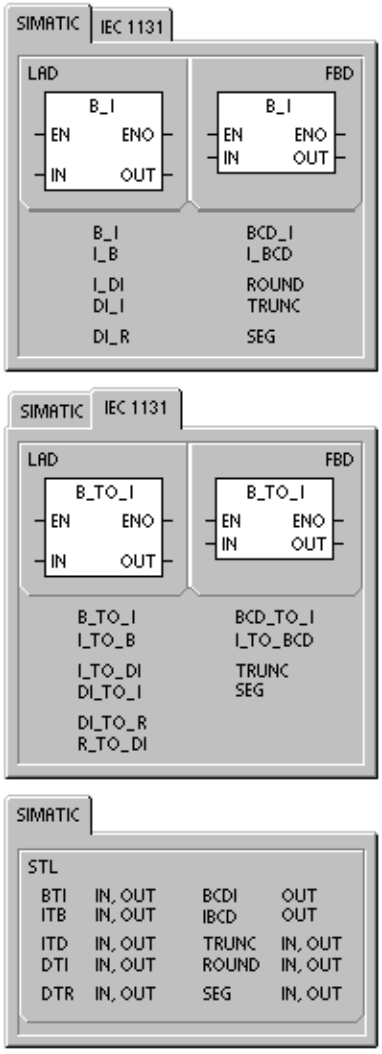
Zaokrouhlení a oříznutí

Instrukce ”Zaokrouhlit” (ROUND) konvertuje reálnou hodnotu IN na double integer a zaokrouhlený výsledek uloží do proměnné specifikované v OUT.

Instrukce ”Oříznout” (TRUNC) konvertuje reálné číslo IN na double integer a celočíselnou část výsledku zapíše do proměnné dané v OUT.

Segment

Instrukce ”Segment” (SEG) generuje bitový vzor, který můžete použít pro řízení sedmisegmentového displeje.



Tabulka 6-17 Operandy platné pro instrukce standarní konverze

Vstupy/Výstupy	Typ dat	Operandy
IN	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
	WORD, INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AIW, AC, *VD, *LD, *AC, konstanta
	DINT	ID, QD, VD, MD, SMD, SD, LD, HC, AC, *VD, *LD, *AC, konstanta
	REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC, konstanta
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC
	WORD, INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, *VD, *LD, *AC
	DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

Činnost instrukce pro konverzi BCD na Integer a Integer na BCD

Instrukce "BCD na Integer" (BCDI) konvertuje vstupní hodnotu v BCD kódu (IN) na celočíselnou hodnotu a výsledek nahraje do proměnné specifikované v OUT. Platný rozsah pro IN je v BCD kódu 0 až 9999.

Instrukce "Integer na BCD" (IBCD) konvertuje vstupní hodnotu integer (IN) na hodnotu v BCD kódu a výsledek zapíše do proměnné dané v OUT. Platný rozsah pro IN je od 0 do 9999 v celých číslech.

Chyby, které nastavují ENO = 0
- SM1.6 (BCD chyba)
- 0006 (nepřímá adresa)

Ovlivněné bity SM:
- SM1.6 (BCD chyba)

Činnost instrukce pro konverzi double integer na reálné číslo

Instrukce "Double integer na real" (DTR) konvertuje 32bitové číslo integer se znaménkem (IN) na 32bitové reálné číslo a výsledek zapíše do proměnné dané v OUT.

Chyby, které nastavují ENO = 0
- 0006 (nepřímá adresa)

Činnost instrukce pro konverzi double integer na integer

Instrukce "Double integer na integer" (DTI) konvertuje double integer (IN) na integer a výsledek zapíše do proměnné dané v OUT.

Pokud je konvertovaná hodnota příliš velká pro reprezentaci na výstupu, nastaví se bit přeplnění a výstup se nezmění.

Chyby, které nastavují ENO = 0
- SM1.1 (Přetečení)
- 0006 (nepřímá adresa)

Ovlivněné bity SM:
- SM1.1 (Přetečení)

Činnost instrukce pro konverzi integer na double integer

Instrukce "Integer na double integer" (ITD) konvertuje integer (IN) na double integer a výsledek zapíše do proměnné dané v OUT. Přidá se znaménkový bit.

Chyby, které nastavují ENO = 0
- 0006 (nepřímá adresa)

Činnost instrukce pro konverzi byte na integer

Instrukce "Byte na integer" (BTI) konvertuje byte IN na integer a výsledek zapíše do proměnné dané v OUT. Byte nemá znaménko, proto se nepoužívá znaménkové rozšíření.

Chyby, které nastavují ENO = 0
- 0006 (nepřímá adresa)

Činnost instrukce pro konverzi integer na byte

Instrukce "Integer na byte" (ITB) konvertuje hodnotu word IN na byte a výsledek zapíše do proměnné dané v OUT. Konvertují se hodnoty 0 až 255. Všechny ostatní hodnoty mají za následek přeplnění a výstup není ovlivněn.

Chyby, které nastavují ENO = 0
- SM1.1 (Přetečení)
- 0006 (nepřímá adresa)

Ovlivněné bity SM:
- SM1.1 (Přetečení)



Tip

Chcete–li změnit integer na reálné číslo, použijte instrukci "Integer na double integer" a potom instrukci "Double integer na real".

Činnost instrukcí ”Zaokrouhlit” a ”Oříznout”

Instrukce ”Zaokrouhlit” (ROUND) konvertuje hodnotu reálného čísla IN na double integer a výsledek uloží do proměnné dané v OUT. Pokud je za desetinnou čárkou 0,5 nebo větší, je číslo zaokrouhleno nahoru.

Instrukce ”Oříznout” (TRUNC) konvertuje hodnotu reálného čísla IN na double integer a výsledek uloží do proměnné dané v OUT. Je konvertována pouze celočíselná část reálného čísla, za desetinnou čárkou se část vynechá.

Chyby, které nastavují ENO = 0

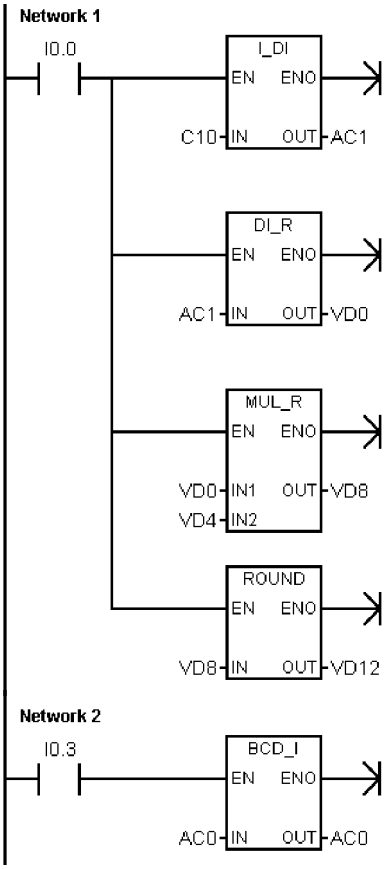
- SM1.1 (Přetečení)
- 0006 (nepřímá adresa)

Ovlivněné bity SM:

- SM1.1 (Přetečení)

Pokud konvertovaná hodnota není platné reálné číslo nebo je příliš velká pro reprezentaci na výstupu, nastaví se bit přetečení a výstup se nezmění.

Příklad: Instrukce pro standardní konverzi




```
Network 1      //Konvertuje palce na centimetry:
               //1. Nahraje hodnotu z čítače (palce) do AC1.
               //2. Konvertuje hodnotu na reálné číslo.
               //3. Vynásobí 2,54 (převod na centimetry).
               //4. Konvertuje hodnotu zpět na integer.
LD     I0.0
ITD    C10, AC1
DTR    AC1, VD0
MOVR   VD0, VD8
*R     VD4, VD8
ROUND  VD8, VD12

Network 2      //Konvertuje hodnotu v BCD na integer
LD     I0.3
BCDI   AC0
```

Double word integer na reálné číslo a zaokrouhlení			BCD na celé číslo	
C10	101	Počet = 101 palců	AC0	1234
VD0	101.0	Počet (jako reálné číslo)		BCDI
VD4	2.54	Konstanta 2,54 (palce na centimetry)	AC0	04D2
VD8	256.54	256,54 centimetrů jako reálné číslo		
VD12	257	257 centimetrů jako double integer		

Činnost instrukce ”Segment”

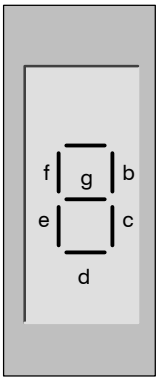
Instrukce ”Segment” (SEG) konvertuje segmenty sedmisegmentového displeje tak, že znak specifikovaný v IN generuje bitový vzorec (byte) na místě daném v OUT.

Označené segmenty reprezentují znak na nejnižší platné číslici vstupního bytu. Obrázek 6-14 ukazuje kódování sedmisegmentového displeje s instrukcí ”Segment”.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)

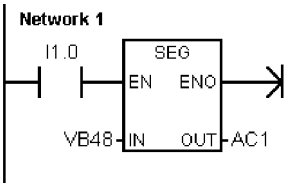
(IN) LSD	Segmentový displej	(OUT) – g f e d c b a
0		0 0 1 1 1 1 1 1
1		0 0 0 0 0 1 1 0
2		0 1 0 1 1 0 1 1
3		0 1 0 0 1 1 1 1
4		0 1 1 0 0 1 1 0
5		0 1 1 0 1 1 0 1
6		0 1 1 1 1 1 0 1
7		0 0 0 0 0 1 1 1



(IN) LSD	Segmentový displej	(OUT) – g f e d c b a
8		0 1 1 1 1 1 1 1
9		0 1 1 0 0 1 1 1
A		0 1 1 1 0 1 1 1
B		0 1 1 1 1 1 0 0
C		0 0 1 1 1 0 0 1
D		0 1 0 1 1 1 1 0
E		0 1 1 1 1 0 0 1
F		0 1 1 1 0 0 0 1

Obr. 6-14 Kódování sedmisegmentového displeje-

Příklad: Instrukce ”Segment”




```
Network 1
LD      I1.0
SEG     VB48, AC1
```

05 SEG 6D

VB48 AC1

(znak displeje)

Konverze na ASCII

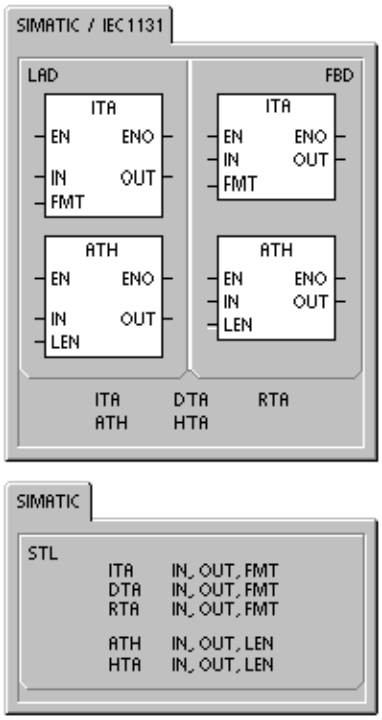
Platné znaky ASCII jsou hexadecimální hodnoty 30 až 39 a 41 až 46.

Konverze mezi ASCII a hexadecimálními hodnotami

Instrukce ”ASCII na hexadecimální” (ATH) konvertuje počet (LEN) ASCII znaku začínající na IN, na hexadecimální číslice začínající v OUT. Instrukce ”Hexadecimální na ASCII” (HTA) konvertuje hexadecimální číslice začínající vstupním bytem IN na znaky ASCII začínající v OUT. Počet konvertovaných hexadecimálních číslic je dán délkou LEN.

Maximální počet znaků ASCII nebo hexadecimálních číslic, které je možno konvertovat, je 255.

Platné vstupní znaky ASCII jsou alfanumerické znaky 0 až 9 s hodnotou hexadecimálního kódu 30 až 39 a velká písmena A až F s hodnotou hexadecimálního kódu 41 až 46.



Chyby, které nastavují ENO = 0

- SM1.7 (naplatné ASCII) *pouze u ”ASCII na hexadecimální”*
- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)

Ovlivněné bity SM:

- SM1.7 (neplatné ASCII)

Konverze číselných hodnot na ASCII

Instrukce ”Integer na ASCII” (ITA), ”Double integer na ASCII” (DTA) a ”Real na ASCII” (RTA) konvertují číselné hodnoty integer, double integer nebo reálného čísla na znaky ASCII.

Tabulka 6-18 Operandy platné pro instrukce konverze ASCII

Vstupy/Výstupy	Typ dat	Operandy
IN	BYTE	IB, QB, VB, MB, SMB, SB, LB, *VD, *LD, *AC
	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, HC, *VD, *LD, *AC, konstanta
	REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC, konstanta
LEN, FMT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, *VD, *LD, *AC

Činnost instrukce pro konverzi integer na ASCII

Instrukce ”Integer na ASCII” (ITA) konvertuje integer word IN na řetězec znaků ASCII. Formát FMT udává přesnost konverze napravo od desetinné čárky a to, má–li se zobrazovat desetinná čárka nebo tečka. Výsledná konverze je zapsaná do 8 za sebou jdoucích bytů počínaje OUT.

Řetězec ASCII má vždy 8 znaků.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- Neplatný formát
- *nnn* > 5

Obrázek 6-15 ukazuje formát operandu v instrukci ”Integer na ASCII”. Výstupní zásobník má vždy velikost 8 bytů. Počet číslic vpravo od desetinné čárky ve výstupním zásobníkuje dán v poli *nnn*. Platný rozsah pole *nnn* je 0 až 5. Zadáním počtu číslic 0 bude hodnota zobrazena bez desetinné čárky. Pro hodnoty *nnn* vyšší než 5 je výstupní zásobníku naplněn mezerami ASCII. V bitu *c* je dáno, zda má být pro oddělení celého čísla a desetinné části použita desetinná čárka (c=1) nebo desetinná tečka (c=0). Horní 4 bity musí být nulové.

Obrázek 6-15 ukazuje příklady hodnot zformátovaných s desetinnou tečkou (c=0) a třemi číslicemi vpravo od desetinné tečky (nnn=011). Výstupní zásobníku je formátován podle následujících pravidel:

- Kladné hodnoty se do výstupního zásobníku zapisují bez znaménka.
- Záporné hodnoty se do výstupního zásobníku zapisují se znaménkem minus (–) před nimi.
- Nuly vlevo od desetinné čárky (s výjimkou nuly těsně vedle desetinné čárky) se nezobrazují.
- Ve výstupním zásobníku jsou hodnoty zarovnané vpravo.

FMT

MSB 7	6	5	4	3	2	1	LSB 0
0	0	0	0	c	n	n	n

c = desetinná čárka (1), nebo desetinná tečka (0)
nnn = počet číslic vpravo od desetinné čárky

	Out	Out +1	Out +2	Out +3	Out +4	Out +5	Out +6	Out +7
in=12				0	.	0	1	2
in = –123			–	0	.	1	2	3
in=1234				1	.	2	3	4
in = –12345		–	1	2	.	3	4	5

Obr. 6-15 Operand FMT pro instrukci ”Integer na ASCII” (ITA)

Činnost instrukce pro konverzi double integer na ASCII

Instrukce ”Double integer na ASCII” (DTA) konvertuje double word IN na řetězec ASCII znaků. Operand formátu FMT udává přesnost konverze za desetinnou čárkou. Výsledná konverze je zapsaná do 12 za sebou jdoucích bytů počínaje OUT.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- Neplatný formát
- *nnn* > 5

Výstupní zásobník má vždy velikost 12 bytů.

Obrázek 6-16 ukazuje formát operandu v instrukci ”Double integer na ASCII”. Počet číslic vpravo od desetinné čárky ve výstupním zásobníkuje dán v poli *nnn*. Platný rozsah pole *nnn* je 0 až 5. Zadáním počtu číslic 0 bude hodnota zobrazena bez desetinné čárky. Pro hodnoty *nnn* vyšší než 5 je výstupní zásobník naplněn mezerami ASCII. V bitu *c* je dáno, zda má být pro oddělení celého čísla a desetinné části použita desetinná čárka (c=1) nebo desetinná tečka (c=0). Horní 4 bity musí být nulové.

Obrázek 6-16 ukazuje příklady hodnot zformátovaných s desetinnou tečkou (c=0) a čtyřmi číslicemi vpravo od desetinné tečky (nnn=100). Výstupní zásobník je formátován podle následujících pravidel:

- Kladné hodnoty se do výstupního zásobníku zapisují bez znaménka.
- Záporné hodnoty se do výstupního zásobníku zapisují se znaménkem minus (–) před nimi.
- Nuly vlevo od desetinné čárky (s výjimkou nuly těsně vedle desetinné čárky) se nezobrazují.
- Ve výstupním zásobníku jsou hodnoty zarovnané vpravo.

FMT

MSB 7	6	5	4	3	2	1	LSB 0
0	0	0	0	c	n	n	n

c = desetinná čárka (1), nebo desetinná tečka (0)
nnn = počet číslic vpravo od desetinné čárky

	Out	Out +1	Out +2	Out +3	Out +4	Out +5	Out +6	Out +7	Out +8	Out +9	Out +10	Out +11
in = –12						–	0	.	0	0	1	2
in=1234567					1	2	3	.	4	5	6	7

Obr. 6-16 Operand FMT pro instrukci ”Double integer na ASCII” (DTA)

Činnost instrukce pro konverzi reálného čísla na ASCII

Instrukce "Real na ASCII" (RTA) konvertuje hodnotu reálného čísla IN na řetězec ASCII. Formát FMT udává přesnost konverze napravo od desetinné čárky, zda se má zobrazovat desetinná čárka nebo tečka a velikost výstupního zásobníku.

Výsledná konverze je zapsaná do výstupního zásobníku začínajícího v OUT.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- nnn > 5
- ssss < 3
- ssss< počet znaků v OUT

Počet znaků (nebo délka) výsledného ASCII řetězce je shodný s velikostí výstupního zásobníku; může to být 3 až 15 bytů.

Formát reálného čísla použitý v S7-200 podporuje maximálně 7 platných číslic. Pokus o zobrazení více než 7 platných číslic způsobí chybu zaokrouhlení.

Obrázek 6-17 ukazuje formát operandu (FMT) pro instrukci RTA. Velikost výstupního zásobníku je dána polem ssss. Velikosti 0, 1 nebo 2 byty jsou neplatné. Počet číslic vpravo od desetinné čárky ve výstupním zásobníku je dán v poli nnn. Platný rozsah pole nnn je 0 až 5. Zadáním počtu číslic 0 bude hodnota zobrazena bez desetinné čárky. Výstupní zásobník je naplněn ASCII mezerami pro hodnoty nnn větší než 5 nebo pokud je zadaná velikost výstupního zásobníku příliš malá na vložení konvertované hodnoty. V bitu c je dáno, zda má být pro oddělení celého čísla a desetinné části použita desetinná čárka (c=1), nebo desetinná tečka (c=0).

Obrázek 6-17 také ukazuje příklady hodnot zformátovaných s desetinnou tečkou (c=0), jednou číslicí vpravo od desetinné tečky (nnn=001) a velikostí zásobníku šest bytů (ssss=0110). Výstupní zásobník je formátován podle následujících pravidel:

- ❏ Kladné hodnoty se do výstupního zásobníku zapisují bez znaménka.
- ❏ Záporné hodnoty se do výstupního zásobníku zapisují se znaménkem minus (–) před nimi.
- ❏ Nuly vlevo od desetinné čárky (s výjimkou nuly těsně vedle desetinné čárky) se nezobrazují.
- ❏ Hodnoty napravo od desetinné čárky jsou zaokrouhleny tak, aby měly daný počet číslic vpravo od desetinné čárky.
- ❏ Velikost výstupního zásobníku musí být alespoň o tři byty větší, než je počet číslic vpravo od desetinné čárky.
- ❏ Ve výstupním zásobníku jsou hodnoty zarovnané vpravo.

FMT

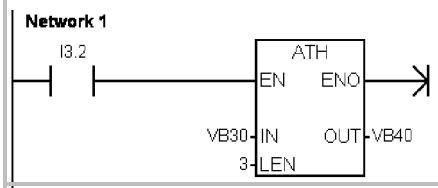
MSB							LSB
7	6	5	4	3	2	1	0
s	s	s	s	c	n	n	n

ssss = velikost výstupního zásobníku
c = desetinná čárka (1), nebo desetinná tečka (0)
nnn = počet číslic vpravo od desetinné čárky

	Out	Out +1	Out +2	Out +3	Out +4	Out +5
in = 1234,5	1	2	3	4	.	5
in = –0,0004				0	.	0
in = –3,67526			–	3	.	7
in = 1,95				2	.	0

Obr. 6-17 Operand FMT pro instrukci "Real na ASCII" (RTA)

Příklad: Instrukce ”ASCII na hexadecimální”

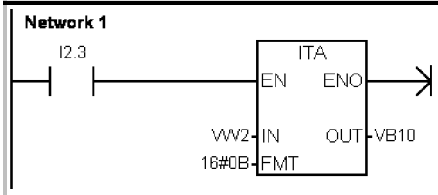



```
Network 1
LD      I3.2
ATH     VB30, VB40, 3
```

‘3’	‘E’	‘A’			
33	45	41	ATH	3E	Ax
VB30				VB40	

Poznámka: X naznačuje, že se ”nibble” (polovina bytu) nezměnil.

Příklad: Instrukce ”Integer na ASCII”

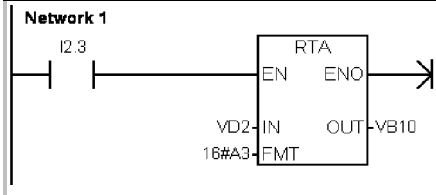



```
Network 1   //Konvertuje hodnotu integer ve VW2
            //na 8 ASCII znaků začínajících VB10,
            //s použitím formátu 16#0B (desetinná
            //čárka, po níž následují 3 číslice).

LD      I2.3
ITA     VW2, VB10, 16#0B
```

		‘ ’	‘ ’	‘1’	‘2’	‘,’	‘3’	‘4’	‘5’
12345	ITA	20	20	31	32	2C	33	34	35
VW2		VB10	VB11	...					

Příklad: Instrukce ”Real na ASCII”




```
Network 1   //Konvertuje hodnotu reálného čísla ve VD2
            //na 10 ASCII znaků začínajících VB10,
            //s použitím formátu 16#A3 (desetinná
            //tečka, po níž následují 3 číslice).

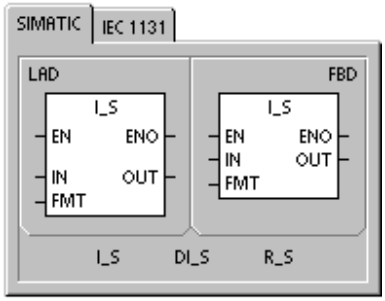
LD      I2.3
RTA     VD2, VB10, 16#A3
```

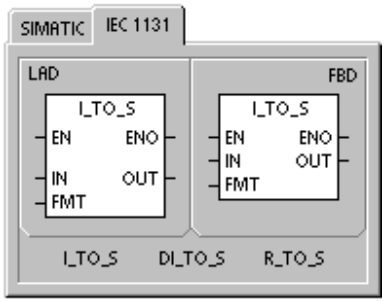
		‘ ’	‘ ’	‘ ’	‘1’	‘2’	‘3’	‘.’	‘4’	‘5’	‘0’
123.45	RTA	20	20	20	31	32	33	2E	34	35	30
VD2		VB10	VB11	...							

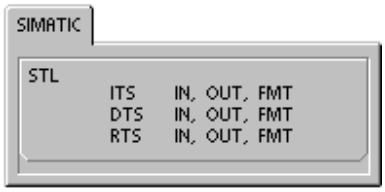
Konverze řetězců

Konverze číselných hodnot na řetězec

Instrukce ”Integer na řetězec” (ITS), ”Double integer na řetězec” (DTS) a ”Real na řetězec” (RTS) konvertují hodnoty integer, double integer nebo reálného čísla na řetězec znaků ASCII (OUT).







Činnost ”Integer na řetězec”

Instrukce ”Integer na řetězec” (ITS) konvertuje integer o velikosti word (IN) na řetězec ASCII o délce 8 znaků. Formát (FMT) udává přesnost konverze vpravo od desetinné čárky a to, má–li se zobrazovat desetinná čárka, nebo tečka. Výsledný řetězec je zapsán do 9 za sebou jdoucích bytů s počátkem v OUT. Více informací najdete v kapitole 4 v odstavci o formátu řetězců.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)
- Neplatný formát (nnn > 5)

Obrázek 6-18 ukazuje formát operandu v instrukci ”Double integer na řetězec”. Výstupní řetězec má vždy velikost 8 znaků. Počet číslic vpravo od desetinné čárky ve výstupním zásobníku je dán v poli nnn. Platný rozsah pole nnn je 0 až 5. Zadáním počtu číslic 0 bude hodnota zobrazena bez desetinné čárky. Pro hodnoty nnn vetší než 5 je výstupem řetězec 8 ASCII mezer. V bitu c je dáno, zda má být pro oddělení celého čísla a desetinné části použita desetinná čárka (c=1), nebo desetinná tečka (c=0). Horní 4 bity formátu musí být nulové.

Obrázek 6-18 také ukazuje příklady hodnot zformátovaných s desetinnou tečkou (c=0) a třemi číslicemi vpravo od desetinné tečky (nnn = 011). Hodnota v OUT je délka řetězce.

Výstupní řetězec je formátován podle následujících pravidel:

- ❏ Kladné hodnoty se do výstupního zásobníku zapisují bez znaménka.
- ❏ Záporné hodnoty se do výstupního zásobníku zapisují se znaménkem minus (–) před nimi.
- ❏ Nuly vlevo od desetinné čárky (s výjimkou nuly těsně vedle desetinné čárky) se nezobrazují.
- ❏ Ve výstupním řetězci jsou hodnoty zarovnané vpravo.

Tabulka 6-19 Operandy platné pro instrukce, které konvertují číselné hodnoty na řetězce

Vstupy/Výstupy	Typ dat	Operandy
IN	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AIW, *VD, *LD, *AC, konstanta
	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, HC, *VD, *LD, *AC, konstanta
	REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC, konstanta
FMT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
OUT	ŘETĚZEC	VB, LB, *VD, *LD, *AC

FMT

MSB							LSB
7	6	5	4	3	2	1	0
0	0	0	0	c	n	n	n

c = desetinná čárka (1), nebo desetinná tečka (0)
nnn = počet číslic vpravo od desetinné čárky

	Out	Out +1	Out +2	Out +3	Out +4	Out +5	Out +6	Out +7	Out +8
in = 12	8				0	.	0	1	2
in = –123	8				0	.	1	2	3
in = 1234	8				1	.	2	3	4
in = –12345	8		–	1	2	.	3	4	5

Obr. 6-18 Operand FMT pro instrukci ”Integer na řetězec”

Činnost instrukce ”Double integer na řetězec”

Instrukce ”Double integer na řetězec” (DTS) konvertuje double integer (IN) na řetězec ASCII o délce 12 znaků. Formát (FMT) udává přesnost konverze vpravo od desetinné čárky a to, má–li se zobrazovat desetinná čárka nebo tečka. Výsledný řetězec je zapsán do 13 za sebou jdoucích bytů s počátkem v OUT. Více informací najdete v kapitole 4 v odstavci, který popisuje formát řetězců.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)
- Neplatný formát (nnn > 5)

Obrázek 6-19 ukazuje formát operandu v instrukci ”Double integer na řetězec”. Výstupní řetězec má vždy velikost 8 znaků. Počet číslic vpravo od desetinné čárky ve výstupním zásobníku je dán v poli nnn. Platný rozsah pole nnn je 0 až 5. Zadáním počtu číslic 0 bude hodnota zobrazena bez desetinné čárky. Pro hodnoty nnn vetší než 5 je výstupem retezec 12 ASCII mezer. V bitu c je dáno, zda má být pro oddělení celého čísla a desetinné části použita desetinná čárka (c=1), nebo desetinná tečka (c=0). Horní 4 bity formátu musí být nulové.

Obrázek 6-19 ukazuje příklady hodnot zformátovaných s desetinnou tečkou (c = 0) a čtyřmi číslicemi vpravo od desetinné tečky (nnn = 100). Hodnota v OUT je délka řetězce. Výstupní řetězec je formátován podle následujících pravidel:

- ❏ Kladné hodnoty se do výstupního zásobníku zapisují bez znaménka.
- ❏ Záporné hodnoty se do výstupního zásobníku zapisují se znaménkem minus (–) před nimi.
- ❏ Nuly vlevo od desetinné čárky (s výjimkou nuly těsně vedle desetinné čárky) se nezobrazují.
- ❏ Ve výstupním řetězci jsou hodnoty zarovnané vpravo.

FMT

MSB 7	6	5	4	3	2	1	LSB 0	
0	0	0	0	c	n	n	n	in = 12
								in = –1234567

c = desetinná čárka (1), nebo desetinná tečka (0)
nnn = počet číslic vpravo od desetinné čárky

Out	Out +1	Out +2	Out +3	Out +4	Out +5	Out +6	Out +7	Out +8	Out +9	Out +10	Out +11	Out +12
12					.	–	0	.	0	0	1	2
12					1	2	3	.	4	5	6	7

Obr. 6-19 Operand FMT pro instrukci ”Double Integer na řetězec”

Činnost instrukce ”Real na řetězec”

Instrukce ”Real na řetězec” (RTS) konvertuje reálnou hodnotu (IN) na řetězec ASCII. Formát (FMT) udává přesnost konverze napravo od desetinné čárky, délku výstupního řetězce a to, má–li se zobrazovat desetinná čárka, nebo tečka.

Výsledná konverze je zapsaná do řetězce začínajícího v OUT. Délka výsledného řetězce je specifikována veformátu a může být 3 až 15 znaků. Více informací najdete v kapitole 4 v odstavci, který popisuje formát řetězců.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)
- Neplatný formát:
  nnn > 5
  ssss < 3
  ssss < počet požadovaných znaků

Formát reálného čísla použitý v S7-200 podporuje maximálně 7 platných číslic. Pokus o zobrazení více než 7 platných číslic způsobí chybu zaokrouhlení.

Obrázek 6-20 ukazuje formát operandu v instrukci ”Real na řetězec”. Délka výstupního řetězce je dána polem ssss. Velikosti 0, 1 nebo 2 byty jsou neplatné. Počet číslic vpravo od desetinné čárky ve výstupním zásobníku je dán v poli nnn. Platný rozsah pole nnn je 0 až 5. Zadáním počtu číslic 0 bude hodnota zobrazena bez desetinné čárky. Pro hodnoty nnn větší než 5 nebo je–li zadaná délka výstupního řetězce krátká na vložení konvertované hodnoty je naplněn výstupní řetězec ASCII mezerami. V bitu c je dáno, zda má být pro oddělení celého čísla a desetinné části použita desetinná čárka (c=1), nebo desetinná tečka (c=0).

Obrázek 6-20 také ukazuje příklady hodnot zformátovaných s desetinnou tečkou (c = 0), jednou číslicí vpravo od desetinné tečky (nnn = 001) a délkou výstupního řetězce 6 znaků (ssss = 0110). Hodnota v OUT je délka řetězce. Výstupní řetězec je formátován podle následujících pravidel:

- ❏ Kladné hodnoty se do výstupního zásobníku zapisují bez znaménka.
- ❏ Záporné hodnoty se do výstupního zásobníku zapisují se znaménkem minus (–) před nimi.
- ❏ Nuly vlevo od desetinné čárky (s výjimkou nuly těsně vedle desetinné čárky) se nezobrazují.
- ❏ Hodnoty napravo od desetinné čárky jsou zaokrouhleny tak, aby měly daný počet číslic vpravo od desetinné čárky.
- ❏ Velikost výstupního řetězce musí být alespoň o tři byty větší, než je počet číslic vpravo od desetinné čárky.
- ❏ Ve výstupním řetězci jsou hodnoty zarovnané vpravo.

FMT

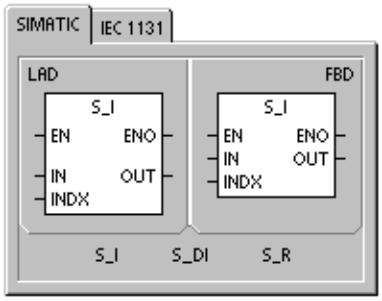
MSB							LSB
7	6	5	4	3	2	1	0
s	s	s	s	c	n	n	n

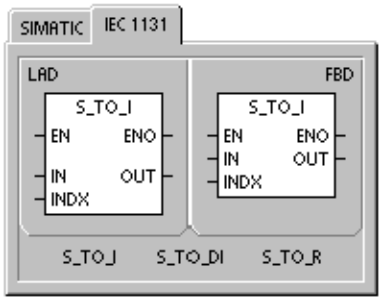
ssss = délka výstupního řetězce
c = desetinná čárka (1), nebo desetinná tečka (0)
nnn = počet číslic vpravo od desetinné čárky

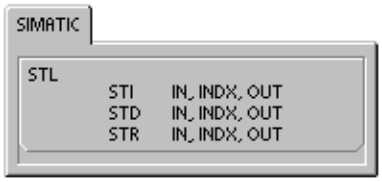
	Out	Out +1	Out +2	Out +3	Out +4	Out +5	Out +6
in = 1234.5	6	1	2	3	4	.	5
in = –0,0004	6				0		0
in = –3,67526	6			–	3	.	7
in = 1,95	6				2	.	0

Obr. 6-20 Operand FMT pro instrukci ”Real na řetězec”

Konverze řetězců na číselné hodnoty







Instrukce ”Řetězec na integer” (STI), ”Řetězec na double integer” (STD) a ”Řetězec na real” (STR) konvertují hodnotu řetězce IN začínající s offsetem INDX na hodnotu integer, double integer nebo reálné číslo v OUT.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)
- 009B (index = 0)
- SM1.1 (Přetečení)

Instrukce ”Řetězec na integer” a ”Řetězec na double integer” konvertují řetězce s následujícím tvarem:
[mezery] [+ nebo –] [číslice 0 – 9]

Instrukce ”Řetězec na real” konvertuje řetězce s následujícím tvarem:
[mezery] [+ nebo –] [číslice 0 – 9] [. nebo ,] [číslice 0 – 9]

Hodnota INDX je normálni nastavena na 1, což zahájí konverzi od prvního znaku řetězce. Je možné nastavit hodnotu INDX na jiné hodnoty pro zahájení konverze na různých místech uvnitř řetězce. Toho lze využít, jestliže vstupní řetězec obsahuje text, který není součástí konvertovaného čísla. Například pokud je vstupní řetězec ”Teplota: 77,8”, nastavíte INDX na hodnotu 13, aby se přeskočilo slovo ”Teplota:” na začátku řetězce.

Instrukce ”Řetězec na real” nekonvertuje řetězce, které používají zobrazení s pohyblivou řádovou čárkou nebo exponenciální tvary reálných čísel. Instrukce nedává chybu přeplnění (SM1.1), ale konvertuje řetězec na reálné číslo až k exponenciálnímu tvaru a pak konverzi ukončí. Například řetězec‘1.234E6’ je konvertován bez chyb na reálnou hodnotu 1,234.

Konverze se ukončí, když je dosaženo konce řetězce nebo když je zjištěn první neplatný znak. Neplatný je kterýkoliv znak, který není číslice (0 – 9).

Chyba přetečení (SM1.1) je ohlášena, kdykoliv je výsledkem konverze hodnota integer, která je příliš velká pro výstupní hodnotu. Například instrukce ”Řetězec na integer” nastaví chybu přetečení, pokud je výsledkem vstupního řetězce hodnota větší než 32767 nebo menší než –32768.

Chyba přetečení (SM1.1) je také nastavena, pokud není konverze možná, protože vstupní řetězec neobsahuje platnou hodnotu. Jestliže například vstupní řetězec obsahuje ’A123’, nastaví instrukce pro konverzi SM1.1 (přetečení) a výstupní hodnota zůstává nezměněna.

Tabulka 6-20 Operandy platné pro instrukce, které konvertují řetězce na číselné hodnoty

Vstupy/Výstupy	Typ dat	Operandy
IN	ŘETĚZEC	IB, QB, VB, MB, SMB, SB, LB, *VD, *LD, *AC, konstanta
INDX	BYTE	VB, IB, QB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
OUT	INT DINT, REAL	VW, IW, QW, MW, SMW, SW, T, C, LW, AC, AQW, *VD, *LD, *AC VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

Platné vstupní řetězce pro integer a double integer

Vstupní řetězec	Výstupní integer
‘123’	123
‘–00456’	–45
‘123.45’	6123
‘+2345’	2345
‘000000123ABCD’	123

Platné vstupní řetězce pro reálná čísla

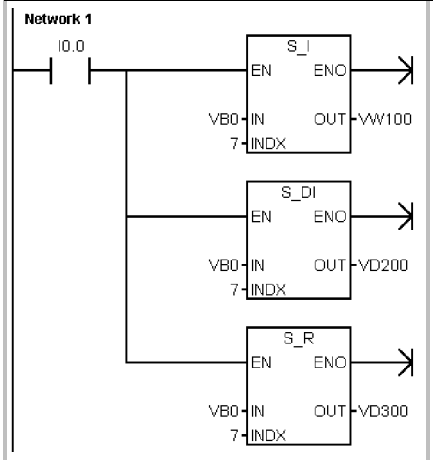
Vstupní řetězec	Výstupní reálné číslo
‘123’	123.0
‘–00456’	–456.0
‘123.45’	123.45
‘+2345’	2345.0
‘00.000000123’	0.000000123

Neplatné vstupní řetězce

Vstupní řetězec
‘A123’
‘ ’
‘++123’
‘+–123
‘+ 123’

Obr. 6-21 Příklady platných a neplatných vstupních řetězců

Příklad: Konverze řetězce: Subřetězec na integer, double integer nebo reálné číslo




```
Network 1       //Konvertuje číselný řetězec na integer.
                //Konvertuje číselný řetězec na double integer.
                //Konvertuje číselný řetězec na reálné číslo.
LD      I0.0
STI     VB0.7,VW100
STD     VB0.7,VD200
STR     VB0.7,VD300
```

VB0 … VB11

11	’T’	’e’	’m’	’p’	’	’	’9’	’8’	’.’	’6’	’F’
					,	,					

Po provedení networku:

VW100 (integer) = 98

VD200 (double integer) = 98

VD300 (reálné číslo) = 98,6

Zakódování a dekódování

Kódování

Instrukce ”Zakódovat” (ENCO) zapíše číslo bitu v nejnižším nastaveném bitu vstupního slova IN do nejnižšího ”půlbytu” (4 bity) výstupního bytu OUT.

Dekódování

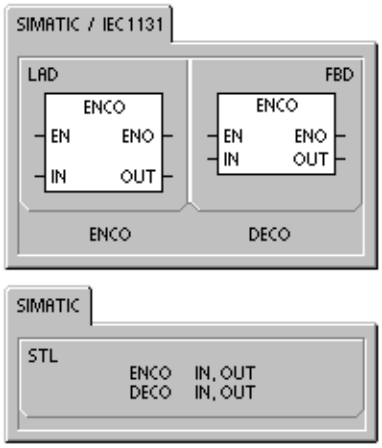
Instrukce ”Dekódovat” (DECO) nastaví bit výstupního slova OUT, který odpovídá číslu bitu danému v nejnižším ”půlbytu” (4 bity) vstupního bytu IN. Všechny ostatní bity výstupního slova jsou nastaveny na 0.

Bity SM a ENO

Jak v instrukci ”Zakódovat”, tak v instrukci ”Dekódovat” mají na ENO vliv následující podmínky.

Chyby, které nastavují ENO = 0

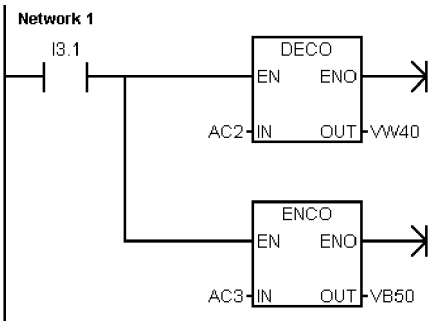
- 0006 (nepřímá adresa)



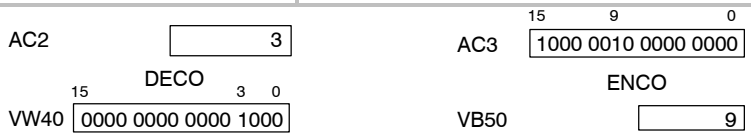
Tabulka 6-21 Operandy platné pro instrukce ”Zakódovat” a ”Dekódovat”

Vstupy/Výstupy	Typy dat	Operandy
IN	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC
	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AQW, *VD, *LD, *AC

Příklad: Instrukce ”Zakódovat” a ”Dekódovat”

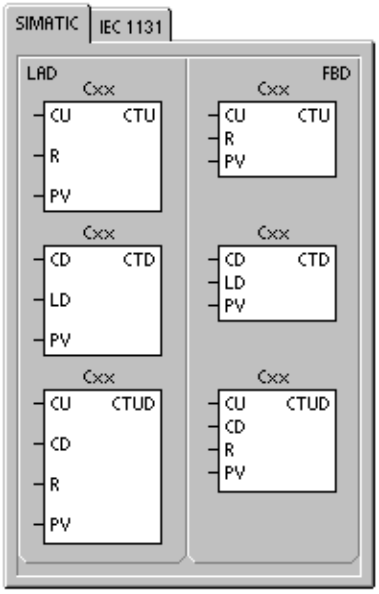


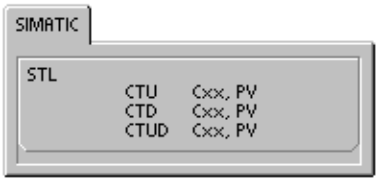

```
Network 1      //AC2 obsahuje chybové bity.
               //1. Instrukce DECO nastavuje
                    bit ve VW40
               //   který odpovídá tomuto chybovému
               //   kódu.
               //2. Instrukce ENCO konvertuje
               //   nejnižší bit nastavený na
               //   chybový kód,
               //   který je uložený ve VB50.
LD     I3.1
DECO   AC2, VW40
ENCO   AC3, VB50
```



Instrukce čítače

Instrukce čítače SIMATIC





Čítač nahoru

Instrukce "Čítač nahoru" (CTU) zvýší aktuální hodnotu o jedničku pokaždé, když vstup pro čítání (CU) přejde z vypnutého do zapnutého stavu. Když je aktuální hodnota Cxx větší nebo rovna nastavené hodnotě PV, nastaví se bit čítače Cxx. Čítač se vynuluje při zapnutí vstupu Reset (R) nebo při provedení instrukce "Reset". Čítač přestane čítat při dosažení maximální hodnoty (32 767).

V STL :

- Vstup "R": Vrchol zásobníku
- Vstup "CU": Hodnota uložená na druhém místě zásobníku

Čítač dolů

Instrukce "Čítač dolů" (CTD) sníží aktuální hodnotu čítače o jedničku pokaždé, když vstup čítání (CD) přejde z vypnutého do zapnutého stavu. Když je aktuální hodnota Cxx rovna 0, zapne se bit čítače Cxx. Čítač vynuluje bit Cxx a načte do aktuální hodnoty nastavenou hodnotu PV při zapnutí vstupu LD (Load). Čítač se zastaví při dosažení nuly a bit čítače Cxx se zapne.

V STL :

- Vstup "LD": Vrchol zásobníku
- Vstup "CD": Hodnota uložená na druhém místě zásobníku.

Čítač nahoru/dolů

Instrukce ”Čítač nahoru/dolů” (CTUD) zvýší aktuální hodnotu o jedničku, když vstup pro čítání (CU) přejde ze stavu log. 0 do log. 1, a sníží aktuální hodnotu o jedničku, když vstup pro čítání (CD) přejde ze stavu log. 0 do log. 1. Aktuální hodnota čítače Cxx uchovává aktuální počet. Nastavená hodnota PV je srovnávána s aktuální hodnotou pokaždé, když je provedena instrukce čítače.

Jestliže aktuální hodnota čítače dosáhne maxima (32 767), další náběžná hrana na vstupu nahoru způsobí nastavení aktuální hodnoty čítače na minimum (–32 768). Jestliže aktuální hodnota čítače dosáhne minima (–32 768), další náběžná hrana na vstupu dolů způsobí nastavení aktuální hodnoty čítače na maximum (32 767).

Když je aktuální hodnota Cxx větší nebo rovna nastavené hodnotě PV, nastaví se bit čítače Cxx. Jinak se bit čítače nastaví na log. 0. Čítač se vynuluje při zapnutí vstupu Reset (R) nebo při provedení instrukce ”Reset”. Při CTUD se čítač zastaví, když dosáhne PV.

V STL :

- Vstup ”R”: Vrchol zásobníku
- Vstup ”CD”: Hodnota uložená na druhém místě zásobníku
- Vstup ”CU”: Hodnota uložená na třetím místě zásobníku

Tabulka 6-22 Operandy platné pro SIMATIC instrukce čítače

Vstupy/Výstupy	Typy dat	Operandy
Cxx	WORD	Konstanta (C0 až C255)
CU, CD, LD, R	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
PV	INT	IW, QW, VW, MW, SMW, SW, LW, T, C, AC, AIW, *VD, *LD, *AC, konstanta



Tip

Protože má každý čítač jen jednu aktuální hodnotu, nepřiřazujte stejné číslo více než jednomu čítači. (Čítače nahoru, dolů, nahoru/dolů se stejným číslem přistupují na stejnou aktuální hodnotu.)

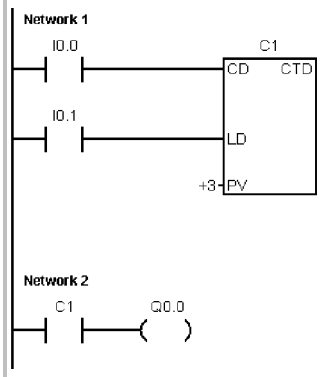
Při vynulování čítače pomocí instrukce ”Reset” se vynuluje bit čítače a aktuální hodnota čítače se nastaví na nulu. Číslo čítače použijte pro přístup k aktuální hodnotě i k bitu daného čítače.

Tabulka 6-23 Operace instrukcí čítače

Typ	Činnost	Bit čítače	Výpadek napájení nebo první cyklus
CTU	CU zvyšuje aktuální hodnotu. Aktuální hodnota se zvyšuje, dokud nedosáhne 32 767.	Bit čítače se nastaví na log. 1, když: Aktuální hodnota >= Nastavená	Bit čítače je nastaven na log. 0. Aktuální hodnotu je možné uchovat.[1]
CTUD	CU zvyšuje aktuální hodnotu. CD snižuje aktuální hodnotu. Aktuální hodnota se stále zvyšuje nebo snižuje, dokud není čítač vynulován.	Bit čítače se nastaví na log. 1, když: Aktuální hodnota >= Nastavená	Bit čítače je nastaven na log. 0. Aktuální hodnotu je možné uchovat.[1]
CTD	CD snižuje aktuální hodnotu, dokud aktuální hodnota nedosáhne 0.	Bit čítače se nastaví na log. 1, když: Aktuální hodnota = 0	Bit čítače je nastaven na log. 0. Aktuální hodnotu je možné uchovat.[1]

[1] Můžete nastavit, aby aktuální hodnota čítače byla remanentní. Více informací o remanentní paměti CPU S7-200 najdete v kapitole 4.

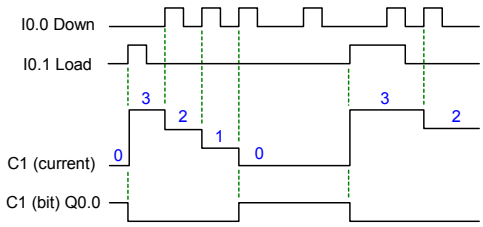
Příklad: Instrukce ”Čítač dolů”



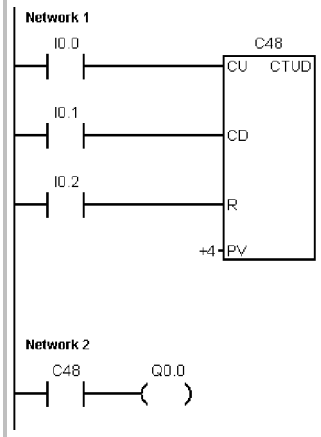

```
Network 1       //Aktuální hodnota čítače C1
                //čítá od 3 do 0
                //a I0.1 je nastaven na log. 0,
                //I0.0 ”Off–on” snižuje aktuální hodnotu C1
                //I0.1 ”On” obnoví nastavenou hodnotu odpočtu 3
LD      I0.0
LD      I0.1
CTD     C1, +3

Network 2       //Bit C1 je nastaven na log. 1, když se aktuální
hodnota čítače C1 = 0
LD      C1
=       Q0.0
```

Časový diagram



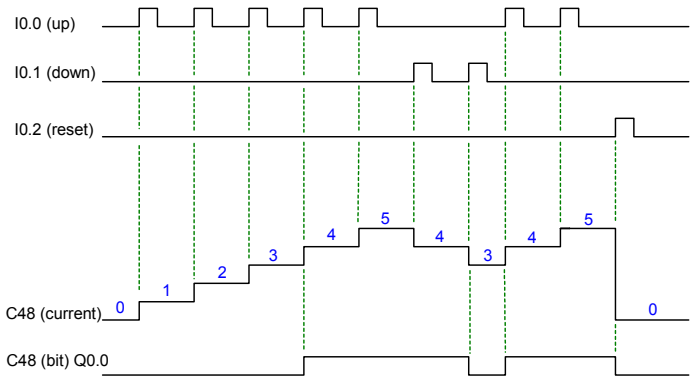
Příklad: Instrukce ”Čítač nahoru/dolů”




```
Network 1       //I0.0 čítá nahoru
                //I0.1 čítá dolů
                //I0.2 vynuluje aktuální hodnotu
LD      I0.0
LD      I0.1
LD      I0.2
CTUD    C48, +4

Network 2       //Čítač nahoru/dolů C48
                //zapne bit C48, když
                //aktuální hodnota >= 4
LD      C48
=       Q0.0
```

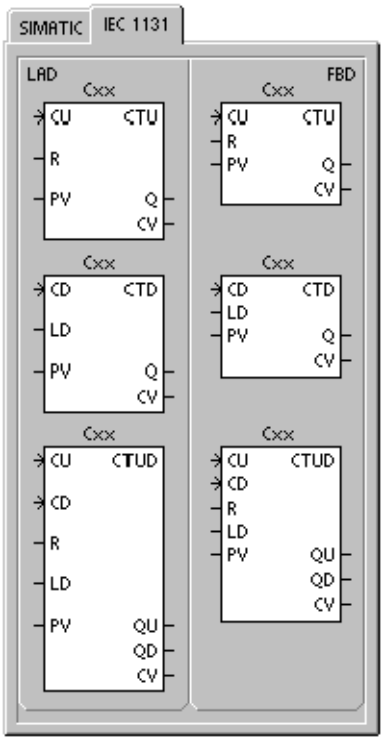
Časový diagram



Čítač IEC

Čítač nahoru

Pokud je na vstupu ”čítač nahoru” (CU) náběžná hrana, instrukce ”Čítač nahoru” (CTU) zvyšuje aktuální hodnotu o jedničku (až do nastavené hodnoty (PV) Je–li aktuální hodnota (CV) větší než nebo rovna nastavené hodnotě, nastaví se výstupní bit čítače (Q). Čítač se vynuluje, když je na resetovací vstup (R) přivedena log. 1. ”Čítač nahoru” přestane čítat při dosažení nastavené hodnoty.



Čítač dolů

Instrukce ”Čítač dolů” (CTD) snižují nastavenou hodnotu (PV) o jedničku při výskytu náběžné hrany na vstupu ”Čítač dolů” (CD). Když je aktuální hodnota (CV) rovna nule, nastaví se výstupní bit čítače (Q) na hodnotu log. 1. Když se povolí vstup (LD), čítač se vynuluje a obnoví aktuální hodnotu na nastavenou. Čítač dolů se zastaví, když dosáhne nuly.

Čítač nahoru/dolů

Instrukce ”Čítač nahoru/dolů” (CTUD) čítá nahoru nebo dolů od aktuální hodnoty (CV) při výskytu náběžných hran na vstupu ”čítač nahoru” (CU) nebo ”čítač dolů” (CD). Rovná–li se aktuální hodnota nastavené, výstup nahoru (QU) se nastaví na hodnotu log. 1. Rovná–li se aktuální hodnota nule, výstup dolů (QD) se nastaví na hodnotu log. 1. Čítač obnoví aktuální hodnotu na nastavenou (PV), pokud se na vstup (LD) přivede log. 1. Obdobně se čítač vynuluje a aktuální hodnota nastaví na 0, když se na vstup ”Reset” (R) přivede log. 1. Čítač přestane čítat, když dosáhne nastavené hodnoty nebo 0.

Tabulka 6-24 Operandy platné pro IEC instrukce čítače

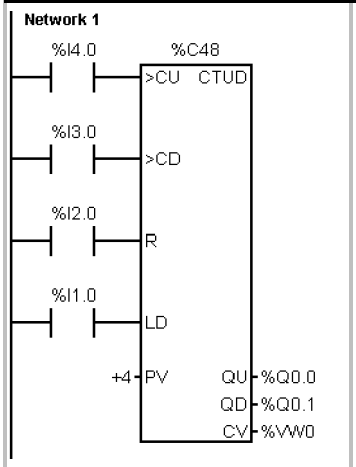
Vstupy/Výstupy	Typy dat	Operandy
Cxx	CTU, CTD, CTUD	Konstanta (C0 až C255)
CU, CD, LD, R	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
PV	INT	IW, QW, VW, MW, SMW, SW, LW, AC, AIW, *VD, *LD, *AC, konstanta
Q, QU, QD	BOOL	I, Q, V, M, SM, S, L
CV	INT	IW, QW, VW, MW, SW, LW, AC, *VD, *LD, *AC



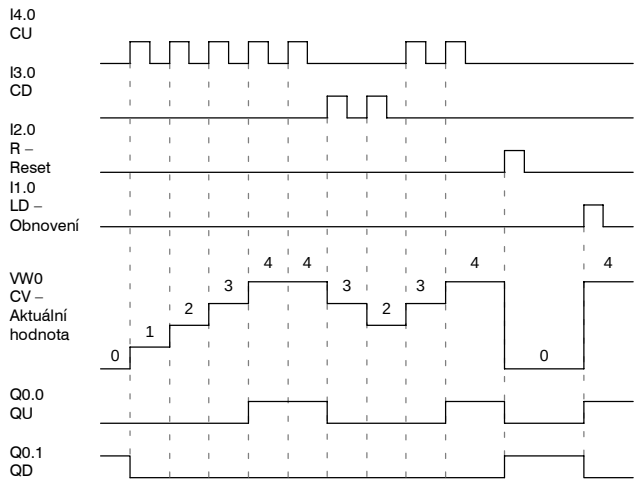
Tip

Protože má každý čítač jen jednu aktuální hodnotu, nepřiřazujte stejné číslo více než jednomu čítači. (Čítače nahoru, čítače dolů a čítače nahoru/dolů se stejným číslem přistupují na stejnou aktuální hodnotu.)

Příklad: Čítač IEC

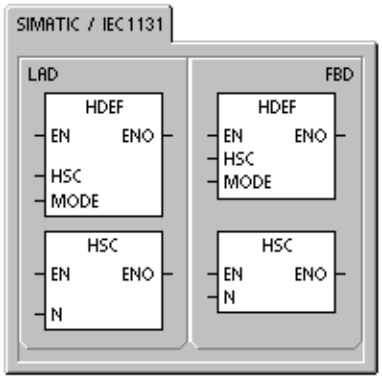


Časový diagram



Instrukce vysokorychlostního čítače

Definice vysokorychlostního čítače



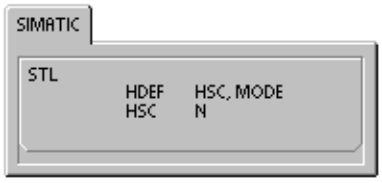
Instrukce "Definice vysokorychlostního čítače" (HDEF) přiřadí provozní režim příslušnému vysokorychlostnímu čítači (HSCx). Režim definuje pro vysokorychlostní čítač hodiny řízení směru, start a nulování.

Pro každý vysokorychlostní čítač použijte jednu instrukci "Definice vysokorychlostního čítače".

Chyby, které nastavují ENO = 0

- 0003 (konflikt vstupů)
- 0004 (nepovolená instrukce v přerušení)
- 000A (předefinování HSC)

Vysokorychlostní čítač



Instrukce "Vysoko rychlostní čítač" (HSC) konfiguruje a řídí vysokorychlostní čítač na základě stavu bitů speciální paměti HSC. Parametr N udává číslo čítače.

Vysokorychlostní čítače mohou být konfigurovány až pro dvanáct různých pracovních režimů. Viz tabulku 6-26.

Každý čítač má vstupy vyhrazené pro hodinové pulzy, řízení směru, nulování a start, pokud tyto funkce podporuje. Ve dvoufázových čítačích mohou oboje hodiny běžet na nejvyšší frekvenci. V kvadraturních režimech je možné zvolit jednonásobnou (1x) nebo čtyřnásobnou (4x) maximální rychlost čítače. Všechny čítače mohou běžet s maximální rychlostí bez vzájemného rušení.

Chyby, které nastavují ENO = 0

- 0001 (HSC před HDEF)
- 0005 (současné HSC/PLS)

Tabulka 6-25 Operandy platné pro instrukce vysokorychlostních čítačů

Vstupy/Výstupy	Typy dat	Operandy
HSC, MODE	BYTE	Konstanta
N	WORD	Konstanta


Tipy a triky

Na CD s dokumentací naleznete programovací tipy pro programy, které používají vysokorychlostní čítače. Viz tip 4 a tip 29.

Vysokorychlostní čítače čítají události, které se jinak nedají při dané rychlosti programového cyklu S7-200 zachytit. Maximální frekvence čítání vysokorychlostního čítače závisí na typu CPU S7-200. Další informace jsou v příloze A.



Tip

CPU 221 a CPU 222 podporují čtyři vysokorychlostní čítače: HSC0, HSC3, HSC4 a HSC5. Tyto CPU nepodporují HSC1 a HSC2.

CPU 224, CPU 224XP a CPU 226 podporují šest vysokorychlostních čítačů: HSC0 až HSC5.

Běžně se vysokorychlostní čítač používá jako zdroj pulzů pro krokový motor, kde je hřídel rotující konstantní rychlostí zapojena na inkrementální snímač. Kodér hřídele dává určité množství pulzů na otáčku a nulovací pulz, který vzniká jednou za otáčku. Hodiny a nulovací impulz z kodéru hřídele dávají vstupy pro vysokorychlostní počítadlo.

Dále se do vysokorychlostního čítače nahraje první z více požadovaných hodnot a aktivují se požadované výstupy na časový interval, dokud je aktuální počet menší než počet požadovaný. Čítač je nastaven tak, že vyvolá přerušení, když se aktuální počet rovná požadovanému a když dojde k vynulování.

Při každém výskytu události, která vyvolá přerušení, kdy ”aktuální hodnota je rovná požadované“, se nastaví nová požadovaná hodnota a pro výstup se nastaví další stav. Když se objeví událost přerušení od nulování, nastaví se první požadovaná hodnota a počáteční stavy výstupu a cyklus se opakuje.

Protože se přerušení objevují méně často, než je frekvence vysokorychlostních čítačů, je možné implementovat přesnou kontrolu vysokorychlostních operací s relativně malým vlivem na celkový programový cyklus PLC. Metoda přiřazení přerušení umožňuje, aby se každé nové nahrání požadované hodnoty uskutečnilo v samostatném podprogramu přerušení, čímž se ulehčuje stavová kontrola. (Alternativně je možné zpracovávat všechny události přerušení jedním podprogramem přerušení.)

Vysvětlení různých vysokorychlostních čítačů

Všechny čítače pracují stejným způsobem, když jsou ve stejném režimu. Existují čtyři základní typy čítačů: jednofázový čítač s vnitřním řízením směru, jednofázový čítač s vnějším řízením směru, dvoufázový čítač se 2 hodinovými vstupy a A/B fázový kvadraturní čítač. Všimněte si, že všechny režimy nejsou podporovány všemi čítači. Každý typ můžete používat: bez nulovacího a startovacího vstupu, s nulovacím vstupem a bez startovacího nebo se startovacím i nulovacím vstupem.

- Když aktivujete nulovací vstup, vymaže se aktuální hodnota čítače a zůstane tak až do deaktivace nulovacího vstupu.
- Když aktivujete startovací vstup, začne čítač počítat. Když je startovací vstup deaktivován, je aktuální hodnota čítače konstantní a hodinové pulzy jsou ignorovány.
- Při aktivaci nulovacího vstupu v době, kdy je startovací vstup neaktivní, je nulování ignorováno a aktuální hodnota se nemění. Při aktivaci startovacího vstupu a aktivním nulovacím vstupu je aktuální hodnota vymazána.

Před použitím vysokorychlostního čítače použijte pro nastavení jeho režimu instrukci HDEF (definice vysokorychlostního čítače). Použijte paměťový bit prvního programového cyklu, SM0.1 (tento bit je zapnutý jen při prvním programovém cyklu, potom se vypne), který vyvolá podprogram s instrukcí HDEF.

Programování vysokorychlostního čítače



Průvodce instrukcemi

Pro konfigurování čítače můžete použít průvodce pro instrukci HSC. Průvodce používá následující informace: typ a režim čítače, nastavená hodnota čítače, aktuální hodnota čítače a výchozí směr počítání. Chcete–li spustit průvodce pro HSC vyberte příkaz menu **Nástroje > Průvodce instrukcemi** a pak vyberte HSC z okna průvodce HSC.

Abyste naprogramovali vysokorychlostní čítač, musíte provést následující základní úlohy:

- Definování čítače a režimu.
- Nastavení řídicího bytu.
- Nastavení aktuální hodnoty (počáteční hodnota).
- Nastavení požadované hodnoty (cílová hodnota).
- Přiřazení a povolení přerušovacího programu.
- Aktivace vysokorychlostního čítače.

Definice režimů a vstupů čítače

Pro definování režimů a vstupů čítače použijte instrukci ”Definice vysokorychlostního čítače”.

Tabulka 6-26 ukazuje vstupy použité pro funkce hodin, řízení směru, nulování a spuštění pro vysokorychlostní čítače. Stejný vstup nemůže být použit pro dvě různé funkce, ale libovolný vstup, který není využit současným režimem vysokorychlostního čítače, může být použit pro jiný účel. Jestliže například HSC0 pracuje v režimu 1, který využívá I0.0 a I0.2, může být I0.1 použit pro přerušení hrany signálu nebo pro HSC3.



Tip

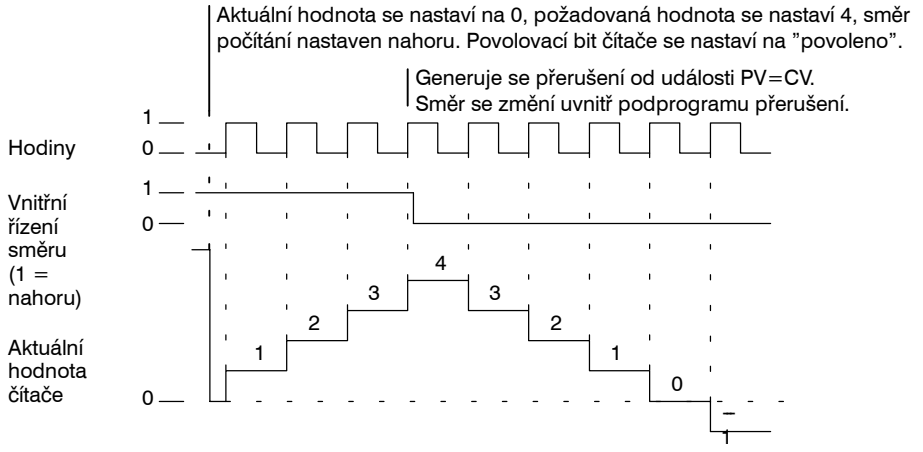
Všimněte si, že všechny režimy HSC0 (kromě režimu 12) vždy používají I0.0 a všechny režimy HSC4 vždy používají I0.3, takže při použití uvedených čítačů nejsou tyto vstupy nikdy k dispozici pro jiné použití.

Tabulka 6-26 Vstupy pro vysokorychlostní čítače

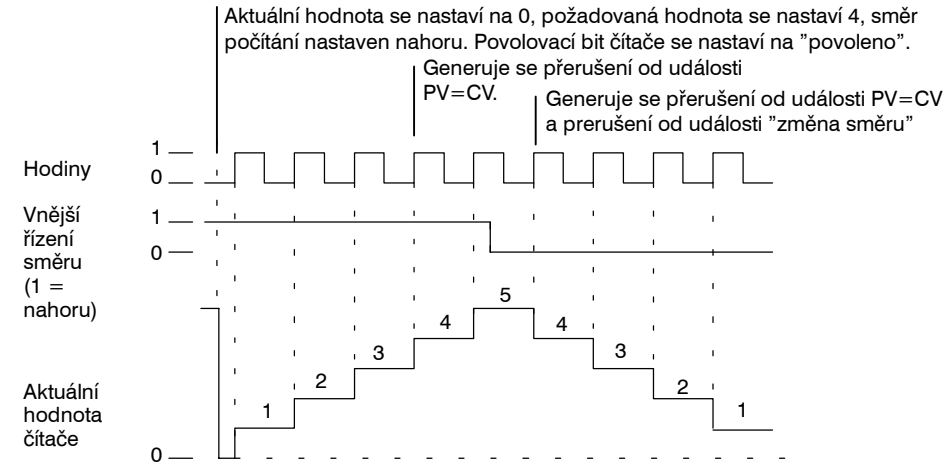
Režim	Popis	Vstupy			
	HSC0	I0.0	I0.1	I0.2	
	HSC1	I0.6	I0.7	I1.0	I1.1
	HSC2	I1.2	I1.3	I1.4	I1.5
	HSC3	I0.1			
	HSC4	I0.3	I0.4	I0.5	
	HSC5	I0.4			
0	Jednofázový čítač s vnitřním řízením směru	Hodiny			
1		Hodiny		Nulování	
2		Hodiny		Nulování	Start
3	Jednofázový čítač s vnějším řízením směru	Hodiny	Směr		
4		Hodiny	Směr	Nulování	
5		Hodiny	Směr	Nulování	Start
6	Dvojfázový čítač se 2 hodinovými vstupy	”Hodiny nahoru”	”Hodiny dolů”		
7		”Hodiny nahoru”	”Hodiny dolů”	Nulování	
8		”Hodiny nahoru”	”Hodiny dolů”	Nulování	Start
9	A/B fázový kvadraturní čítač	Hodiny A	Hodiny B		
10		Hodiny A	Hodiny B	Nulování	
11		Hodiny A	Hodiny B	Nulování	Start
12	Režim 12 podporují pouze HSC0 a HSC3. HSC0 počítá počet pulzů vycházejících z Q0.0. HSC3 počítá počet pulzů vycházejících z Q0.1.				

Příklady režimů HSC

Časové diagramy na obrázcích 6-22 až 6-26 ukazují funkci jednotlivých čítačů podle režimu.

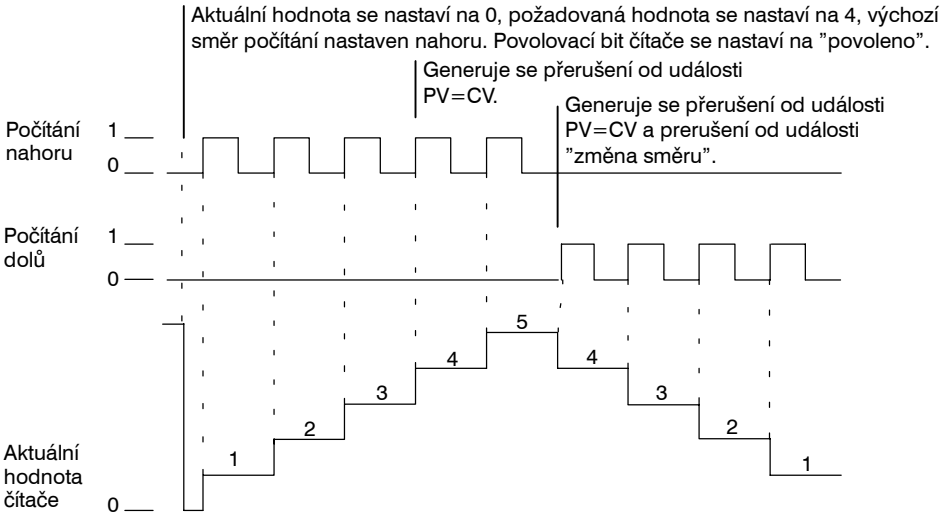


Obr. 6-22 Příklad funkce režimu 0, 1 nebo 2

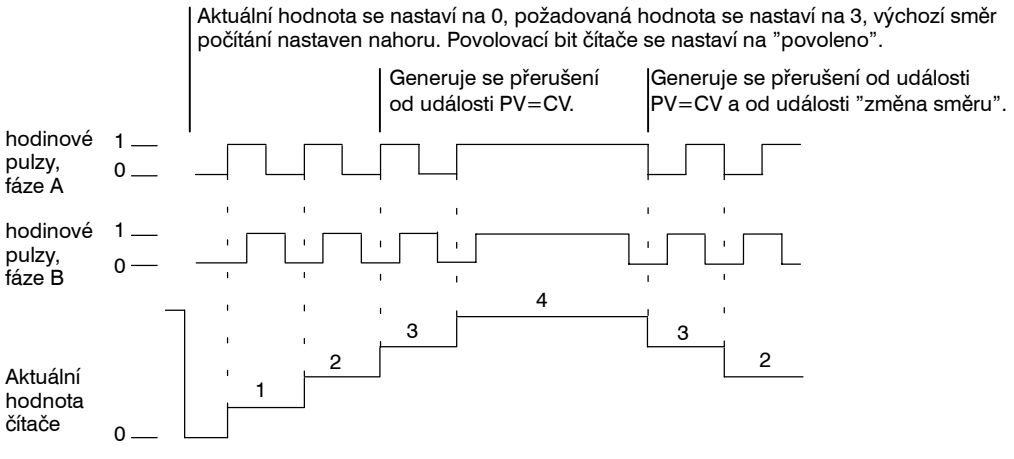


Obr. 6-23 Príklad funkce režimu 3, 4 nebo 5

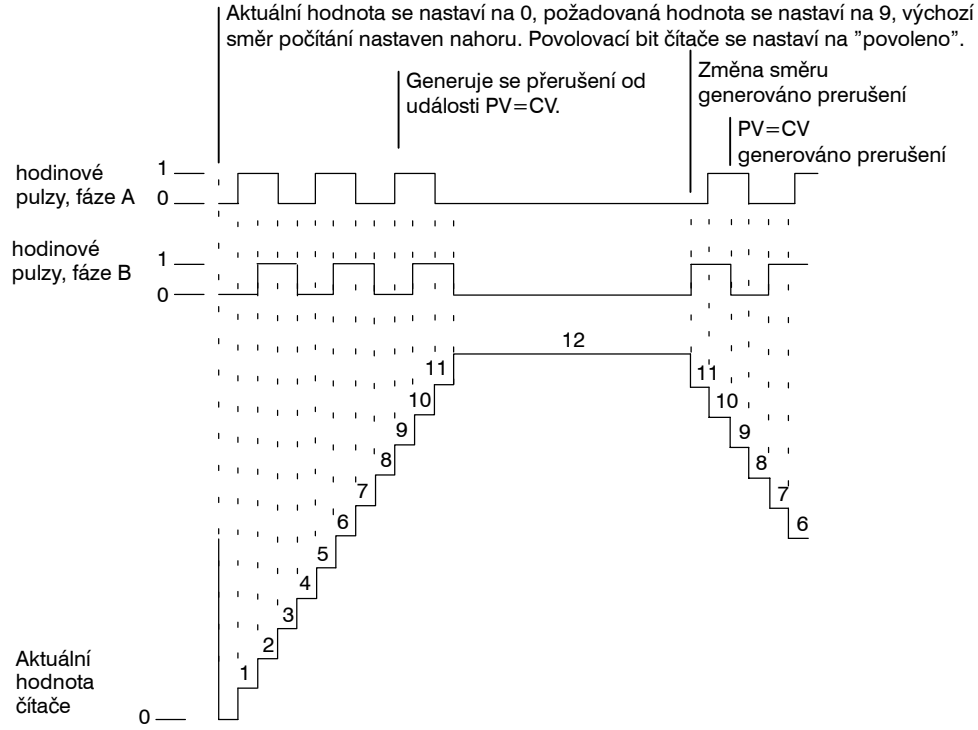
Pokud se v režimech 6, 7 nebo 8 vzestupné hrany na vstupech ”hodiny nahoru” a ”hodiny dolů” objeví v rozmezí 0,3 mikrosekundy, může vysokorychlostní čítač považovat tyto dvě události za současné. V takovém případě se aktuální hodnota nezmění a neindikuje se ani změna směru počítání. Pokud je časový odstup vzestupných hran vstupů hodin nahoru a dolů větší než tento časový úsek, zachytí vysokorychlostní čítač každou událost zvlášť. V obou případech se negeneruje chyba a čítač si udržuje správnou aktuální hodnotu.



Obr. 6-24 Příklad funkce režimu 6, 7 nebo 8



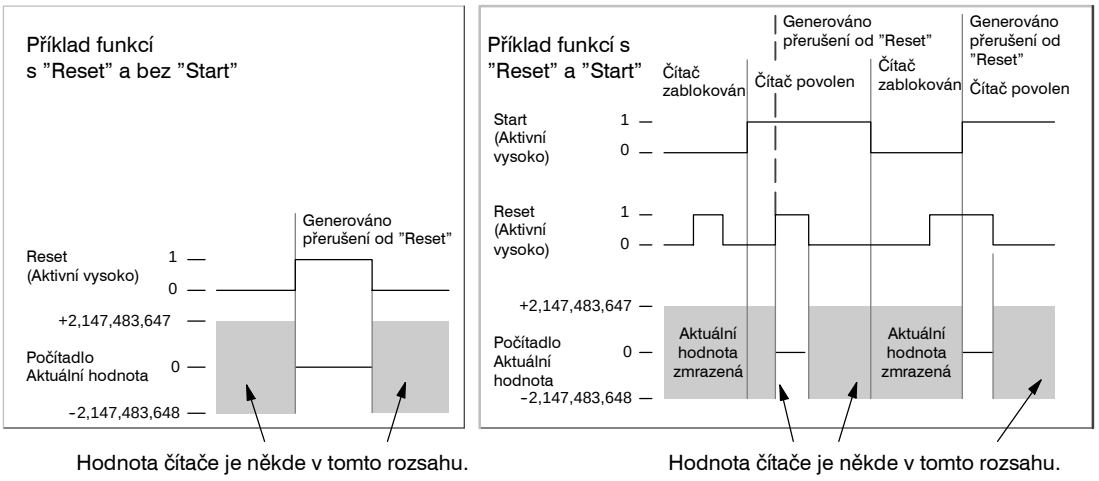
Obr. 6-25 Příklad funkce režimu 9, 10 nebo 11 (kvadraturní 1x režim)



Obr. 6-26 Příklad funkce režimu 9, 10 nebo 11 (kvadraturní 4x režim)

Vstupy "Reset" a "Start"

Funkce vstupů "Reset" a "Start" znázorněná na obrázku 6-27 platí pro všechny režimy, které používají tyto vstupy. V diagramech vstupů "Reset" a "Start" jsou nulování i start naprogramovány jako aktivním v úrovni log. 1.



Obr. 6-27 Příklady funkcí s "Reset" a se "Start" a bez něj

Čtyři čítače mají tři řídicí bity, které se používají pro konfiguraci aktivního stavu nulovacích a startovacích vstupů a pro zvolení 1x nebo 4x režimů (pouze kvadraturní čítače). Tyto bity jsou umístěny v řídicím bytu příslušného čítače a jsou použity pouze při provádění instrukce HDEF. Uvedené bity jsou definovány v tabulce 6-27.

Tip

Před provedením instrukce HDEF musíte nastavit tyto tři řídicí bity na požadované hodnoty. Jinak se čítač nastaví na implicitní konfiguraci daného režimu.

Po provedení instrukce HDEF se nastavení čítače nedá měnit, pokud nejprve nepřepnete automat S7-200 do režimu STOP.

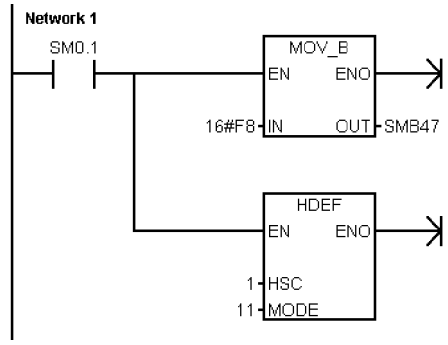
Tabulka 6-27 Aktivní úroveň pro vstupy "Reset", "Start" a 1x/4x řídicí bity

HSC0	HSC1	HSC2	HSC4	Popis (použito pouze po provedení HDEF)
SM37.0	SM47.0	SM57.0	SM147.0	Aktivní úroveň řídicího bitu pro "Reset"[1]: 0 = "Reset" je aktivní v úrovni log. 1 1 = "Reset" je aktivní v úrovni log. 0
– – –	SM47.1	SM57.1	– – –	Aktivní úroveň řídicího bitu pro "Start"[1]: 0 = "Start"je aktivní v úrovni log. 1 1 = "Start" je aktivní v úrovni log. 0
SM37.2	SM47.2	SM57.2	SM147.2	Výběr rychlosti čítání pro kvadraturní čítače: 0 = 4X rychlost počítání 1 = 1X rychlost počítání

1 Implicitní nastavení vstupů "Reset" a "Start" je aktivní stav v úrovni log. 1 a kvadraturní rychlost počítání 4x (neboli čtyřikrát vstupní frekvence hodin).

Příklad: Instrukce "Definice vysokorychlostního čítače"

MAIN




```
Network 1      //V prvním programovém cyklu:
               //1. Naprogramuje startovací a
               //   nulovací vstup na aktivní úroveň
               //   a zvolí 4x režim.
               //2. Konfiguruje HSC1 pro
               //   kvadraturní režimu s nulovacím
               //   a startovacím vstupem
LD     SM0.1
MOVB   16#F8, SMB47
HDEF   1, 11
```

Nastavení řídicího bytu

Po definici čítače a jeho režimu můžete naprogramovat dynamické parametry čítače. Každý vysokorychlostní čítač má řídicí byte, který umožňuje následující činnosti:

- ❏ Povolení nebo blokování čítače
- ❏ Řízení směru (pouze režimy 0, 1 a 2) nebo počáteční směr čítání pro všechny ostatní režimy
- ❏ Nahrání aktuální hodnoty
- ❏ Nahrání nastavené hodnoty

Provedením instrukce HSC je možné prohlédnout řídicí byte a příslušné aktuální a nastavené hodnoty. Tabulka 6-28 popisuje každý z těchto řídicích bitů.

Tabulka 6-28 Řídicí bity pro HSC0, HSC1, HSC2, HSC3, HSC4 a HSC5

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	Popis
SM37.3	SM47.3	SM57.3	SM137.3	SM147.3	SM157.3	Řídicí bit pro směr počítání: 0 = čítání dolů 1 = čítání nahoru
SM37.4	SM47.4	SM57.4	SM137.4	SM147.4	SM157.4	Zapsání směru počítání do HSC: 0 = Beze změny 1 = Aktualizuje směr
SM37.5	SM47.5	SM57.5	SM137.5	SM147.5	SM157.5	Zapsání nové požadované hodnoty do HSC: 0 = Beze změny 1 = Aktualizuje požadované hodnoty
SM37.6	SM47.6	SM57.6	SM137.6	SM147.6	SM157.6	Zapsání nové aktuální hodnoty do HSC: 0 = Beze změny 1 = Aktualizuje aktuální hodnotu
SM37.7	SM47.7	SM57.7	SM137.7	SM147.7	SM157.7	Povolení HSC: 0 = Zablokování HSC 1 = Povolení HSC

Nastavení aktuálních hodnot a požadovaných hodnot

Každý vysokorychlostní čítač má 32bitovou aktuální hodnotu a 32bitovou požadovanou hodnotu. Obě jsou integer se znaménkem. Pro nahrání nové aktuální nebo požadované hodnoty do vysokorychlostního čítače je třeba nastavit řídicí byte a byty speciální paměti, které obsahují dané hodnoty, a také provést instrukci HSC, aby se nové hodnoty přesunuly do vysokorychlostního čítače. V tabulce 6-29 jsou byty speciální paměti používané pro uchování aktuální a požadované hodnoty.

Kromě řídicích bytů a bytů obsahujících nové požadované a aktuální hodnoty je možné také přečíst aktuální hodnotu každého vysokorychlostního čítače pomocí datového typu HC, po kterém následuje číslo čítače (0, 1, 2, 3, 4 nebo 5), jak je ukázáno v tabulce 6-29. Aktuální hodnota je tedy přímo přístupná pro operace čtení, ale zapisovat se dá pouze instrukcí HSC.

Tabulka 6-29 Nové aktuální a nové požadované hodnoty HSC0, HSC1, HSC2, HSC3, HSC4 a HSC5

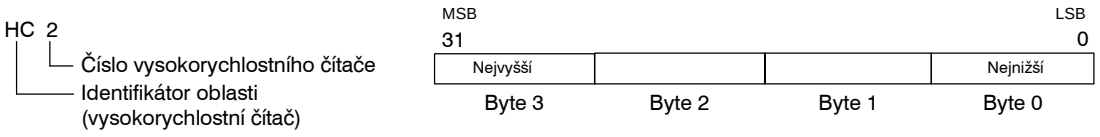
Nahrávaná hodnota	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5
Nová aktuální hodnota	SMD38	SMD48	SMD58	SMD138	SMD148	SMD158
Nová přednastavená hodnota	SMD42	SMD52	SMD62	SMD142	SMD152	SMD162

Tabulka 6-30 Aktuální hodnoty HSC0, HSC1, HSC2, HSC3, HSC4 a HSC5

Hodnota	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5
Aktuální hodnota	HC0	HC1	HC2	HC3	HC4	HC5

Adresování vysokorychlostních čítačů (HC)

Pro přístup k hodnotě vysokorychlostního čítače je třeba zadat adresu čítače zadáním typu paměti (HC) a čísla čítače (jako HC0). Aktuální hodnotu čítače je možné pouze číst a je možné k ní přistupovat pouze jako double word (32 bitů), jak ukazuje obrázek 6-28.



Obr. 6-28 Přístup k aktuálním hodnotám vysokorychlostních čítačů

Přiřazení přerušení

Všechny režimy čítače podporují přerušení, když se aktuální hodnota HSC rovná načtené přednastavené hodnotě. Režimy, které využívají externí nulovací vstup, podporují přerušení při aktivaci externího nulování. Všechny režimy čítače s výjimkou režimu 0, 1 a 2 podporují přerušení při změně směru čítání. Každá z těchto podmínek přerušení může být samostatně povolena nebo blokována. Použití přerušení je podrobně probráno v odstavci o přerušení a komunikaci.

Poznámka

Při použití přerušení od externího nulovacího vstupu se nepokoušejte nahrát novou aktuální hodnotu nebo blokovat a následně povolit vysokorychlostní čítač z přerušovacího podprogramu přiřazeného k dané události. Výsledkem může být výskyt kritické chyby.

Stavový byte

Stavový byte pro každý vysokorychlostní čítač má stavové paměťové bity indikující aktuální směr čítání a obsahující informaci o tom, zda je aktuální hodnota větší nebo rovna požadované hodnotě. Tabulka 6-31 definuje tyto stavové bity pro každý vysokorychlostní čítač.



Tip

Stavové bity jsou platné pouze v době provádění podprogramu přerušení pro vysokorychlostní čítač. Účelem monitorování stavu vysokorychlostního čítače je povolit přerušení pro události, které jsou pro právě prováděnou operaci důležité.

Tabulka 6-31 Stavové bity pro HSC0, HSC1, HSC2, HSC3, HSC4 a HSC5

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	Popis
SM36.0	SM46.0	SM56.0	SM136.0	SM146.0	SM156.0	Nevyužito
SM36.1	SM46.1	SM56.1	SM136.1	SM146.1	SM156.1	Nevyužito
SM36.2	SM46.2	SM56.2	SM136.2	SM146.2	SM156.2	Nevyužito
SM36.3	SM46.3	SM56.3	SM136.3	SM146.3	SM156.3	Nevyužito
SM36.4	SM46.4	SM56.4	SM136.4	SM146.4	SM156.4	Nevyužito
SM36.5	SM46.5	SM56.5	SM136.5	SM146.5	SM156.5	Stavový bit aktuálního směru čítání: 0 = čítání dolů 1 = počítání nahoru
SM36.6	SM46.6	SM56.6	SM136.6	SM146.6	SM156.6	Stavový bit rovnosti požadované a aktuální hodnoty: 0 = Nejsou si rovny 1 = Rovny
SM36.7	SM46.7	SM56.7	SM136.7	SM146.7	SM156.7	Stavový bit příznaku aktuální hodnota větší než požadovaná: 0 = Menší nebo rovna 1 = Větší než

Příklad inicializačních sekvencí pro vysokorychlostní čítače

V následujících popisech inicializačních a provozních sekvencí je jako modelový čítač použit HSC1. Při popisech inicializace se předpokládá, že byl do režimu RUN právě přepnut automat S7–200, a že tedy platím paměťový bit prvního programového cyklu. Pokud není tato podmínka splněna, je třeba si uvědomit, že instrukce HDEF může být pro každý vysokorychlostní čítač provedena po přepnutí do režimu RUN pouze jednou. Provedení instrukce HDEF pro vysokorychlostní čítač podruhé generuje chybu a nezmění nastavení čítače, které bylo zadáno v první HDEF.



Tip

Přestože následující sekvence ukazují, jak změnit směr, aktuální hodnotu a požadovanou hodnotu jednotlivě, je možné změnit všechny tyto prvky nebo jejich libovolnou kombinaci v jediné sekvenci příslušným nastavením hodnoty SMB47 a následným provedením instrukce HSC.

Inicializace režimů 0, 1 nebo 2

Následující kroky popisují inicializaci HSC1 pro jednofázový čítač nahoru/dolu s vnitřním ovládáním směru (režimy 0, 1 nebo 2).

1. Na vyvolání inicializačního podprogramu použijte paměťový bit prvního programového cyklu. Protože používáte volání podprogramu, následující programové cykly už podprogram nevolají, čímž se zkracuje čas provádění programového cyklu a zlepšuje se strukturovanost programu.
2. V inicializačním podprogramu naplňte SMB47 podle požadované řídicí operace. Například:

   SMB47 = 16#F8 *má za následek:*
   Povolí čítač
   Zapíše novou aktuální hodnotu
   Zapíše novou požadovanou hodnotu
   Nastaví směr čítání nahoru
   Nastaví startovací a nulovací vstup na aktivní v úrovni log. 1.
3. Proveďte instrukci HDEF se vstupem HSC nastaveným na 1 a vstupem MODE nastaveným na jedno z následujícího: 0 bez externího nulování nebo startu, 1 pro externí nulování a bez startu nebo 2 pro externí nulování i start.
4. Uložte do SMD48 (hodnota double word) požadovanou aktuální hodnotu (pro vymazání nahrajte 0).
5. Uložte do SMD52 (hodnota double word) požadovanou nastavenou hodnotu.
6. Abyste zachytili událost, kdy se aktuální hodnota rovná požadované, naprogramujte přerušení přidáním přerušovací události CV = PV (událost 13) do podprogramu přerušení. Úplné podrobné informace o zpracování přerušení najdete v části Instrukce přerušení.
7. Abyste zachytili externí nulovací událost, naprogramujte přerušení přidáním přerušovací události externí nulování (událost 15) do podprogramu přerušení.
8. Proveďte instrukci globálního povolení přerušení (ENI) pro povolení přerušení.
9. Proveďte instrukci HSC, aby automat S7-200 naprogramoval HSC1.
10. Ukončete podprogram.

Inicializace režimů 3, 4 nebo 5

Následující kroky popisují inicializaci HSC1 pro jednofázový čítač nahoru/dolu s externím ovládáním směru (režimy 3, 4 nebo 5).

1. Na vyvolání inicializačního podprogramu použijte paměťový bit prvního programového cyklu. Protože používáte volání podprogramu, následující programové cykly už podprogram nevolají, čímž se zkracuje čas provádění programového cyklu a zlepšuje se strukturovanost programu.
2. V inicializačním podprogramu naplňte SMB47 podle požadované řídicí operace. Například:

   SMB47 = 16#F8 *má za následek:*
   Povolí čítač
   Zapíše novou aktuální hodnotu
   Zapíše novou požadovanou hodnotu
   Nastaví počáteční směr čítání HSC nahoru
   Nastaví startovací a nulovací vstup na aktivní v úrovni log. 1.
3. Proveďte instrukci HDEF se vstupem HSC nastaveným na 1 a vstupem MODE nastaveným na jedno z následujícího: 3 bez externího nulování nebo startu, 4 pro externí nulování a bez startu nebo 5 pro externí nulování i start.
4. Uložte do SMD48 (hodnota double word) požadovanou aktuální hodnotu (pro vymazání nahrajte 0).
5. Uložte do SMD52 (hodnota double word) požadovanou nastavenou hodnotu.
6. Abyste zachytili událost, kdy se aktuální hodnota rovná požadované, naprogramujte přerušení přidáním prerušovací události CV = PV (událost 13) do podprogramu prerušení. Úplné podrobné informace o zpracování přerušení najdete v části Instrukce přerušení.
7. Abyste zachytili změny směru, naprogramujte přerušení přidáním přerušovací události změna směru (událost 14) do podprogramu přerušení.
8. Abyste zachytili externí nulovací událost, naprogramujte přerušení přidáním přerušovací události externí nulování (událost 15) do podprogramu přerušení.
9. Proveďte instrukci globálního povolení přerušení (ENI) pro povolení přerušení.
10. Proveďte instrukci HSC, aby automat S7-200 naprogramoval HSC1.
11. Ukončete podprogram.

Inicializace režimů 6, 7 nebo 8

Následující kroky popisují inicializaci HSC1 pro dvoufázový čítač nahoru/dolů se vstupy pro hodinové pulzy ve směru nahoru/dolů (režimy 6, 7 nebo 8):

1. Na vyvolání inicializačního podprogramu použijte paměťový bit prvního programového cyklu. Protože používáte volání podprogramu, následující programové cykly už podprogram nevolají, čímž se zkracuje čas provádění programového cyklu a zlepšuje se strukturovanost programu.
2. V inicializačním podprogramu naplňte SMB47 podle požadované řídicí operace. Například:

   SMB47 = 16#F8 *má za následek:*
   Povolí čítač
   Zapíše novou aktuální hodnotu
   Zapíše novou požadovanou hodnotu
   Nastaví počáteční směr čítání HSC nahoru
   Nastaví startovací a nulovací vstup na aktivní v úrovni log. 1.
3. Proveďte instrukci HDEF se vstupem HSC nastaveným na 1 a vstupem MODE nastaveným na jedno z následujícího: 6 bez externího nulování nebo startu, 7 pro externí nulování a bez startu nebo 8 pro externí nulování i start.
4. Uložte do SMD48 (hodnota double word) požadovanou aktuální hodnotu (pro vymazání nahrajte 0).
5. Uložte do SMD52 (hodnota double word) požadovanou nastavenou hodnotu.

6. Abyste zachytili událost, kdy se aktuální hodnota rovná požadované, naprogramujte přerušení přidáním prerušovací události CV = PV (událost 13) do podprogramu prerušení. Viz část o přerušeních.
7. Abyste zachytili změny směru, naprogramujte přerušení přidáním přerušovací události změna směru (událost 14) do podprogramu přerušení.
8. Abyste zachytili externí nulovací událost, naprogramujte přerušení přidáním přerušovací události externí nulování (událost 15) do podprogramu přerušení.
9. Proveďte instrukci globálního povolení přerušení (ENI) pro povolení přerušení.
10. Proveďte instrukci HSC, aby automat S7-200 naprogramoval HSC1.
11. Ukončete podprogram.

Inicializace režimů 9, 10 nebo 11

Následující kroky popisují inicializaci HSC1 pro A/B fázový kvadraturní čítač (pro režimy 9, 10 nebo 11):

1. Na vyvolání inicializačního podprogramu použijte paměťový bit prvního programového cyklu. Protože používáte volání podprogramu, následující programové cykly už podprogram nevolají, čímž se zkracuje čas provádění programového cyklu a zlepšuje se strukturovanost programu.
2. V inicializačním podprogramu naplňte SMB47 podle požadované řídicí operace.

   Například (při režimu 1x):
   SMB47 = 16#FC *má za následek:*
   Povolí čítač
   Zapíše novou aktuální hodnotu
   Zapíše novou požadovanou hodnotu
   Nastaví počáteční směr čítání HSC nahoru
   Nastaví startovací a nulovací vstup na aktivní v úrovni log. 1.

   Například (při režimu 4x):
   SMB47 = 16#F8 *má za následek:*
   Povolí čítač
   Zapíše novou aktuální hodnotu
   Zapíše novou požadovanou hodnotu
   Nastaví počáteční směr čítání HSC nahoru
   Nastaví startovací a nulovací vstup na aktivní v úrovni log. 1.
3. Proveďte instrukci HDEF se vstupem HSC nastaveným na 1 a vstupem MODE nastaveným na jedno z následujícího: 9 bez externího nulování nebo startu, 10 pro externí nulování a bez startu nebo 11 pro externí nulování i start.
4. Uložte do SMD48 (hodnota double word) požadovanou aktuální hodnotu (pro vymazání nahrajte 0).
5. Uložte do SMD52 (hodnota double word) požadovanou nastavenou hodnotu.
6. Abyste zachytili událost, kdy se aktuální hodnota rovná požadované, naprogramujte přerušení přidáním prerušovací události CV = PV (událost 13) do podprogramu prerušení. Úplné podrobné informace o zpracování přerušení najdete v části o povolování přerušení (ENI).
7. Abyste zachytili změny směru, naprogramujte přerušení přidáním přerušovací události změna směru (událost 14) do podprogramu přerušení.
8. Abyste zachytili externí nulovací událost, naprogramujte přerušení přidáním přerušovací události externí nulování (událost 15) do podprogramu přerušení.
9. Proveďte instrukci globálního povolení přerušení (ENI) pro povolení přerušení.
10. Proveďte instrukci HSC, aby automat S7-200 naprogramoval HSC1.
11. Ukončete podprogram.

Inicializační režim 12

Následující kroky popisují, jak inicializovat HSC0 pro počítání pulzů generovaných PTO0 (Režim 12).

1. Na vyvolání inicializačního podprogramu použijte paměťový bit prvního programového cyklu. Protože používáte volání podprogramu, následující programové cykly už podprogram nevolají, čímž se zkracuje čas provádění programového cyklu a zlepšuje se strukturovanost programu.
2. V inicializačním podprogramu naplňte SMB37 podle požadované řídicí operace. Například:

SMB37 = 16#F8	*má za následek:* Povolí čítač Zapíše novou aktuální hodnotu Zapíše novou požadovanou hodnotu Nastaví směr čítání nahoru Nastaví startovací a nulovací vstup na aktivní v úrovni log. 1.

3. Proveďte instrukci HDEF se vstupem HSC nastaveným na 0 a vstupem MODE nastaveným na 12:
4. Uložte do SMD38 (hodnota double word) požadovanou aktuální hodnotu (pro vymazání nahrajte 0).
5. Uložte do SMD42 (hodnota double word) požadovanou nastavenou hodnotu.
6. Abyste zachytili událost, kdy se aktuální hodnota rovná požadované, naprogramujte přerušení přidáním přerušovací události CV = PV (událost 13) do podprogramu přerušení. Úplné podrobné informace o zpracování přerušení najdete v části Instrukce přerušení.
7. Proveďte instrukci globálního povolení přerušení (ENI) pro povolení přerušení.
8. Proveďte instrukci HSC, aby automat S7-200 naprogramoval HSC0.
9. Ukončete podprogram.

Změna směru v režimech 0, 1, 2 nebo 12

Následující kroky popisují konfiguraci HSC1 pro změnu směru pro jednofázový čítač s vnitřním ovládáním směru (režimy 0, 1, 2 nebo 12):

1. Pro zapisování v požadovaném smiru uložte do SMB47:

SMB47 = 16#90	Povolí čítač Nastaví směr čítání HSC dolů
SMB47 = 16#98	Povolí čítač Nastaví směr čítání HSC nahoru

2. Proveďte instrukci HSC, aby automat S7-200 naprogramoval HSC1.

Zadání nové aktuální hodnoty (libovolný režim)

Během změny aktuální hodnoty je čítač blokován. Během blokování čítače se negenerují přerušení a nepokračuje čítání.

Následující kroky popisují změnu aktuální hodnoty čítače HSC1 (platí pro všechny režimy):

1. Pro zapisování požadované aktuální hodnoty uložte do SMB47:

SMB47 = 16#C0	Povolí čítač Zapíše novou aktuální hodnotu

2. Uložte do SMD48 (hodnota double word) požadovanou aktuální hodnotu (pro vymazání nahrajte 0).
3. Proveďte instrukci HSC, aby automat S7-200 naprogramoval HSC1.

Zadání nové požadované hodnoty (libovolný režim)

Následující kroky popisují změnu požadované hodnoty čítače HSC1 (platí pro všechny režimy):

1. Pro zapisování nové požadované hodnoty uložte do SMB47 :

   SMB47 = 16#A0 Povolí čítač
   Zapíše novou požadovanou hodnotu

2. Uložte do SMD52 (hodnota double word) požadovanou nastavenou hodnotu.
3. Proveďte instrukci HSC, aby automat S7-200 naprogramoval HSC1.

Pozastavení vysokorychlostního čítače (libovolný režim)

Následující kroky popisují způsob, jak pozastavit vysokorychlostní čítač HSC1 (platí pro všechny režimy):

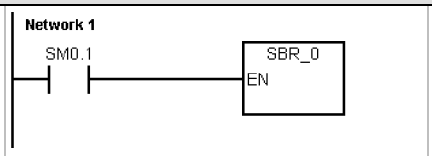
1. Uložte do SMB47 pro pozastavení čítače:

   SMB47 = 16#00 Pozastaví čítač

2. Proveďte instrukci HSC, abyste čítač pozastavili.

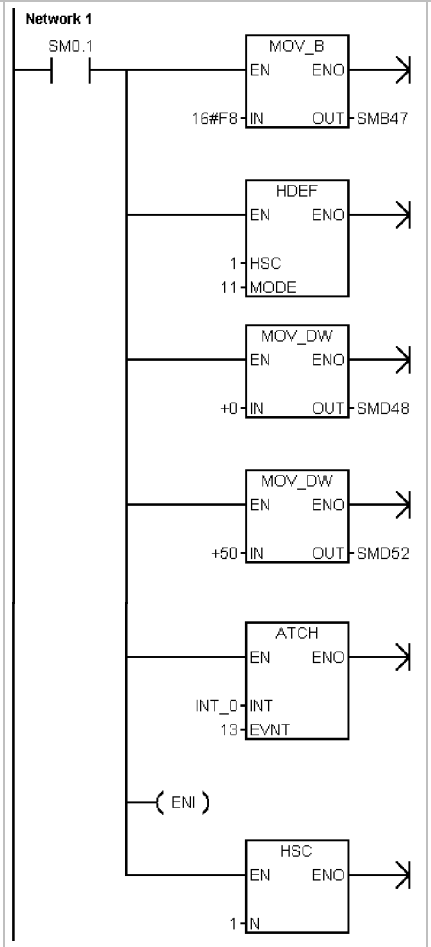
Příklad: Instrukce vysokorychlostního čítače

MAIN




```
Network 1       //V prvním programovém cyklu volejte SBR_0.
LD      SM0.1
CALL    SBR_0
```

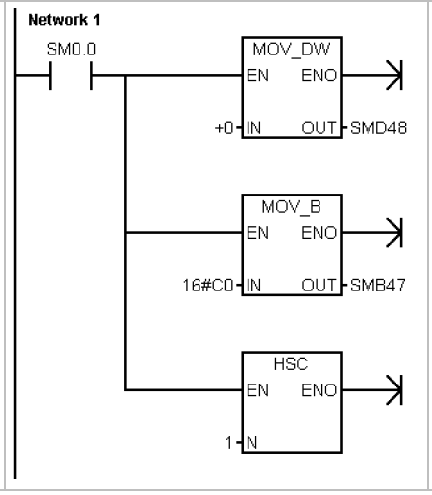
SBR 0




```
Network 1       //V prvním programovém cyklu konfigurujte HSC1:
                //1. Povolí čítač.
                //   – Zapíše novou aktuální hodnotu.
                //   – Zapíše novou požadovanou hodnotu.
                //   – Nastaví počáteční směr čítání nahoru.
                //   – Naprogramuje startovací a nulovací vstupy
                //     na aktivní úroveň.
                //   – Nastaví 4x režim.
                //2. Konfiguruje HSC1 pro kvadraturní režim
                //   se vstupy "Reset" a "Start".
                //3. Vymaže aktuální hodnotu HSC1.
                //4. Nastaví požadovanou hodnotu HSC1 na 50.
                //5. Pokud aktuální hodnota HSC1 = požadované
                //hodnotě, přiřadí událost 13 k přerušovacímu
                //podprogramu INT_0.
                //6. Globální povolení přerušení.
                //7. Naprogramuje HSC1.

LD      SM0.1
MOVB    16#F8, SMB47
HDEF    1, 11
MOVD    +0, SMD48
MOVD    +50, SMD52
ATCH    INT_0, 13
ENI
HSC     1
```

INT 0




```
Network 1       //Naprogramuje HSC1:
                //1. Vymaže aktuální hodnotu HSC1.
                //2. Vybere zapisování pouze nové aktuální hodnoty
                //   a ponechejte HSC1 povolený.

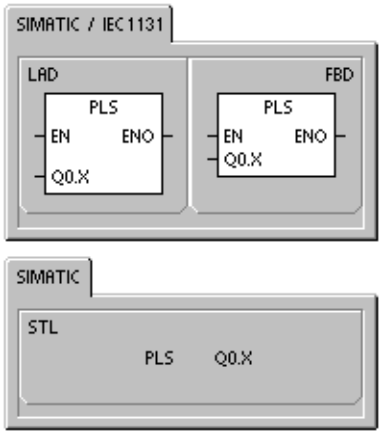
LD      SM0.0
MOVD    +0, SMD48
MOVB    16#C0, SMB47
HSC     1
```

Instrukce pulzního výstupu



Řízení polohy

Instrukce ”Pulzní výstup” (PLS) se používá pro řízení funkce pulzních výstupů (PTO) a pulzní šířkové modulace (PWM) dostupných na vysokorychlostních výstupech (Q0.0 a Q0.1).



Vylepšený ”Průvodce polohováním” vytváří instrukce upravené pro vaši aplikaci, která zjednodušuje programovací úlohy a využívá dalších funkcí CPU S7-200. Více informací o “Průvodci polohováním” najdete v kapitole 9.

Můžete dále používat starou instrukci PLS k vytvoření vlastní pohybové aplikace, ale lineární náběh na PTO podporují pouze instrukce vytvořené vylepšeným průvodcem polohováním.

PTO generuje obdélníkový signál (50% střída) s možností řízení periody a počtu pulzů.

PWM generuje kontinuální výstup s proměnnou střídou s možností řízení periody a počtu pulzů.

Programovatelný automat S7-200 má dva PTO/PWM generátory pro generování sledu vysokorychlostních pulzů a šířkově modulovaných signálů. Jeden generátor má přiřazen digitální výstup Q0.0 a druhý generátor Q0.1. Určená oblast speciální paměti (SM) uchovává pro každý generátor následující data: řídicí byte (8bitová hodnota), hodnotu počtu pulzů (32bitová hodnota bez znaménka) a hodnotu periody a šířky pulzu (16bitová hodnota bez znaménka).

Generátory PTO/PWM a registr obrazu procesu se dělí o použití Q0.0 a Q0.1. Když je funkce PTO nebo PWM aktivní na Q0.0 nebo Q0.1, je výstup řízen generátorem PTO/PWM a normální použití výstupního není povoleno. Tvar výstupní signálu není ovlivněn stavem registru obrazu procesu, vnucenou (forced) hodnotou výstupu nebo provedením instrukcí přímého výstupu. Když je generátor PTO/PWM neaktivní, řízení výstupu přebírá zpět registr obrazu procesu. Registr obrazu procesu určuje stav výstupního signálu a řídí, zda je signál v log. 1 nebo log. 0.

Tabulka 6-32 Operandy platné pro instrukci pulzního výstupu

Vstupy/Výstupy	Typy dat	Operandy
Q0.X	WORD	Konstanta: 0 (= Q0.0) *nebo* 1 (= Q0.1)



Tip

Před povolením PTO nebo PWM nastavte hodnotu registru obrazu procesu pro Q0.0 a Q0.1 na 0.

Implicitní hodnoty pro všechny řídicí bity, čas cyklu, šířku pulzu a počet pulzů jsou 0.

Výstupy PTO/PWM musí být zatíženy minimálně 10 % nominální zátěže, aby generovaly dostatečně ostré přechody z vypnutého do zapnutého stavu a ze zapnutého do vypnutého stavu.



Tipy a triky

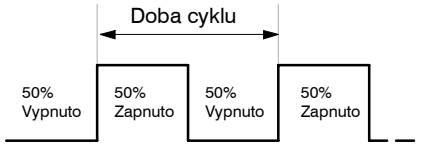
Programovací tipy pro programy, které používají instrukci PLS pro operaci PTO/PWM, najdete na CD s dokumentací. Viz tip 7, tip 22, tip 23, tip 30 a tip 50.

Činnost pulzního výstupu (PTO)

PTO generuje obdélníkový signál (střída 50 %) s daným počtem pulzů a danou periodou (viz obrázek NO TAG.) PTO může generovat jednu nebo více serií pulzů (s použitím pulzu profilu). Specifikuje se počet pulzů a perioda (v mikrosekundových nebo milisekundových přírůstcích):

- Počet pulzů: 1 až 4 294 967 295
- Perioda: 10µs až 65.535µs
  nebo
  2 ms až 65.535 ms.

Pokud má perioda formu lichého čísla (např. 75 ms) způsobuje to určitou deformaci signalu (duty cycle).



Obr. 6-29 Pulzní výstup (PTO)

Omezení počtu pulzů a periody najdete v tabulce 6-33.

Tabulka 6-33 Počet pulzů a periody ve funkci PTO

Počet pulzů, perioda	Odezva
Perioda < 2 časové jednotky	Perioda se implicitně nastaví na 2 časové jednotky.
Počet pulzů = 0	Počet pulzů je implicitně nastaven na 1 pulz.

Funkce PTO umožňuje ”zřetězení” nebo ”zřetězené zpracování” sekvence pulzů. Když se ukončí aktivní sekvence pulzů, bezprostředně začne výstup z nové sekvence pulzů. Tak se udržuje spojitost mezi za sebou jdoucími výstupními sekvencemi pulzů.

Použití Průvodce polohováním

Průvodce polohováním automaticky řídí řetězení samostatných a vícenásobných segmentů pulzů PTO, šířkovou modulaci pulzů, konfiguraci míst v paměti SM a vytváření tabulky profilu. Informace zde uvedené jsou pro vaši referenci. Doporučuje se, abyste používali Průvodce polohováním. Podrobné informace o Průvodci polohováním jsou uvedeny v kapitole 9.

Řetězení samostatných segmentů pulzů PTO

Při řetězení samostatných segmentů odpovídá programátor za aktualizaci SM oblasti pro následující sekvenci. Když byl spuštěn první PTO segment, musíte bezprostředně změnit SM oblasti tak, aby odpovídaly průběhu následujícího signálu, a opětovně provést instrukci PLS. Atributy druhé sekvence se drží v ”řetězu”, dokud se neukončí první sekvence pulzů. V ”řetězu” může být v dané době pouze jedna položka. Když se ukončí první sekvence, začne výstup druhé vlny a ”řetěz” je připraven pro zadání atributů nové sekvence. Tento proces můžete zopakovat pro nastavení atributů další sekvence pulzů.

Přechod mezi sekvencemi pulzů je hladký, pokud nedojde ke změně časové základny nebo pokud se neukončí aktivní sekvence pulzů před zadáním atributů nové sekvence pomocí provedení instrukce PLS.

Řetězení více segmentů pulzů PTO

Při řetězení více segmentů S7-200 automaticky čte atributy každé sekvence pulzů z takzvané tabulky profilu, která se nachází v paměti V. Z paměti SM se v tomto režimu pro tabulku profilu používá řídicí byte, stavový byte a počáteční offset paměti V (SMW168 nebo SMW178). Časová základna může být dána v mikrosekundách nebo milisekundách, ale je stejná pro všechny časy cyklů v tabulce profilu a v době běhu profilu se nedá měnit. Řetězení s více segmenty se spouští provedením instrukce PLS.

Každá položka pro segment má délku 8 bytů a skládá se z 16bitové hodnoty času cyklu, 16bitové hodnoty změny periody a 32bitové hodnoty počtu pulzů. Tabulka 6-34 ukazuje formát tabulky profilu. Periodu můžete automaticky zvýšit nebo snížit naprogramováním příslušné hodnoty pro každý pulz. Kladná hodnota pro změnu periody prodlužuje periodu, záporná hodnota ji zkracuje; zadáním 0 se čas cyklu nemění.

V době běhu profilu PTO se počet aktuálni aktivních segmentu dá zjistit v SMB166 (nebo SMB176).

Tabulka 6-34 Formát tabulky profilu pro režim PTO s více segmenty

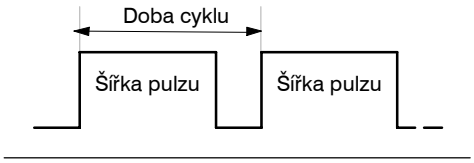
Posunutí bytu	Segment	Popis položek tabulky
0		Počet segmentů: 1až 255[1]
1	#1	Počáteční perioda (2 až 65.535 jednotek časové základny)
3		změna periody na jeden pulz (hodnota se znaménkem) (–32.768 až 32.767 jednotek časové základny)
5		Počet pulzů (1 až 4.294.967.295)
9	#2	Počáteční perioda (2 až 65.535 jednotek časové základny)
11		změna periody na jeden pulz (hodnota se znaménkem) (–32.768 až 32.767 jednotek časové základny)
13		Počet pulzů (1 až 4.294.967.295)
(Pokračování)	#3	(Pokračování)

1 Vložení hodnoty 0 jako počtu segmentů generuje nekritickou chybu. Negeneruje se výstup PTO.

Pulzní šířková modulace (PWM)

Funkce PWM dává výstup s pevnou periodou a s proměnou délkou pulzu. (Viz obrázek 6-30.) Periodu a šířku pulzu můžete zadat v mikrosekundových nebo milisekundových přírůstcích:

- Perioda: 10µs až 65.535µs nebo 2 ms až 65.535 ms
- Šířka pulzu: 0µs až 65.535µs nebo 0 ms až 65.535 ms



Obr. 6-30 Pulzní šířková modulace (PWM)

Jak je patrné z tabulky 6-35, při nastavení šířky pulzu tak, že se rovná periodě (střída 100 %), je výstup trvale ve stavu log. 1. Nastavení šířky pulzu na 0 (střída 0 %) způsobí, že výstup je v log. 0.

Tabulka 6-35 Šířka pulzu, perioda a odezva funkce PWM

Šířka pulzu, perioda	Odezva
Šířka pulzu >= Hodnota periody	Střída je 100 %: výstup je trvale zapnutý.
Šířka pulzu = 0	Střída je 0 %: výstup je vypnutý.
Perioda < 2 časové jednotky	Perioda je implicitně nastavena na dvě časové jednotky.

Existují dva různé způsoby, jak změnit charakteristiku PWM signálu:

- Synchronní aktualizace: Pokud není třeba měnit časovou základnu, je možné použít synchronní aktualizaci. U synchronní aktualizace se změna tvaru signálu děje na konci periody; proto dochází jen k hladkým přechodům.
- Asynchronní aktualizace: Pro operaci PWM je typické, že se mění šířka pulzu, zatímco perioda zůstává konstantní; proto se nepožaduje změna časové základny. Když je však změna časové základny generátoru PTO/PWM požadována, používá se asynchronní aktualizace. Ta zastaví PTO/PWM generátor asynchronně s PWM signálem. To může být příčinou nežádoucího nestabilního chování řízeného zařízení. Z tohto důvodu se doporučuje používat synchronní způsob. Zvolte si takovou časovou základnu, která vyhovuje všem předpokládaným délkám periody.

Tip

Příznakový bit metody aktualizace PWM (SM67.4 nebo SM77.4) v řídicím bytu specifikuje typ aktualizace použitý při provádění instrukce PLS.

Při změně časové základny dojde k asynchronní aktualizaci nezávisle na stavu příznakového bitu příznaku způsobu aktualizace PWM.

Použití míst v paměti SM pro konfiguraci a řízení operací PTO/PWM

Instrukce PLS čte data uložená na specifikovaných místech paměti SM a programuje podle toho generátor PTO/PWM. SMB67 řídí PTO 0 nebo PWM 0 a SMB77 řídí PTO 1 nebo PWM 1. Tabulka 6-36 popisuje registry použité pro řízení funkcí PTO/PWM. Tabulku 6-37 můžete použít pro rychlé určení hodnoty, kterou je třeba uložit do řídicího registru PTO/PWM pro spuštění požadované operace.

Charakteristiku signálu PTO nebo PWM můžete změnit úpravou bytů v oblasti paměti SM (včetně řídicího bytu) a následujícím provedením instrukce PLS. Generování signálu PTO nebo PWM můžete kdykoliv pozastavit zapsáním 0 do povolovacího bitu PTO/PWM v řídicím bytu (SM67.7 nebo SM77.7) a následujícím provedením instrukce PLS.

Bit ”PTO nečinný” ve stavovém bytu (SM66.7 nebo SM76.7) slouží pro indikaci ukončení naprogramované sekvence pulzu. Dále je možné po ukončení sekvence vyvolat přerušení. (Viz popis instrukcí přerušení a komunikace.) Pokud používáte zřetězení více segmentů, je přerušovací podprogram vyvolán po dokončení tabulky profilu.

Následující podmínky nastavují SM66.4 (nebo SM76.4) a SM66.5 (nebo SM76.5):

- Při zadání hodnoty změny periody tak, že je výsledná perioda po daném počtu pulzů mimo povolený rozsah, se objeví matematické přetečení, které ukončí funkci PTO a nastaví bit chyby výpočtu změny periody (SM66.4 nebo SM76.4) na 1.Výstup se přepne na řízení pomocí registru obrazu procesu.
- Ruční přerušení (provádění) profilu PTO během činnosti nastaví bit přerušení uživatelem (SM66.5 nebo SM76.5) na 1.
- Při pokusu o nahrání atributů do plného ”řetězu” se nastaví bit přetečení PTO (SM66.6 nebo SM76.6) na 1. Pokud chcete detekovat další přetečení, je nutné po přetečení tento bit ručně vymazat. Přechod do režimu RUN nastaví tento bit na 0.



Tip

Jestliže chcete nastavit novou hodnotu počtu pulzů (SMD72 nebo SMD82), šířky pulzu (SMW70 nebo SMW80) nebo periody (SMW68 nebo SMW78), nastavte také příslušné aktualizační bity v řídicím registru dříve, než provedete instrukci PLS. Při použití režimu s více segmenty je též nutné před provedením instrukce PLS zapsat počáteční offset (SMW168 nebo SMW178) tabulky profilu.

Tabulka 6-36 Umístění řídicích registrů PTO / PWM v paměti SM

Q0.0	Q0.1	Stavové bity		
SM66.4	SM76.4	Profil PTO přerušen (chyba výpočtu změny):	0 = bez chyby	1 = přerušeno
SM66.5	SM76.5	Profil PTO přerušen uživatelským příkazem:	0 = bez přerušení	1 = přerušení
SM66.6	SM76.6	Přetečení/nenaplnění ”řetězu” PTO:	0 = bez přetečení	1 = přetečení / nenaplnění
SM66.7	SM76.7	PTO nečinný:	0 = postupuje	1 = PTO nečinný
Q0.0	**Q0.1**	**Řídicí bity**		
SM67.0	SM77.0	Příznak aktualizace periody PTO/PWM:	0 = bez aktualizace	1 = aktualizuje periodu
SM67.1	SM77.1	Příznak aktualizace času šířky pulzu PWM: šířku pulzu	0 = bez aktualizace	1 = aktualizuje
SM67.2	SM77.2	Příznak aktualizace počtu pulzů PTO:	0 = bez aktualizace	1 = aktualizuje počet pulzů
SM67.3	SM77.3	Volba časové základny PTO/PWM:	0 = 1#σs/pulz	1 = 1 ms/pulz
SM67.4	SM77.4	Příznak metody aktualizace PWM:	0 = asynchronní	1 = synchronní
SM67.5	SM77.5	Režim PTO s jedním/více segmenty:	0 = s 1 segmentem	1 = s více segmenty
SM67.6	SM77.6	Volba režimu PTO/PWM:	0 = PTO	1 = PWM
SM67.7	SM77.7	Povolení PTO/PWM:	0 = PTO/PWM pozastaven 1 = PTO/PWM povolen	
Q0.0	**Q0.1**	**Další registry PTO/PWM**		
SMW68	SMW78	Hodnota periody PTO/PWM	: 2 až 65 535	
SMW70	SMW80	Hodnota šířky pulzu PWM	: 0 až 65 535	
SMD72	SMD82	Hodnot počtu pulzů PTO	: 1 až 4 294 967 295	
SMB166	SMB176	Počet zpracovávaných segmentů	Pouze-při režimu PTO s více segmenty	
SMW168	SMW178	Začáteční adresa tabulky profilu (posunutí v bytech od V0)	Pouze-při režimu PTO s více segmenty	
SMB170	SMB180	Stavový byte lineárního profilu		
SMB171	SMB181	Výsledný registr lineárního profilu		
SMD172	SMD182	Registr frekvence ručního režimu		

Tabulka 6-37 Řídicí byte PTO/PWM

Řídicí registr (Hexa hodnota)		Výsledek provedení instrukce PLS						
	Povolit	Režim výběru	PTO režim segmentů	PWM Metoda aktualizace	Časová základna	Počet pulzů	Šířka pulzů	Doba cyklu
16#81	Ano	PTO	Jediný		1 σs/perioda			Zadat
16#84	Ano	PTO	Jediný		1 σs/perioda	Zadat		
16#85	Ano	PTO	Jediný		1 σs/perioda	Zadat		Zadat
16#89	Ano	PTO	Jediný		1 ms/perioda			Zadat
16#8C	Ano	PTO	Jediný		1 ms/perioda	Zadat		
16#8D	Ano	PTO	Jediný		1 ms/perioda	Zadat		Zadat
16#A0	Ano	PTO	Více		1 σs/perioda			
16#A8	Ano	PTO	Více		1 ms/perioda			
16#D1	Ano	PWM		Synchronní	1 σs/perioda			Zadat
16#D2	Ano	PWM		Synchronní	1 σs/perioda		Zadat	
16#D3	Ano	PWM		Synchronní	1 σs/perioda		Zadat	Zadat
16#D9	Ano	PWM		Synchronní	1 ms/perioda			Zadat

Tabulka 6-37 Řídicí byte PTO/PWM, pokračování

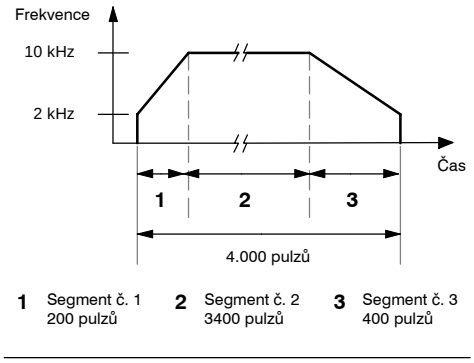
16#DA	Ano	PWM		Synchronní	1 ms/perioda		Zadat	
16#DB	Ano	PWM		Synchronní	1 ms/perioda		Zadat	Zadat

Výpočet hodnot tabulky profilu

Režim generátorů PTO/PWM s řetězením více segmentů je užitečný v mnoha aplikacích, obzvláště při řízení krokového motoru.

Funkci PTO s profilem pulzu můžete například použít na řízení krokového motoru v průběhu jednoduché sekvence rozběhu, provozu a doběhu nebo pro komplikovanější sekvence tak, že definujete profil pulzu, který se skládá až z 255 segmentů, kde každý segment odpovídá operaci rozběhu, chodu nebo doběhu.

Obrázek 6-31 ukazuje příklad hodnot tabulky profilu potřebných pro generování výstupního signálu, který rozběhne krokový motor (segment 1), udrží motor na konstantní rychlosti (segment 2) a zpomalí ho (segment 3).



Obr. 6-31 Časově frekvenční diagram

V tomto příkladu: Počáteční a konečná frekvence pulzů je 2 kHz, maximální frekvence pulzů je 10 kHz. Pro dosažení požadovaných otáček motoru je zapotřebí 4000 pulzů. Protože hodnoty v tabulce profilu jsou vyjádřeny v počtech period místo frekvence, je třeba převést dané hodnoty frekvence na periodu. Proto je počáteční (výchozí) a konečná (závěrečná) perioda 500#ōs a perioda odpovídající maximální frekvenci je 100#ōs. Během rozběhové fáze profilu má být dosaženo maximální frekvence pulzů po přibližně 200 pulzech. Zpomalovací část profilu má být ukončena přibližně za 400 pulzů.

Pro stanovení hodnoty změny periody daného segmentu, který generátor PTO/PWM používá pro přizpůsobení periody každého pulzu, můžete použít následující vzorec:

$$\text{Změna periody pro daný segment} = | \text{End_CT}_{seg} - \text{Init_CT}_{seg} | / \text{Quantity}_{seg}$$

kde: End_CT_{seg} = Konečná perioda pro daný segment
Init_CT_{seg} = Počáteční perioda pro daný segment
Quantity_{seg} = Počet pulzů v tomto segmentu

Použití tohoto vzorce pro výpočet hodnot změny periody pro daný příklad:

Segment 1 (rozběh):
Změna periody = −2

Segment 2 (konstantní rychlost):
Změna periody = 0

Segment 3 (zpomalení):
Změna periody = 1

Tabulka 6-38 uvádí hodnoty pro generování signálu z příkladu (za předpokladu, že tabulka profilu se nachází v paměti V se začátkem na V500). Do svého programu můžete zahrnout instrukce pro zapsání těchto hodnot do paměti V nebo můžete hodnoty profilu definovat v datovém bloku programu.

Tabulka 6-38 Hodnoty tabulky profilu

Adresa	Hodnota	Popis	
VB500	3	Celkový počet segmentů	
VW501	500	Počáteční perioda	Segment 1
VW503	−2	Počáteční změna periody	
VD505	200	Počet pulzů	
VW509	100	Počáteční perioda	Segment 2
VW511	0	Změna periody	
VD513	3400	Počet pulzů	
VW517	100	Počáteční perioda	Segment 3
VW519	1	Změna periody	
VD521	400	Počet pulzů	

Abyste stanovili, zda jsou přechody mezi segmenty signálu přijatelné, potřebujete určit periodu posledního pulzu v segmentu. Pokud není změna periody 0, musíte vypočítat periodu posledního pulzu v segmentu, protože tato hodnota není přímo dána v tabulce profilu. Pro výpočet periody posledního pulzu použijte následující vzorec:

Perioda posledního pulzu v segmentu = $Init_CT_{seg}$ + ($Delta_{seg}$ * ($Quantity_{seg}$ – 1))

kde: $Init_CT_{seg}$ = Počáteční perioda pro daný segment

$Delta_{seg}$ = Změna periody pro daný segment

$Quantity_{seg}$ = Počet pulzů v tomto segmentu

Výše uvedený zjednodušený příklad je užitečný jako úvod, ale skutečná použití mohou používat podstatně složitější profily. Mějte na paměti, že změna periody může být zadána jen jako celé číslo v mikrosekundách nebo milisekundách a změny periody se provádí při každém pulzu.

Výsledkem je, že výpočet změny periody pro daný segment může vyžadovat iterativní přístup. Pravděpodobně bude nutné použít proměnné hodnoty času poslední periody, případně počtu pulzů pro daný segment.

Doba trvání daného segmentu profilu může být užitečná při určování správných hodnot tabulky profilu. Na výpočet času potřebného k provedení daného segmentu se používá následující vzorec:

Doba trvání segmentu = $Quantity_{seg}$ * ($Init_CT$ + (($Delta_{seg}$/2) * ($Quantity_{seg}$ – 1)))

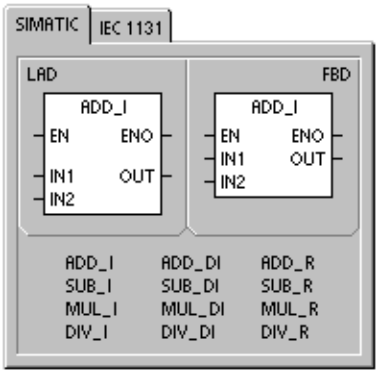
kde: $Quantity_{seg}$ = Počet pulzů v tomto segmentu

$Init_CT_{seg}$ = Počáteční perioda pro daný segment

$Delta_{seg}$ = Změna periody pro daný segment

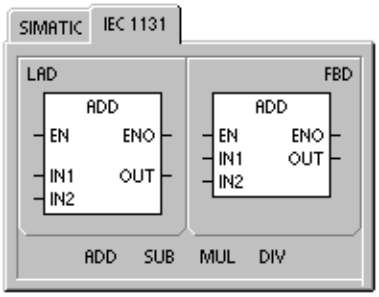
Matematické instrukce

Sčítání, odčítání, násobení a dělení



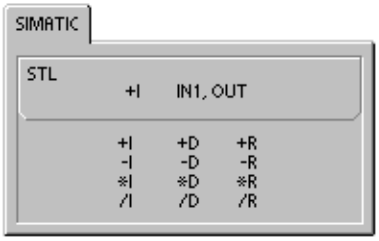
Sčítání	Odčítání	
IN1 + IN2 = OUT	IN1 – IN2 = OUT	*LAD a FBD*
IN1 + OUT = OUT	OUT – IN1 = OUT	*STL*

Instrukce ”Sečíst integer” (+I) a ”Odečíst integer” (–I) sečte nebo odečte dvě 16bitová celá čísla (integer) – výsledkem je 16bitová hodnota. Instrukce ”Sečíst double integer” (+D) nebo ”Odečíst double integer” (–D) přičte a odečte dvě 32bitová celá čísla (integer), výsledkem je 32bitová hodnota. Instrukce ”Sečíst real” (+R) nebo ”Odečíst real” (–R) přičte a odečte dvě 32bitová reálná čísla – výsledkem je 32bitová hodnota (reálné číslo).



Násobení	Dělení	
IN1 * IN2 = OUT	IN1 / IN2 = OUT	*LAD a FBD*
IN1 * OUT = OUT	OUT / IN1 = OUT	*STL*

Instrukce ”Násobit integer” (*I) a ”Dělit integer” (/I) vynásobí a vydělí dvě 16bitová celá čísla (integer) – výsledkem je 16bitová hodnota. (Při dělení se neuchovává zbytek.) Instrukce ”Násobit double integer” (*D) a ”Dělit double integer” (/D) vynásobí a vydělí dvě 32bitová celá čísla (integer) – výsledkem je 32bitová hodnota. (Při dělení se neuchovává zbytek.) Instrukce ”Násobit real” (*R) a ”Dělit real” (/R) vynásobí a vydělí dvě 32bitová reálná čísla – výsledkem je 32bitová hodnota (reálné číslo).



Bity SM a ENO

SM1.1 indikuje chyby přetečení a neplatné hodnoty. Pokud je nastaven SM1.1, je stav SM1.0 a SM1.2 neplatný a původní vstupní operandy se nemění. Pokud SM1.1 a SM1.3 nejsou nastaveny, skončila matematická operace s platným výsledkem a SM1.0 a SM1.2 obsahují platný stav. Pokud je SM1.3 nastaven během operace dělení, zůstávají ostatní stavové bity matematických operací nezměněny.

Chyby, které nastavují ENO = 0

- SM1.1 (přetečení)
- SM1.3 (dělení nulou)
- 0006 (nepřímá adresa)

Ovlivněné bity speciální paměti

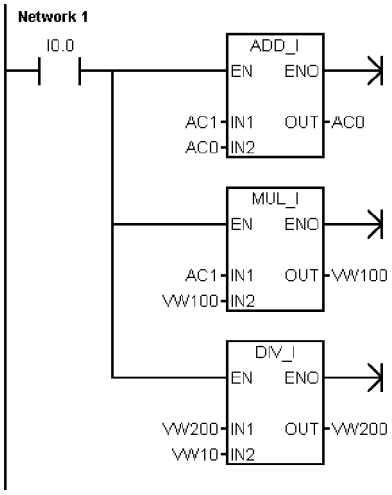
- SM1.0 (nula)
- SM1.1 (přetečení, během operace generována neplatná hodnota nebo zjištěn neplatný vstupní parametr)
- SM1.2 (záporný výsledek)
- SM1.3 (dělení nulou)

Tabulka 6-39 Operandy platné pro instrukce sčítání, odčítání, násobení a dělení

Vstupy/Výstupy	Typy dat	Operandy
IN1, IN2	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *AC, *LD, konstanta
	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, HC, *VD, *LD, *AC, konstanta
	REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC, konstanta
OUT	INT	IW, QW, VW, MW, SMW, SW, LW, T, C, AC, *VD, *AC, *LD
	DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

Reálná čísla (čísla s pohyblivou řádovou čárkou) jsou reprezentována ve formátu popsaném v normě ANSI/IEEE 754–1985 (jednoduchá přesnost). Další informace najdete v této normě.

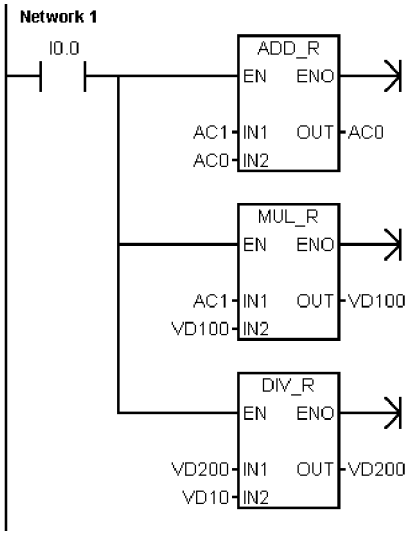
Příklad: Matematika celých čísel




```
Network 1
LD    I0.0
+I    AC1, AC0
*I    AC1, VW100
/I    VW10, VW200
```

Sčítání				Násobení				Dělení			
40	+ 60	= 100		40	* 20	= 800		4000	/ 40	= 100	
AC1	AC0	AC0		AC1	VW100	VW100		VW200	VW10	VW200	

Příklad: Matematika reálných čísel




```
Network 1
LD    I0.0
+R    AC1, AC0
*R    AC1, VD100
/R    VD10, VD200
```

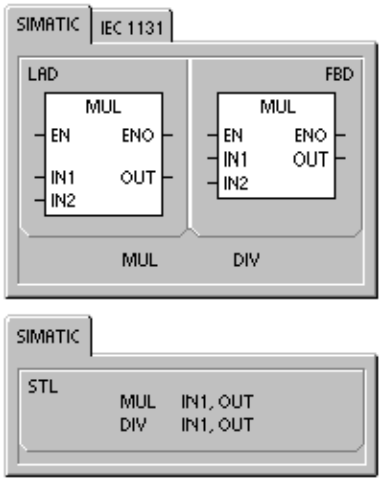
Sčítání				Násobení				Dělení			
4000,0	+ 6000,0	= 10000,0		400,0	* 200,0	= 80000,0		4000,0	/ 41,0	= 97,5609	
AC1	AC0	AC0		AC1	VD100	VD100		VD200	VD10	VD200	

Násobení celých čísel s 32bitovým výsledkem a dělení celých čísel se zbytkem

Násobení integer s 32bitovým výsledkem

IN1 * IN2 = OUT LAD a FBD
IN1 * OUT = OUT STL

Instrukce "Násobit Integer do double integer" (MUL) vynásobí dvě 16bitová celá čísla (integer) – výsledkem je 32bitová hodnota (double integer). V STL instrukci MUL se nejnižší platný word (16 bitů) z 32bitového OUT používá jako jeden z činitelů.



Dělení integer se zbytkem

IN1 / IN2 = OUT LAD a FBD
OUT / IN1 = OUT STL

Instrukce "Dělit integer se zbytkem" (DIV) vydělí dvě 16bitová celá čísla (integer) s 32bitovým výsledkem (double integer), který se skládá z 16bitového zbytku (nejvyšší word) a 16bitového podílu (nejnižší word).

V STL se nejnižší platná hodnota word (16 bitů) z 32bitového OUT používá jako dělenec.

Bity SM a ENO

U obou instrukcí indikují na této stránce bity speciální paměti (SM) chyby a neplatné hodnoty. Pokud je během dělení nastavena SM1.3 (dělení nulou), zůstávají ostatní stavové bity matematiky nezměněny. Jinak všechny bity matematiky obsahují po dokončení matematické operace platný stav.

Chyby, které nastavují ENO = 0

- SM1.1 (přetečení)
- SM1.3 (dělení nulou)
- 0006 (nepřímá adresa)

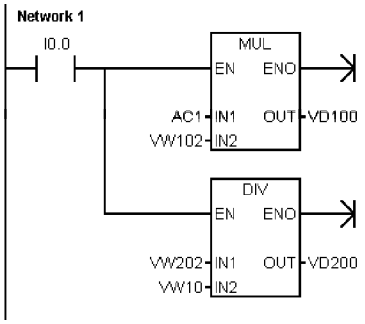
Ovlivněné bity speciální paměti

- SM1.0 (nula)
- SM1.1 (přetečení)
- SM1.2 (záporný výsledek)
- SM1.3 (dělení nulou)

Tabulka 6-40 Operandy platné pro násobení celých čísel s 32bitovým výsledkem a dělení celých čísel se zbytkem

Vstupy/Výstupy	Typy dat	Operandy
IN1, IN2	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
OUT	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

Příklad: Instrukce násobení celých čísel s 32bitovým výsledkem a instrukce dělení celých čísel se zbytkem




```
Network 1
LD    I0.0
MUL   AC1, VD100
DIV   VW10, VD200
```

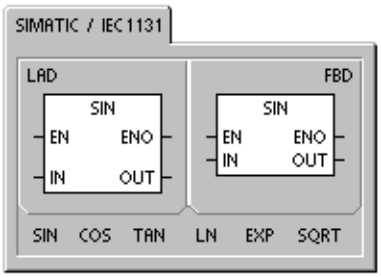
Násobení integer s 32bitovým výsledkem: 400 (AC1) * 200 (VW102) = 80000 (VD100)

Dělení integer se zbytkem: 4000 (VW202) / 41 (VW10) = zbyt. 23 (VW200), podíl 97 (VW202) — VD200

Poznámka: VD100 obsahuje VW100 a VW102; VD200 obsahuje VW200 a VW202.

Numerické funkce

Sinus, kosinus a tangens

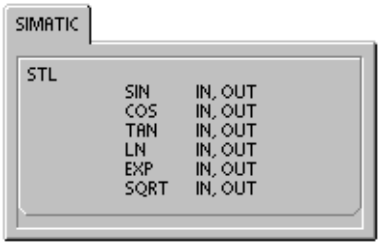


Instrukce ”Sinus” (SIN), ”Cosinus” (COS) a ”Tangens” (TAN) vypočítají trigonometrickou funkci úhlu daného v IN a výsledek zapíšou do OUT. Hodnota vstupního úhlu je radiánech.

SIN (IN) = OUT COS (IN) = OUT TAN (IN) = OUT

Chcete–li převést úhel ze stupňů na radiány: Použijte instrukci MUL_R (*R) pro vynásobení úhlu ve stupních koeficientem 1,745329E–2 (přibližne ϕ/180).

Přirozený logaritmus a přirozený exponent



Instrukce ”Přirozený logaritmus” (LN) provede přirozený logaritmus z hodnoty IN a výsledek uloží do OUT.

Instrukce ”Přirozený exponent” (EXP) umocní e na exponent uvedený v parametru IN a výsledek zapíše do OUT.

LN (IN) = OUT EXP (IN)= OUT

Chcete–li získat z přirozených logaritmů dekadické logaritmy: Vydělte přirozený logaritmus 2,302585 (přibližně přirozený logaritmus 10).

Chcete–li vypočítat libovolnou reálnou mocninu reálného čísla včetně zlomkových exponentů: Zkombinujte instrukci ”Přirozený exponent” s instrukcí ”Přirozený logaritmus”. Chcete–li například umocnit X na Y, vložte následující instrukci: EXP (Y * LN (X)).

Druhá odmocnina

Instrukce ”Druhá odmocnina” (SQRT) vypočítá druhou odmocninu z reálného čísla (IN) a uloží ji do reálného výsledku v OUT.

SQRT (IN)= OUT

Chcete–li vypočítat jiné odmocniny:

5 na třetí = 5 ^ 3 = EXP(3*LN(5)) = 125
Třetí odmocnina 125 = 125 ^ (1/3) = EXP((1/3)*LN(125))= 5
Druhá odmocnina 5 na třetí = 5 ^ (3/2) = EXP(3/2*LN(5)) = 11,18034

Bity SM a ENO pro instrukce numerických funkcí

U všech instrukcí popsaných na této stránce je SM1.1 použit pro indikaci chyb přetečených a neplatných hodnot. Pokud je nastaven SM1.1, je stav SM1.0 a SM1.2 neplatný a původní vstupní operandy se nemění. Pokud SM1.1 není nastaven, skončila matematická operace s platným výsledkem a SM1.0 a SM1.2 obsahují platný stav.

Chyby, které nastavují ENO = 0

- SM1.1 (přetečení)
- 0006 (nepřímá adresa)

Ovlivněné bity speciální paměti

- SM1.0 (nula)
- SM1.1 (přetečení)
- SM1.2 (záporný výsledek)

Tabulka 6-41 Operandy platné pro numerické funkce

Vstupy/Výstupy	Typy dat	Operandy
IN	REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC, konstanta
OUT	REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

Reálná čísla (čísla s pohyblivou řádovou čárkou) jsou reprezentována ve formátu popsaném v normě ANSI/IEEE 754–1985 (jednoduchá přesnost). Další informace najdete v této normě.

Inkrementace a dekrementace

Inkrementace

IN +1 = OUT *LAD a FBD*
OUT +1 = OUT *STL*

Dekrementace

IN – 1 = OUT *LAD a FBD*
OUT – 1 = OUT *STL*

Instrukce inkrementace a dekrementace připočítávají 1 ke vstupu IN nebo odpočítávají 1 od vstupu IN a výsledek zapisují do proměnné dané v OUT.

Operace ”Inkrementovat byte” (INCB) a ”Dekrementovat byte” (DECB) jsou bez znaménka.

Operace ”Inkrementovat word” (INCW) a ”Dekrementovat word” (DECW) jsou se znaménkem.

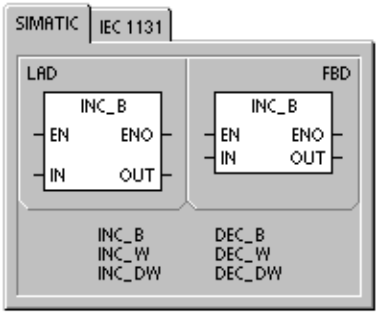
Operace ”Inkrementovat double word” (INCD) a ”Dekrementovat double word” (DECD) jsou se znaménkem.

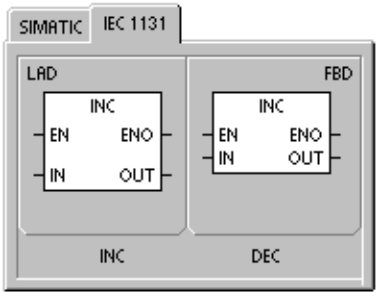
Chyby, které nastavují ENO = 0:

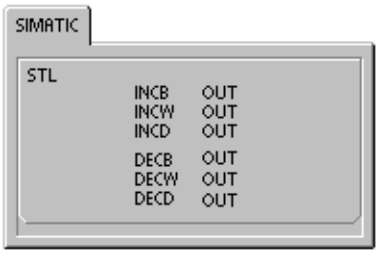
- SM1.1 (přetečení)
- 0006 (nepřímá adresa)

Ovlivněné bity speciální paměti:

- SM1.0 (nula)
- SM1.1 (přetečení)
- SM1.2 (záporný výsledek) pro operace s word a double word



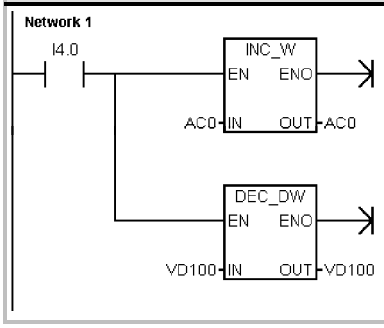




Tabulka 6-42 Operandy platné pro instrukce ”Inkrementace” a ”Dekrementace”

Vstupy/Výstupy	Typy dat	Operandy
IN	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, HC, *VD, *LD, *AC, konstanta
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD
	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC,*VD, *LD, *AC
	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

Příklad: Inkrementace a dekrementace



Network 1

```
LD    I4.0
INCW  AC0
DECD  VD100
```

Inkrementace word	125	+ 1 =	126	
	AC0		AC0	
Dekrementace double word	128000	– 1 =	127999	
	VD100		VD100	

Instrukce PID algoritmu

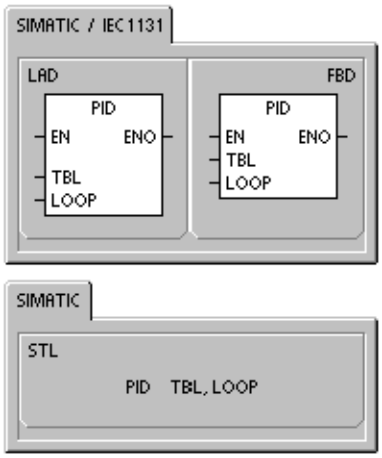
Instrukce ”PID algoritmus” (PID) provede výpočet PID smyčky na základě vstupu a konfigurace dané v tabulce (TBL).

Chyby, které nastavují ENO = 0:

- SM1.1 (přetečení)
- 0006 (nepřímá adresa)

Ovlivněné bity speciální paměti:

- SM1.1 (přetečení)



Instrukce ”PID algoritmus” (PID = proporcionální, integrační a derivační regulátor) se používá pro provedení výpočtu PID. Aby mohl být výpočet PID proveden, musí být vrchol zásobníku (TOS) ve stavu log. 1 (signálový tok). Instrukce má dva operandy: TABLE – adresa, kde začíná tabulka algoritmu, LOOP – konstanta, číslo od 0 do 7.

V programu může být použito až osmi instrukcí PID. Pokud mají dvě nebo více instrukcí PID stejné číslo (i kdyby měly různé adresy tabulek), budou se výpočty PID vzájemně rušit a výstup bude nepředvídatelný.

V tabulce algoritmu je uloženo devět parametrů používaných pro řízení a monitorování činnosti algoritmu. Tabulka obsahuje aktuální a předcházející hodnoty regulované veličiny, požadovanou hodnotu výstupu, zesílení, vzorkovací periodu, integrační časovou konstantu (reset), derivační časovou konstantu (rate) a integrační sumu (bias).

Aby byl PID výpočet proveden s požadovanou vzorkovací periodou, musí být instrukce PID vykonána buď v přerušení, nebo v hlavním programu, kde musí být rychlost řízena časovačem. Vzorkovací perioda musí být zadána jako vstup smyčky PID pomocí tabulku algoritmu.

Do instrukce PID byla začleněna schopnost automatického ladění. Podrobný popis automatického ladění naleznete v kapitole 15. Ovládací panel ladění PID funguje pouze se smyčkami PID vytvořenými průvodcem PID.

Tabulka 6-43 Operandy platné pro Instrukci ”PID algoritmus”

Vstupy/Výstupy	Typy dat	Operandy
TBL	BYTE	VB
LOOP	BYTE	Konstanta (0 až 7)


Průvodce

STEP 7-Micro/WIN nabízí průvodce, který vás provede definováním algoritmu PID pro řídicí proces s uzavřenou řídicí smyčkou. Vyberte příkaz menu **Nástroje > Průvodce instrukcemi** a pak vyberte **PID** z okna průvodce.



Tip

Dolní mez a horní mez žádané hodnoty by měly odpovídat dolní mezi a horní mezi regulované veličiny.

Vysvětlení algoritmu PID

V ustáleném stavu provozu řídí regulátor PID hodnotu výstupu tak, aby odchylka (e) byla od požadované hodnoty nulová. Míra odchylky je dána rozdílem mezi požadovanou hodnotou (SP) a regulovanou veličinou (PV). Princip řízení PID je založen na následující rovnici, která vyjadřuje výstup M(t) jako funkci proporcionálního, integračního a derivačního členu:

Výstup	=	Proporcionální člen	+	Integrační člen	+	Derivační člen
M(t)	**=**	$K_C * e$	**+**	$K_C \int_0^t e\,dt + M_{initial}$	**+**	$K_C * de/dt$

kde:
- $M_{(t)}$ je výstup algoritmu jako funkce času
- K_C je zesílení
- e je regulační odchylka (rozdíl mezi požadovanou hodnotou a regulovanou veličinou)
- $M_{initial}$ je počáteční hodnota výstupu

Aby bylo možné implementovat tuto řídicí funkci v digitálním počítači, musí být původní spojitá funkce transformována na periodické vzorky regulační odchylky s následným přepočtem výstupu. Příslušná rovnice, která je základem pro implementaci regulačního algoritmu na digitálním počítači, je:

M_n	=	$K_c * e_n$	+	$K_I * \sum_1^n e_x + M_{initial}$	+	$K_D * (e_n - e_{n-1})$
výstup	=	proporcionální člen	+	integrační člen	+	derivační člen

kde:
- M_n je vypočítaná hodnota výstupu algoritmu ve vzorkovací periodě n
- K_C je zesílení
- e_n je hodnota regulační odchylky ve vzorkovací periodě n
- e_{n-1} je předcházející hodnota regulační odchylky (ve vzorkovací periodě n – 1)
- e_x je hodnota regulační odchylky ve vzorkovací periodě x
- K_I je proporcionální konstanta integračního členu
- $M_{initial}$ je počáteční hodnota výstupu
- K_D je proporcionální konstanta derivačního členu

Z rovnice vidíme, že integrační člen je funkcí všech odchylek od prvního po aktuální vzorek. Diferenční člen je funkcí aktuálního a předcházejícího vzorku, zatímco proporcionální člen je funkcí pouze aktuálního vzorku. Není praktické ani potřebné uchovávat v digitálním počítači všechny vzorky regulační odchylky.

Protože počítač musí počítat výstupní hodnotu při každém vzorkování odchylky počínaje prvním vzorkem, je nutné uložit pouze předcházející hodnotu odchylky a předcházející hodnotu integračního členu. V důsledku repetiční povahy řešení je možné provést zjednodušení rovnice, která je řešena v každém vzorkovacím cyklu. Zjednodušená rovnice vypadá takto:

M_n	=	$K_c * e_n$	+	$K_I * e_n + MX$	+	$K_D * (e_n - e_{n-1})$
výstup	=	proporcionální člen	+	integrační člen	+	derivační člen

kde:
- M_n je vypočítaná hodnota výstupního algoritmu ve vzorkovací periodě n
- K_C je zesílení
- e_n je hodnota regulační odchylky ve vzorkovací periodě n
- e_{n-1} je předcházející hodnota regulační odchylky (ve vzorkovací periodě n – 1)
- K_I je proporcionální konstanta integračního členu
- MX je předcházející hodnota integračního členu (ve vzorkovací periodě n – 1)
- K_D je proporcionální konstanta derivačního členu

Programovatelný automat S7-200 používá pro výpočet výstupní hodnoty smyčky modifikovanou formu výše uvedené zjednodušené rovnice. Modifikovaná rovnice je:

M_n	=	MP_n	+	MI_n	+	MD_n
výstup	=	proporcionální člen	+	integrační člen	+	derivační člen

kde:
- Mn je vypočítaná hodnota výstupního algoritmu ve vzorkovací periodě n
- MP_n je hodnota proporcionálního členu výstupu algoritmu ve vzorkovací periodě n
- MI_n je hodnota integračního členu výstupu algoritmu ve vzorkovací periodě n
- MDn je hodnota derivačního členu výstupu algoritmu ve vzorkovací periodě n

Vysvětlení proporcionálního členu rovnice PID

Proporcionální člen MP je součinem zesílení (K_C), které řídí citlivost výstupního výpočtu, a odchylky (e), která představuje rozdíl mezi požadovanou hodnotou (SP) a regulovanou veličinou (PV) v dané vzorkovací periodě. Rovnice pro proporcionální člen, jak ji řeší S7-200, zní:

MP_n	=	K_C	*	$(SP_n - PV_n)$

kde:
- MPn je hodnota proporcionálního členu výstupu algoritmu ve vzorkovací periodě n
- K_C je zesílení
- SP_n je požadovaná hodnota ve vzorkovací periodě n
- PV_n je hodnota regulované veličiny ve vzorkovací periodě n

Vysvětlení integračního členu rovnice PID

Integrační člen MI je přímo úměrný součtu regulačních odchylek v průběhu času. Rovnice pro integrační člen, jak ji řeší S7-200, zní:

MI_n	=	K_C	*	T_S	/	T_I	*	$(SP_n - PV_n)$	+	MX

kde:
- MI_n je hodnota integračního členu výstupu algoritmu ve vzorkovací periodě n
- K_C je zesílení
- T_S je vzorkovací perioda
- T_I je integrační konstanta (také nazývaná integrační časová konstanta nebo reset)
- SP_n je požadovaná hodnota ve vzorkovací periodě n
- PV_n je hodnota regulované veličiny ve vzorkovací periodě n
- MX je hodnota integračního členu ve vzorkovací periodě n – 1 (nazývaný také integrační suma)

Integrační suma (MX) je průběžný součet všech předcházejících hodnot integračního členu. Po každém výpočtu MI_n je integrační suma aktualizována hodnotou MI_n, která může být přízpůsobena nebo omezena (podrobnosti viz odstavec ”Proměnné a rozsahy”). Počáteční hodnota integračního součtu je nastavena na ($M_{initial}$) těsně před výpočtem prvního výstupu algoritmu. Součástí integračního členu je také několik konstant, zesílení (K_C), vzorkovací perioda (T_S), což je perioda, po kterém PID algoritmus opět počítá výstupní hodnotu, a integrační časová konstanta neboli reset (T_I), což je čas použitý pro ovlivnění výpočtu výstupu integračním členem.

Vysvětlení derivačního členu rovnice PID

Derivační člen MD je přímo úměrný změně odchylky. Programovatelný automat S7-200 používá pro derivační člen následující rovnici:

$$MD_n = K_C * T_D / T_S * ((SP_n - PV_n) - (SP_{n-1} - PV_{n-1}))$$

Aby se zabránilo skokovým změnám a nárazům na výstupu způsobeným derivačním charakterem změny požadované hodnoty, je tato rovnice upravena na základě předpokladu, že požadovaná hodnota je konstanta ($SP_n = SP_{n-1}$). Výsledkem je výpočet změny regulované veličiny namísto změny odchylky, jak je patrné z rovnice:

$$MD_n = K_C * T_D / T_S * (SP_n - PV_n - SP_n + PV_{n-1})$$

nebo jen:

$$MD_n = K_C * T_D / T_S * (PV_{n-1} - PV_n)$$

kde:

MD_n	je hodnota derivačního členu výstupu algoritmu ve vzorkovací periodě n
K_C	je zesílení
T_S	je vzorkovací perioda
T_D	je derivační konstanta (také nazývaná derivační časová konstanta nebo rate)
SP_n	je požadovaná hodnota ve vzorkovací periodě n
SP_{n-1}	je požadovaná hodnota ve vzorkovací periodě n–1
PV_n	je hodnota regulované veličiny ve vzorkovací periodě n
PV_{n-1}	je hodnota regulované veličiny ve vzorkovací periodě n–1

Pro další výpočet derivačního členu je třeba uchovat spíše regulovanou veličinu než odchylku. V době prvního vzorku je hodnota PV_{n-1} inicializována, aby se rovnala PV_n.

Výběr typu řídícího algoritmu

V mnoha aplikacích může být potřeba použít pouze jednu nebo dvě složky řídicího algoritmu. Může být například požadováno pouze proporcionální řízení nebo proporcionální a integrační řízení. Výběr požadovaného typu řízení se provádí nastavením hodnoty konstantních parametrů.

Pokud chcete vynechat integrační složku (PD regulátor), musí být pro integrační časovou konstantu (reset) zadáno jako hodnota nekonečno "INF". I bez integrace nemusí být hodnota integračního členu nulová díky původní hodnotě integračního součtu MX.

Pokud chcete vynechat derivační složku (PI regulátor), musí být pro derivační časovou konstantu (rate) zadána hodnota 0,0.

Pokud chcete vynechat proporcionální složku a chcete řízení I nebo ID, musí být pro zesílení zadána hodnota 0,0. Protože je v rovnici zesílení smyčky faktorem pro výpočet integračního a derivačního členu, bude mít ve výpočtech integračních a derivačních členů nastavení hodnoty 0,0 pro zesílení smyčky za důsledek použití hodnoty 1,0.

Konverze a normalizace vstupů algoritmu

Algoritmus má dvě vstupní proměnné, požadovanou hodnotu a regulovanou veličinu. Požadovaná hodnota je zpravidla konstantní hodnota, jako je například nastavení rychlosti pohybu. Regulovaná veličina je hodnota, která je svázána s výstupem algoritmu, a měří proto vliv výstupního algoritmu na řízený systém. Ve výše uvedeném příkladu nastavení rychlosti by regulovanou veličinou byl vstup do tachodynama, které měří rychlost kol otáčení.

Požadovaná hodnota i regulovaná veličina jsou hodnoty reálného světa, jejichž velikost, rozsah a jednotky mohou být různé. Před zpracováním těchto hodnot reálného světa pomocí instrukce PID je třeba převést je na normalizovaná čísla s pohyblivou řádovou čárkou.

Prvním krokem je převést hodnotu reálného světa z 16bitového celého čísla (integer) na číslo s pohyblivou řádovou čárkou neboli reálné číslo. Konverze integer na reálné číslo se provádí podle následující sekvence instrukcí.

```
ITD     AIW0, AC0      //Konverze vstupní hodnoty na hodnotu double word
DTR     AC0,  AC0      //Konverze- 32bitového celého čísla (double integer) na číslo reálné
```

Dalším krokem je konverze hodnoty reálného světa reprezentované reálným číslem na normalizovanou hodnotu z intervalu 0,0 až 1,0. Pro normalizaci požadované hodnoty nebo regulované veličiny se používá následující rovnice:

$$\mathbf{R_{Norm} = ((R_{Raw} / Span) + Offset)}$$

kde:

- R_{Norm} je normalizovaná hodnota reálného čísla reprezentujícího hodnotu reálného světa
- R_{Raw} je nenormalizovaná neboli nezpracovaná hodnota reálného čísla reprezentujícího hodnotu reálného světa
- Offset je 0,0 pro unipolární hodnoty; je 0,5 pro bipolární hodnoty
- Span je rozdíl maximální a minimální možné hodnoty: = 32.000 pro unipolární hodnoty (obvykle); = 64.000 pro bipolární hodnoty (obvykle)

Následující sekvence instrukcí ukazuje, jak normalizovat bipolární hodnotu v AC0 (s rozsahem 64.000) jako pokračování předcházející sekvence instrukcí:

```
/R      64000,0,  AC0  //Normalizace hodnoty v akumulátoru
+R      0,5,  AC0      //Offset hodnoty do rozsahu 0,0 až 1,0
MOVR    AC0,  VD100    //Uložení normalizované hodnoty do tabulky algoritmu.
```

Konverze výstupu na integer s upraveným měřítkem

Výstup algoritmu je řídicí proměnná, například v našem příkladu o řízení rychlosti automobilu je to nastavení průtoku paliva. Výstup je normalizované reálné číslo mezi 0,0 a 1,0. Před použitím výstupu algoritmu na řízení analogového výstupu je nutné ho převést na 16bitový integer s upraveným měřítkem. Tento proces je opakem konverze PV a SP na normalizovanou hodnotu. Prvním krokem je převedení výstupu na reálné číslo s upraveným měřítkem pomocí následujícího vzorce:

$$R_{Scal} = (M_n - Offset) * Span$$

*k*de:
- R_{Scal} je reálná hodnota s upraveným měřítkem výstupu smyčky
- M_n je normalizovaná reálná hodnota výstupu smyčky
- Offset je 0,0 pro unipolární hodnoty; je 0,5 pro bipolární hodnoty
- Span je rozdíl maximální a minimální možné hodnoty = 32.000 pro unipolární hodnoty (obvykle) = 64.000 pro bipolární hodnoty (obvykle)

Následující sekvence instrukcí ukazuje, jak upravit měřítko výstupu algoritmu:

```
MOVR   VD108,  AC0      //Uloží výstup do akumulátoru
-R     0,5,  AC0        //Tuto instrukci provést pouze v případě bipolární hodnoty
*R     64000,0,  AC0    //Upraví měřítko hodnoty v akumulátoru
```

Teď je třeba převést reálné číslo s upraveným měřítkem, které představuje výstup smyčky, na 16bitový integer. Tato konverze se uskuteční pomocí následující sekvence instrukcí:

```
ROUND  AC0, AC0         //Konverze reálného čísla na 32bitové celé číslo
DTI    AC0,  LW0        //Konverze hodnoty na 16bitový integer
MOVW   LW0, AQW0        //Zápis hodnoty na analogový výstup
```

Přímá nebo reverzní smyčka

Smyčka je přímá, je–li zesílení kladné, a reverzní, je–li zesílení záporné. (U řízení I a ID, kde je hodnota zesílení 0,0, způsobí zadání kladné hodnoty pro integrační a derivační časovou konstantu přímou činnost smyčky, zadání záporné hodnoty způsobí reverzní činnost smyčky.)

Proměnné a rozsahy

Regulovaná veličina a požadovaná hodnota jsou vstupy do PID výpočtu. Proto může instrukce PID odpovídající políčka tabulky číst, ale nikoliv měnit.

Výstupní hodnota je generována výpočtem PID, proto je políčko výstupu v tabulce smyčky aktualizováno po dokončení každého výpočtu PID. Výstupní hodnota leží mezi 0,0 a 1,0. Políčko výstupní hodnoty může být použito jako vstup pro zadání počáteční hodnoty výstupu uživatelem při přechodu od ručního řízení výstupu na řízení výstupu pomocí instrukce PID (auto). (Další informace najdete v sekci ”Režimy”.)

Pokud se používá integrační řízení, je hodnota integračního součtu aktualizována výpočtem PID; aktualizovaná hodnota je použita jako vstup pro další výpočet PID. Jestliže je vypočítaná výstupní hodnota mimo rozsah (výstup by byl menší než 0,0 nebo větší než 1,0), upraví se integrační součet podle následujících vzorců:

$$MX = 1.0 - (MP_n + MD_n) \quad \textit{když je vypočítaný výstup } M_n > 1{,}0$$

nebo

$$MX = -(MP_n + MD_n) \quad \textit{když je vypočítaný výstup } M_n < 0{,}0$$

*k*de:
- MX je hodnota upraveného integračního součtu
- MP_n je hodnota proporcionálního členu výstupu algoritmu ve vzorkovací periodě n
- MD_n je hodnota derivačního členu výstupu algoritmu ve vzorkovací periodě n
- M_n je hodnota výstupu algoritmu ve vzorkovací periodě n

Upravením integračního součtu výše uvedeným způsobem se dosáhne zlepšení citlivosti systému, jakmile se vypočítaný výstup vrátí do správného rozsahu. Vypočítaný integrační součet se rovněž nachází v rozsahu 0,0 až 1,0; po dokončení každého výpočtu PID se zapíše do políčka pro integrační součet v tabulce algoritmu. Hodnota uložená v tabulce algoritmu je použita v příštím výpočtu PID.

Před provedením instrukce PID může uživatel modifikovat hodnotu integrační sumy v tabulce algoritmu, aby v určitých situacích vyřešil problémy s hodnotou integračního součtu. Při ruční úpravě integrační sumy se musí postupovat opatrně; jakákoliv hodnota integrační sumy zapsaná do tabulky algoritmu musí být reálné číslo v rozmezí 0,0 a 1,0.

Pro derivační část výpočtu PID je v tabulce uložena porovnávací hodnota regulované veličiny. Tato hodnota by neměla být modifikována uživatelem.

Režimy

PID algoritmu programovatelného automatu S7-200 nemají žádné zabudované řízení režimu. Výpočet PID se provádí pouze při aktivním signálovém toku do bloku PID. "Automatický" režim neboli "auto" znamená cyklické provádění výpočtu PID. "Manuální" režim existuje, když se výpočet PID neprovádí.

Instrukce PID má bit historie signálového toku, obdobně jako instrukce čítače. Tento bit používá instrukce pro detekci přechodu signálového toku z 0 na 1. Při detekci takového přechodu instrukce provede řadu činností, aby zabezpečila beznárazovou změnu z manuálního řízení na automatické. Aby byla změna na automatický režim řízení beznárazová, musí být hodnota výstupu (nastavená při manuálním řízení) nastavena i jako vstup pro instrukci PID (zapsána do položky tabulky smyčky pro M_n) dříve, než se přejde na automatické řízení. Instrukce PID provede s hodnotami v tabulce algoritmu následující činnosti, aby zajistila bezproblémovou změnu z manuálního na automatické řízení při přechodu signálového toku z 0 na 1:

- Požadovaná hodnota (SP_n) = regulovaná veličina (PV_n)
- Předchozí regulovaná veličina (PV_{n-1}) = regulovaná veličina (PV_n)
- Integrační suma (MX) = výstupní hodnota (M_n)

Implicitní nastavení bitů historie PID je log.1. Tento stav je při spuštění i při každém přechodu programovatelného automatu z režimu STOP do režimu RUN. Při prvním provádění smyčky PID po přepnutí do režimu RUN se při aktivním signálového toku nespustí činnost beznárazového přechodu.

Kontrola poruchových signálů a speciální funkce

Instrukce PID je jednoduchý, ale účinný nástroj pro provádění výpočtů PID. Pokud se požadují další funkce, jako je například kontrola poruchových signálů nebo proměnných smyčky, je nutné je implementovat pomocí základních instrukcí podporovaných automatem S7-200.

Chyby

Pokud jsou počáteční adresa tabulky algoritmu nebo operandy PID v instrukci mimo rozsah, CPU ohlásí při kompilaci kompilační chybu (chybu rozsahu) a kompilace se zastaví.

Některé vstupní hodnoty v tabulce nejsou instrukcí PID kontrolovány, zda se nacházejí v povoleném rozsahu. Je třeba věnovat velkou pozornost tomu, aby regulovaná veličina a požadovaná hodnota (stejně jako integrační součet a předcházející regulovaná veličina, pokud jsou použity jako vstup) byla reálná čísla v rozsahu mezi 0,0 a 1,0.

Pokud se v době provádění matematických operací výpočtu PID vyskytne jakákoliv chyba, nastaví se SM1.1 (přetečení nebo neplatná hodnota) a provádění instrukce PID se ukončí. (Aktualizace výstupních hodnot v tabulce algoritmu může být neúplná, proto tyto hodnoty neberte v úvahu a před příštím provedením instrukce PID opravte vstupní hodnotu, která matematickou chybu způsobila.)

Tabulka algoritmu

Tabulka má velikost 80 bytů a formát podle tabulky 6-44.

Tabulka 6-44 Tabulka smyčky

Offset	Pole	Formát	Typ	Popis
0	Regulovaná veličina (PV_n)	REAL	vstup	Obsahuje regulovanou veličinu, která musí být v rozsahu mezi 0,0 a 1,0.
4	Bod nastavení (SP_n)	REAL	vstup	Obsahuje požadovanou hodnotu, která musí být v rozsahu mezi 0,0 a 1,0.
8	Výstup (M_n)	REAL	vstup/ výstup	Obsahuje vypočítanou výstupní hodnotu, která musí být v rozsahu mezi 0,0 a 1,0.
12	Zesílení (K_C)	REAL	vstup	Obsahuje zesílení, které je proporcionální konstanta. Může to být kladné nebo záporné číslo.
16	Vzorkovací perioda (T_S)	REAL	vstup	Obsahuje periodu vzorkování v sekundách. Musí to být kladné číslo.
20	Integrační časová konstanta (T_I)	REAL	vstup	Obsahuje integrační časovou konstantu v minutách. Musí to být kladné číslo.
24	Derivační časová konstanta (T_D)	REAL	vstup	Obsahuje derivační časovou konstantu v minutách. Musí to být kladné číslo.
28	Integrační suma (MX)	REAL	vstup/ výstup	Obsahuje hodnotu integrační sumy v rozsahu mezi 0,0 a 1,0.
32	Předcházející regulovaná veličina (PV_{n-1})	REAL	vstup/ výstup	Obsahuje hodnotu regulované veličiny uloženou od posledního provedení instrukce PID.
36 až 79	Rezervováno pro proměnné automatického ladění. Podrobnosti viz tabulka 15-1.			

Instrukce přerušení

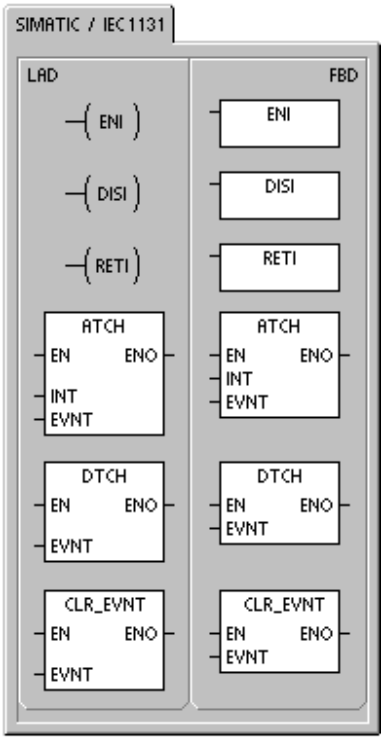
Povolení a zakázání přerušení

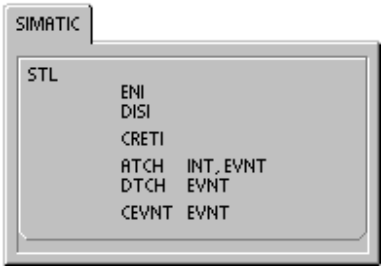
Instrukce "Povolit přerušení" (ENI) globálně povoluje zpracování všech přiřazených událostí vyvolávajících přerušení. Instrukce "Zakázat přerušení" (DISI) globálně pozastaví zpracování všech událostí vyvolávajících přerušení.

Při přechodu do režimu RUN jsou zpočátku přerušení zakázána. Když už je CPU v režimu RUN, je možné povolit zpracování přerušení provedením instrukce "Povolit přerušení". Po provedení instrukce "Zakázat přerušení" se aktivní události pro přerušení zařadí do fronty, ale podprogramy přerušení se nevyvolají.

Chyby, které nastavují ENO = 0:

- 0004 (pokus o vykonání instrukcí ENI, DISI nebo HDEF v přerušení)





Podmíněný návrat z přerušení

Instrukci "Podmíněný návrat z přerušení" (CRETI) je možné použít pro návrat z přerušení na základě předchozího stavu programu.

Připojení přerušení

Instrukce "Připojit přerušení" (ATCH) spojuje událost vyvolávající přerušení EVNT s položkou podprogramu INT a povoluje událost přerušení.

Chyby, které nastavují ENO = 0:

- 0002 (konfliktní přiřazení vstupů k HSC)

Odpojení přerušení

Instrukce "Odpojit přerušení" (DTCH) zruší přiřazení události EVNT ode všech podprogramů přerušení a zakáže tuto událost.

Vymazání události pro přerušení

Instrukce vymazání události přerušení odstraní všechny události přerušení typu EVNT z fronty přerušení. Tuto instrukci použijte pro vymazání fronty přerušení nežádoucích událostí přerušení. Jestliže se tato instrukce používá pro vymazání nežádoucích událostí přerušení, měli byste událost oddělit před vymazáním událostí z fronty. Jinak se nové události přidají do fronty po provedení instrukce vymazání událostí.

Příklad ukazuje vysokorychlostní čítač v kvadraturním režimu, který používá instrukci CLR_EVNT pro odstranění přerušení. Kdyby se senzor krokového motoru světelného stmívače zastavil v poloze, která je na hraně přechodu ze světla do tmy, pak by malé vibrace stroje mohly generovat nežádoucí přerušení před načtením nové hodnoty regulované veličiny.

Tabulka 6-45 Operandy platné pro instrukce přerušení

Vstupy/Výstupy	Typy dat	Operandy		
INT	BYTE	Konstanta (0 až 127)		
EVNT	BYTE	Konstanta	*CPU 221 a CPU 222:*	0 až 12, 19 až 23 a 27 až 33
			CPU 224:	0 až 23 a 27 až 33
			CPU 224XP a CPU 226:	0 až 33

Vysvětlení instrukcí ”Připojit přerušení” a ”Odpojit přerušení”

Dříve, než může být vyvolán podprogram přerušení, se musí vytvořit přiřazení mezi událostí a segmentem programu, který je třeba vykonat po výskytu události. Pro přiřazení události (specifikované jejím číslem) k programovému segmentu (specifikovanému číslem podprogramu přerušení) použijte instrukci ”Připojit přerušení”. Je možné přiřadit více událostí k jednomu podprogramu přerušení, ale jedna událost nemůže být současně přiřazena k více podprogramům přerušení.

Při přiřazení události k podprogramu přerušení je toto přerušení automaticky povoleno. Pokud zakážete všechna přerušení pomocí globální instrukce zakázání přerušení, zařadí se každá událost vyvolávající přerušení, která se objeví, do fronty do doby, než jsou přerušení opět povolena pomocí globální instrukce povolení přerušení nebo dokud není fronta přerušení přeplněna.

Jednotlivé události vyvolávající přerušení můžete zakázat zrušením spojení mezi přerušovací událostí a podprogramem přerušení pomocí instrukce ”Odpojit přerušení”. Tato instrukce vrací přerušení do neaktivního stavu. Tabulka 6-46 uvádí různé typy událostí.

Tabulka 6-46 Přerušovací události

Událost	Popis		CPU 221 CPU 222	CPU 224	CPU 224XP CPU 226
0	I0.0	Nábežná hrana	Y	Y	Y
1	I0.0	Sestupná hrana	Y	Y	Y
2	I0.1	Náběžná hrana	Y	Y	Y
3	I0.1	Sestupná hrana	Y	Y	Y
4	I0.2	Náběžná hrana	Y	Y	Y
5	I0.2	Sestupná hrana	Y	Y	Y
6	I0.3	Náběžná hrana	Y	Y	Y
7	I0.3	Sestupná hrana	Y	Y	Y
8	Port 0	Příjem znaku	Y	Y	Y
9	Port 0	Ukončení přenosu	Y	Y	Y
10	Časované přer. 0	SMB34	Y	Y	Y
11	Časované přer. 1	SMB35	Y	Y	Y
12	HSC0	CV=PV (aktuální hodnota = nastavená hodnota)	Y	Y	Y
13	HSC1	CV=PV (aktuální hodnota = nastavená hodnota)		Y	Y
14	HSC1	Změna směru		Y	Y
15	HSC1	Externí reset		Y	Y
16	HSC2	CV=PV (aktuální hodnota = nastavená hodnota)		Y	Y
17	HSC2	Změna směru		Y	Y
18	HSC2	Externí reset		Y	Y
19	PLS0	Přerušení od ukončení počítání pulzů PTO	Y	Y	Y
20	PLS1	Přerušení od ukončení počítání pulzů PTO	Y	Y	Y
21	Časovač T32	Přerušení CT=PT	Y	Y	Y

Tabulka 6-46 Přerušovací události, pokračování

Událost	Popis		CPU 221 CPU 222	CPU 224	CPU 224XP CPU 226
22	Časovač T96	Přerušení CT=PT	Y	Y	Y
23	Port 0	Ukončení příjmu zprávy	Y	Y	Y
24	Port 1	Ukončení příjmu zprávy			Y
25	Port 1	Příjem znaku			Y
26	Port 1	Ukončení přenosu			Y
27	HSC0	Změna směru	Y	Y	Y
28	HSC0	Externí reset	Y	Y	Y
29	HSC4	CV=PV (aktuální hodnota = nastavená hodnota)	Y	Y	Y
30	HSC4	Změna směru	Y	Y	Y
31	HSC4	Externí reset	Y	Y	Y
32	HSC3	CV=PV (aktuální hodnota = nastavená hodnota)	Y	Y	Y
33	HSC5	CV=PV (aktuální hodnota = nastavená hodnota)	Y	Y	Y

Vysvětlení, jak automat S7-200 zpracovává podprogramy přerušení

Podprogram přerušení se provádí jako odpověď na přiřazenou vnitřní (interní) nebo vnější (externí) událost. Po provedení poslední instrukce podprogramu přerušení se řízení vrátí do hlavního programu. Podprogram je možné opustit provedením instrukce podmíněného návratu z přerušení (CRETI). Tabulka 6-47 zdůrazňuje některé pokyny a omezení pro použití přerušovacích podprogramů v uživatelském programu.

Tabulka 6-47 Pokyny a omezení pro použití podprogramů přerušení

Pokyny
Zpracování přerušení poskytuje možnost rychlé reakce na speciální vnitřní nebo vnější události. Podprogramy přerušení byste měli optimalizovat pro provádění konkrétní úlohy a pak vrátit řízení do hlavního programu. Jestliže jsou podprogramy krátké a učelné, je jejich provedení rychlé a ostatní procesy nejsou dlouho zdržovány. Pokud tomu tak není, mohou nepředpokládané stavy způsobit mimořádné chování zařízení řízeného hlavním programem. Pro přerušení zcela jistě platí axiom ”čím kratší, tím lepší”.
Omezení
V podprogramu přerušení není možné použít instrukce ”Odpojit přerušení” (DISI), ”Připojit přerušení” (ENI), ”Vysokorychlostní spínač” (HDEF) a ”Konec” (END).

Systémová podpora pro přerušení

Protože mohou přerušení ovlivnit logiku kontaktů, cívek a akumulátorů, systém zálohuje a opět obnoví logický zásobník, registry akumulátoru a bity speciální paměti (SM), které indikují stav akumulátoru a instrukcí. Tím se předchází chybám v hlavním uživatelském programu způsobeným větvením do podprogramu přerušení.

Sdílení dat mezi hlavním programem a přerušením

Je možné sdílet data mezi hlavním programem a jedním nebo více přerušovacími podprogramy. Protože není možné předvídat, kdy může programovatelný automat S7-200 generovat přerušení, je žádoucí omezit počet proměnných, které jsou užívány v přerušení i v jiných místech programu. V případech, kdy je provádění instrukcí hlavního programu přerušeno událostmi vyvolávajícími přerušení, mohou nastat problémy s konzistencí sdílených dat způsobené činnostmi podprogramů přerušení. Abyste zajistili, že přerušení používá pouze dočasnou paměť a nepřepisuje data používaná jinde v programu, používejte v podprogramu přerušení tabulku lokálních proměnných.

Existuje mnoho programovacích technik, které můžete použít pro zajištění správného sdílení dat mezi hlavním programem a podprogramy přerušení. Tyto techniky buďto omezují způsob přístupu do sdílených míst paměti, nebo nepovolují přerušení sekvencí instrukcí, které používají sdílená místa paměti.

- Pro program STL, který sdílí jednu proměnnou: Pokud jsou sdílená data o velikosti jeden byte, word nebo double word a program je napsán v STL, je možné zajistit správný sdílený přístup uložením přechodných hodnot z operací se sdílenými daty pouze do nesdílených míst paměti nebo akumulátorů.
- Pro program LAD, který sdílí jednu proměnnou: Pokud jsou sdílená data o velikosti jeden byte, word nebo double word a program je napsán v LAD, pak je možné zajistit správný sdílený přístup stanovením konvence, že přístup ke sdíleným místum paměti bude prováděn pouze pomocí instrukcí "Přesun" (MOVB, MOVW, MOVD, MOVR). Zatímco mnoho instrukcí LAD se skládá z přerušitelných sekvencí instrukcí STL, instrukce "Přesun" se skládají pouze z jediné instrukce STL, jejíž provádění nemůže být ovlivněno událostmi vyvolávající přerušení.
- Pro program STL nebo LAD, který sdílí více proměnných: Pokud se sdílená data skládají z více souvisejících hodnot byte, word nebo double word, je možné pro řízení provádění podprogramu přerušení použít instrukce povolení/zakázání přerušení (DISI a ENI). V bodu hlavního programu, kde mají začít operace se sdílenými místy paměti, zakažte přerušení. Jakmile jsou dokončeny všechny činnosti, které se týkají sdílených míst, přerušení povolte. Během doby, kdy jsou přerušení zakázána, není možné provádět podprogramy přerušení, a nemohou tedy přistupovat ke sdíleným místům paměti; to však může mít za následek opožděnou odezvu na události vyvolávající přerušení.

Volání podprogramů z přerušení

V přerušení je možné volat jednu vnořenou úroveň podprogramu. Akumulátory a logický zásobník jsou sdíleny přerušením a volaným podprogramem.

Typy přerušení podporované programovatelným automatem S7-200

S7-200 podporuje následující typy přerušení:

- Přerušení od komunikačního portu: S7-200 generuje události, které umožní uživatelskému programu řídit komunikační port.
- Přerušení od vstupů/výstupů: S7-200 generuje události pro různé změny stavu různých I/O. Tyto události umožňují uživatelskému programu reagovat na vysokorychlostní čítače, pulzní výstupy či náběžné nebo sestupné hrany na vstupech.
- Cyklická přerušení: S7-200 generuje události, které umožní uživatelskému programu reagovat v přesně daných intervalech.

Přerušení od komunikačního portu

Sériový komunikační port automatu S7-200 může být řízen uživatelským programem. Tento režim práce komunikačního portu se nazývá režim Freeport. V režimu Freeport definuje uživatelský program přenosovou rychlost v baudech, počet bitů na znak, paritu a protokol. Přerušení přijetí a vysílání zprávy usnadňují komunikaci řízenou programem. Více informací najdete v odstavcích, týkajících se instrukcí vysílání a příjmu.

Přerušení od vstupů/výstupů

Přerušení I/O zahrnují přerušení od náběžné/sestupné hrany, přerušení od vysokorychlostních čítačů a přerušení od sekvence pulzů na výstupu. Automat S7-200 může generovat přerušení při náběžné nebo sestupné hraně na vstupu (I0.0, I0.1, I0.2 nebo I0.3). Událost náběžné i sestupné hrany se dá zachytit pro každý z těchto vstupních bodů. Tyto náběžné/sestupné hrany se dají použít jako označení podmínky, které se musí věnovat okamžitá pozornost při výskytu události.

Přerušení od vysokorychlostního čítače umožňují reagovat na podmínky jako "aktuální hodnota dosáhla nastavené hodnoty", "změna směru čítání", která může odpovídat změně směru otáčení hřídele, "nebo externí vynulování čítače". Každá z těchto událostí vysokorychlostního čítače umožňuje reagovat v reálném čase na vysokorychlostní události, které není možné řídit při rychlostech programového cyklu programovatelného automatu.

Přerušení od sekvence pulzů na výstupu poskytují okamžitou informaci o dokončení výstupu předepsaného počtu pulzů. Typickým použitím výstupů se sekvencí pulzů je řízení krokového motoru.

Každé z výše uvedených přerušení můžete povolit přiřazením přerušovacího podprogramu k příslušné I/O události.

Časovaná přerušení

Časovaná přerušení zahrnují cyklické přerušení a přerušení od časovače T32/T96. Na základě cyklického přerušení můžete specifikovat cyklicky prováděné činnosti. Čas cyklu je nastaven v přírůstcích po 1 ms od 1 ms do 255 ms. Pro cyklické přerušení 0 musíte zapsat čas cyklu do SMB34 a pro cyklické přerušení 1 do SMB35.

Událost cyklického přerušení předá řízení příslušnému přerušovacímu podprogramu při každém ukončení intervalu časovače. Typickým použitím časovaných přerušení je řízení vzorkování analogových vstupů nebo provádění PID algoritmu v pravidelných intervalech.

Když přiřadíte podprogram přerušení události časovaného přerušení, je povoleno časované přerušení a začíná odpočítávání času. Během přiřazování systém uchovává hodnotu času cyklu, takže následující změny SMB34 a SMB35 periodu neovlivní. Chcete–li změnit periodu, musíte změnit hodnotu času, a pak opět přiřadit podprogram přerušení k události časovaného přerušení. Když dojde k opětovnému přiřazení, funkce časovaného přerušení vymaže veškerý akumulovaný čas z předcházejícího přiřazení a odpočítávání času začne s novou hodnotou.

Pokud je povoleno, běží časované přerušení trvale a provádí přiřazený podprogram přerušení při každém ukončení specifikovaného časového intervalu. Pokud opustíte režim RUN nebo odpojíte časované přerušení, je časované přerušení zablokováno. Pokud je provedena instrukce globálního zakázání přerušení, dochází k časovaným přerušením i nadále. Každý výskyt časovaného přerušení je zařazen do fronty (do doby, než jsou přerušení povolena nebo je fronta přeplněna).

Přerušení časovače T32/T96 umožní včasnou odezvu na dokončení specifikovaného časového intervalu. Tato přerušení jsou podporována pouze pro zpožděné zapnutí (TON) a zpožděné vypnutí (TOF) časovače T32 a T96 s rozlišením 1 ms. Jinak se časovače T32 a T96 chovají standardně. Jakmile je přerušení povoleno, provede se přiřazený podprogram přerušení, pokud se aktuální hodnota aktivního časovače rovná nastavené hodnotě času během normální 1ms aktualizace časovače prováděné automatem S7-200.

Priorita přerušení a jejich řazení do fronty

S7-200 obsluhuje přerušení na základě pravidla: kdo první přijde, je jako první obsloužen, ale s přihlédnutím k prioritě. V kterémkoliv časovém bodě je vždy prováděn pouze jeden podprogram přerušení. Jakmile je provádění podprogamu přerušení jednou zahájeno, je vykonáván až do svého ukončení. Nemůže být přerušen jiným podprogramem přerušení, ani podprogramem s vyšší prioritou. Přerušení, která se objeví během zpracovávání jiného přerušení, jsou zařazena do fronty pro pozdější zpracování.

Tabulka 6-48 ukazuje tři fronty pro přerušení a maximální počty přerušení, která do nich mohou být založena.

Tabulka 6-48 Maximální počet položek ve frontě přerušení

Fronta	CPU 221, CPU 222, CPU 224	CPU 224XP a CPU 226
Komunikační fronta	4	8
Fronta přerušení od vstupů/výstupů	16	16
Fronta časovaných přerušení	8	8

Potenciálně může dojít k více přerušením, než je fronta schopna pojmout. Proto má systém paměťové bity pro přeplnění fronty (s identifikací událostí vyvolávajících přerušení, které byly ztraceny). Tabulka 6-49 ukazuje bity pro přeplnění fronty přerušení. Tyto bity byste měli používat pouze v podprogramu přerušení, protože jsou při vyprázdnění fronty vynulovány a řízení se vrací do hlavního programu.

Tabulka 6-50 uvádí všechny přerušovací události s jejich prioritou a přiřazeným číslem události.

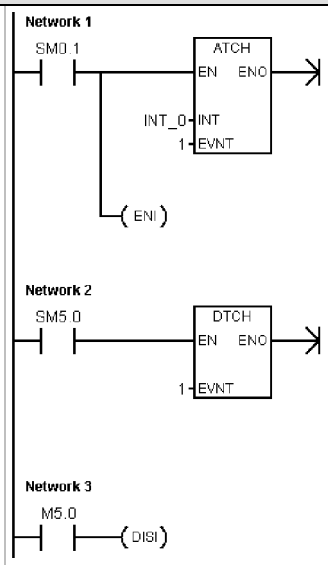
Tabulka 6-49 Bity pro přeplnění fronty přerušení

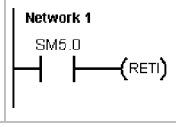
Popis (0 = bez přeplnění, 1 = přeplnění)	SM Bit
Komunikační fronta	SM4.0
Fronta přerušení od vstupů/výstupů	SM4.1
Fronta časovaných přerušení	SM4.2

Tabulka 6-50 Pořadí priority událostí vyvolávající přerušení

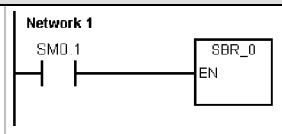
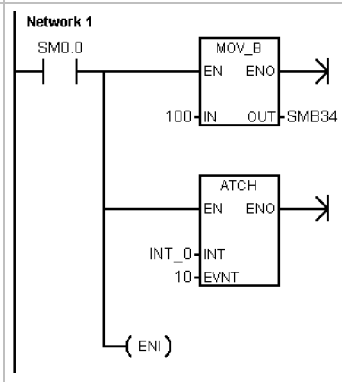
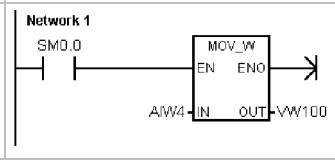
Událost	Popis		Prioritní skupina	Priorita ve skupině
8	Port 0	Příjem znaku	Komunikace *Nejvyšší priorita*	0
9	Port 0	Ukončení přenosu		0
23	Port 0	Ukončení příjmu zprávy		0
24	Port 1	Ukončení příjmu zprávy		1
25	Port 1	Příjem znaku		1
26	Port 1	Ukončení přenosu		1
19	PLS0	Přerušení od ukončení počítání pulzů PTO	Diskrétní *Střední priorita*	0
20	PLS1	Přerušení od ukončení počítání pulzů PTO		1
0	I0.0	Nábežná hrana		2
2	I0.1	Náběžná hrana		3
4	I0.2	Náběžná hrana		4
6	I0.3	Náběžná hrana		5
1	I0.0	Sestupná hrana		6
3	I0.1	Sestupná hrana		7
5	I0.2	Sestupná hrana		8
7	I0.3	Sestupná hrana		9
12	HSC0	CV=PV (aktuální hodnota = nastavená hodnota)		10
27	HSC0	Změna směru		11
28	HSC0	Externí reset		12
13	HSC1	CV=PV (aktuální hodnota = nastavená hodnota)		13
14	HSC1	Změna směru		14
15	HSC1	Externí reset		15
16	HSC2	CV=PV (aktuální hodnota = nastavená hodnota)		16
17	HSC2	Změna směru		17
18	HSC2	Externí reset		18
32	HSC3	CV=PV (aktuální hodnota = nastavená hodnota)		19
29	HSC4	CV=PV (aktuální hodnota = nastavená hodnota)		20
30	HSC4	Změna směru		21
31	HSC4	Externí reset		22
33	HSC5	CV=PV (aktuální hodnota = nastavená hodnota)		23
10	Časované přer. 0	SMB34	Časované *Nejnižší priorita*	0
11	Časované přer. 1	SMB35		1
21	Časovač T32	přerušení CT=PT		2
22	Časovač T96	přerušení CT=PT		3

Příklad: Instrukce přerušení

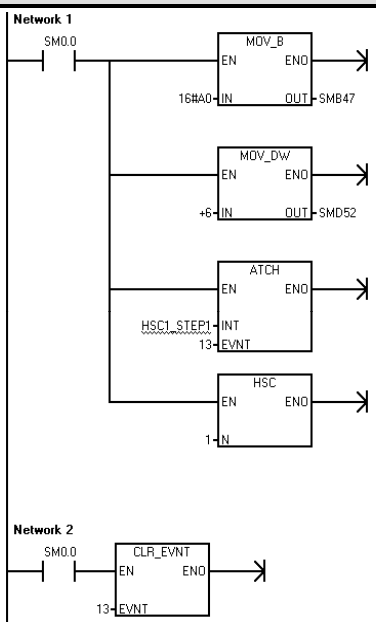
MAIN		Network 1 //V prvním programovém cyklu:
		//1. Definuje přerušovací podprogram INT_0
		// jako přerušení od sestupné hrany pro I0.0.
		//2. Globálně povolí přerušení.
		LD SM0.1
		ATCH INT_0, 1
		ENI
		Network 2 //Pokud se detekuje chyba I/O,
		//zakáže se přerušení od sestupné hrany pro I0.0.
		//Tento network je volitelný.
		LD SM5.0
		DTCH 1
		Network 3 //Když je M5.0 zapnutý,
		//zakáže všechna přerušení.
		LD M5.0
		DISI

INT 0		Network 1 //Podprogram přerušení: //Podmíněný návrat na základě chyby I/O. LD SM5.0 CRETI

Příklad: Časované přerušení pro čtení hodnoty analogového vstupu

MAIN		Network 1 //V prvním programovém cyklu volá podprogram 0. LD SM0.1 CALL SBR_0
SBR 0		Network 1 //1. Nastaví interval pro časované přerušení 0 na hodnotu // 100 ms. //2. Přiřadí časované přerušení 0 (událost 10) k INT_0. //3. Globální povolení přerušení. LD SM0.0 MOVB 100, SMB34 ATCH INT_0, 10 ENI
INT 0		Network 1 //Snímá hodnotu AIW4 každých 100 ms LD SM0.0 MOVW AIW4, VW100

Příklad: Instrukce vymazání události pro přerušení




```
Network 1      // Průvodce pro instrukci HSC

LD      SM0.0
MOVB    16#F8, SMB47
                //Nastaví řídicí bity:
                //zapíše nastavenou hodnotu;

MOVD    +6, SMD52
                //PV = 6;

ATCH    HSC1_STEP1, 13
                //Přerušení HSC1_STEP1: CV = PV pro HC1

Network 2      //Vymaže nežádoucí přerušení způsobená
                //vibracemi stroje.

LD              SM0.0
CEVNT           13
```

Instrukce logické operace

Inverze

Inverze byte, word a double word

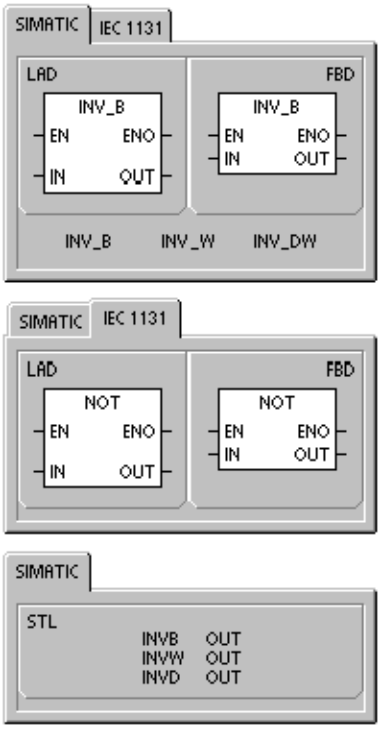
Instrukce ”Invertovat byte” (INVB), ”Invertovat word” (INVW) a ”Invert double word” (INVD) vytvoří jedničkový doplněk vstupu IN a výsledek uloží do místa paměti OUT.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)

Ovlivněné bity SM:

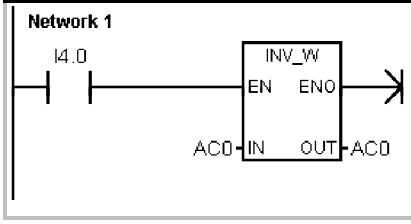
- SM1.0 (nula)



Tabulka 6-51 Operandy platné pro instrukce inverze

Vstupy/Výstupy	Typy dat	Operandy
IN	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
	DWORD	ID, QD, VD, MD, SMD, SD, LD, AC, HC, *VD, *LD, *AC, konstanta
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC,*VD, *LD, *AC
	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, *VD, *LD, *AC
	DWORD	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

Příklad: Instrukce inverze




```
Network 1
LD      I4.0
INVW    AC0
```

Inverze word

AC0 1101 0111 1001 0101

doplněk

AC0 0010 1000 0110 1010

AND, OR a 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.

Bity SM a ENO

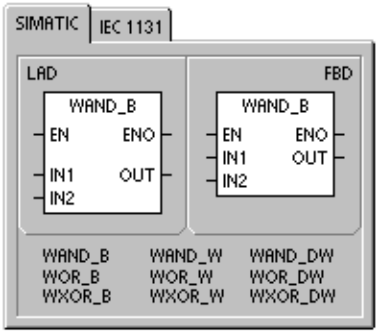
Bity SM a ENO ovlivňují u všech instrukcí popsaných na této stránce následující podmínky.

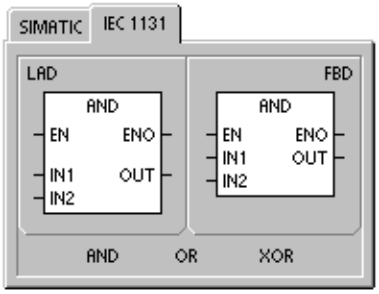
Chyby, které nastavují ENO = 0

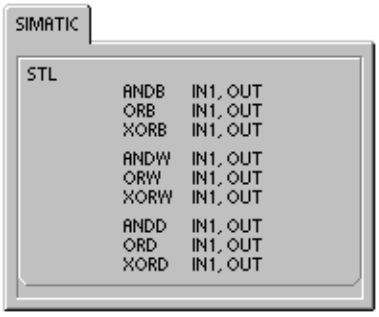
- 0006 (nepřímá adresa)

Ovlivněné bity SM:

- SM1.0 (nula)



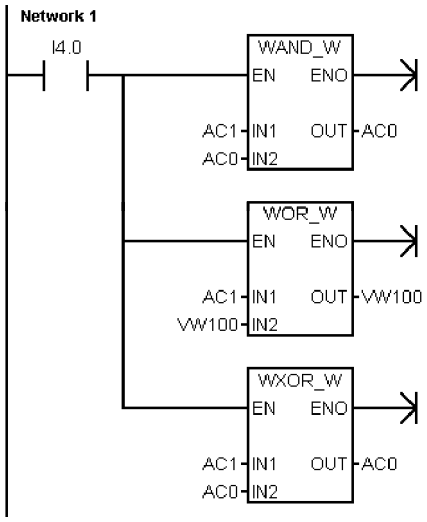




Tabulka 6-52 Operandy platné pro instrukce AND, OR a XOR

Vstupy/Výstupy	Typy dat	Operandy
IN1, IN2	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
	DWORD	ID, QD, VD, MD, SMD, SD, LD, AC, HC, *VD, *LD, *AC, konstanta
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD
	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, *VD, *AC, *LD
	DWORD	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD

Příklad: Instrukce AND, OR a XOR




```
Network 1
LD      I4.0
ANDW    AC1, AC0
ORW     AC1, VW100
XORW    AC1, AC0
```

AND Word

AC1 0001 1111 0110 1101
AND
AC0 1101 0011 1110 0110
rovná se
AC0 0001 0011 0110 0100

OR Word

AC1 0001 1111 0110 1101
OR
VW100 1101 0011 1010 0000
rovná se
VW100 1101 1111 1110 1101

XOR Word

AC1 0001 1111 0110 1101
XOR
AC0 0001 0011 0110 0100
rovná se
AC0 0000 1100 0000 1001

Instrukce přesunu

Přesun byte, word, double word nebo real

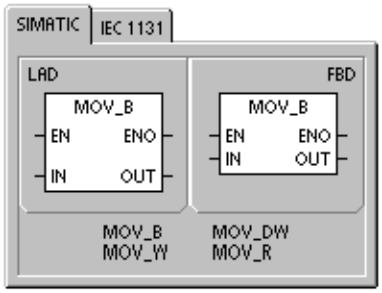
Instrukce ”Přesunout byte” (MOVB), ”Přesunout word” (MOVW), ”Přesunout double word” (MOVD) a ”Přesunout real” (MOVR) přesunou hodnotu z paměťového místa IN na nové paměťové místo OUT, aniž by změnily původní hodnotu.

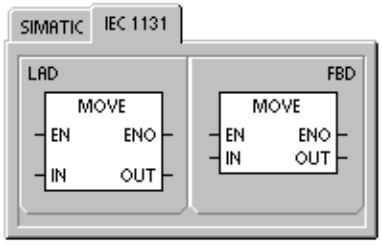
Pro vytvoření ukazatele použijte instrukci ”Přesunout double word”. Více informací naleznete v odstavcích o ukazatelích a nepřímé adresaci v kapitole 4 4.

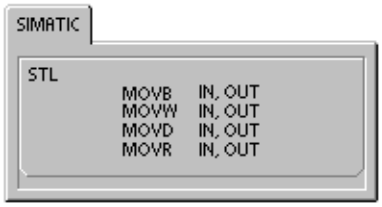
V instrukci IEC pro přesun se mohou typy vstupních a výstupních dat mohou lišit, ale musí mít stejnou velikost.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)



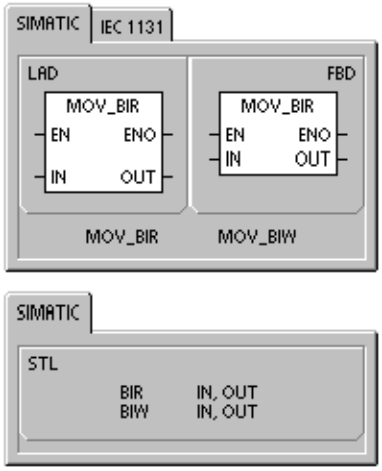




Tabulka 6-53 Operandy platné pro instrukce přesunu

Vstupy/Výstupy	Typy dat	Operandy
IN	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
	WORD, INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *AC, *LD, konstanta
	DWORD, DINT	ID, QD, VD, MD, SMD, SD, LD, HC, &VB, &IB, &QB, &MB, &SB, &T, &C, &SMB, &AIW, &AQW, AC, *VD, *LD, *AC, konstanta,
	REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC, konstanta
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC
	WORD, INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AQW, *VD, *LD, *AC
	DWORD, DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

Okamžitý přenos bytu (čtení a zápis)



Instrukce ”Přesunout byte okamžitě” umožní okamžitý přesun bytu mezi fyzickým vstupem/výstupem a paměťovým místem.

Instrukce ”Přesunout byte na čtení okamžitě” (BIR) snímá fyzický vstup (IN) a zapisuje výsledek do adresy paměti (OUT), ale registr obrazu procesu není aktualizován.

Instrukce ”Přesunout byte na zápis okamžite” (BIW) snímá data z adresy paměti (IN) a zapisuje je na fyzický výstup (OUT) a odpovídající místo obrazu procesu.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- Nemožnost přístupu do rozšiřovacího modulu

Tabulka 6-54 Operandy platné pro instrukci ”Přesunout byte na čtení okamžitě”

Vstupy/Výstupy	Typy dat	Operandy
IN	BYTE	IB, *VD, *LD, *AC
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC

Tabulka 6-55 Operandy platné pro instrukci ”Přesunout byte na zápis okamžitě”

Vstupy/Výstupy	Typy dat	Operandy
IN	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
OUT	BYTE	QB, *VD, *LD, *AC

Přesun bloku

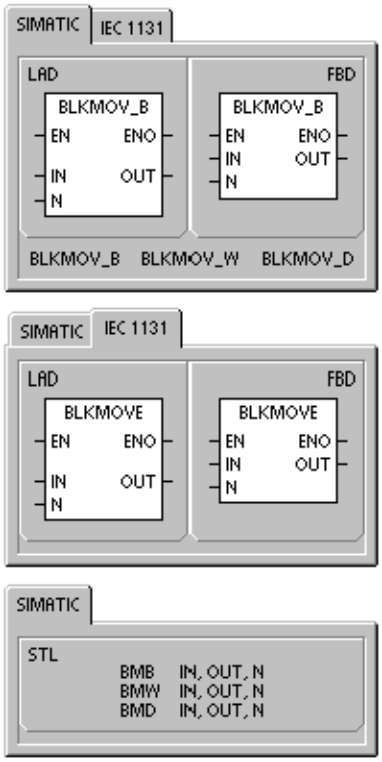
Přesun bloku byte, word nebo double Word

Instrukce ”Přesunout blok byte” (BMB), ”Přesunout blok word” (BMW) a ”Přesunout blok double word” (BMD) přesunou daný objem dat na nové paměťové místo přesunutím zadaného počtu bytů, word nebo double word v parametru N, který začíná na vstupní adrese IN, do nového bloku začínajícího na výstupní adrese OUT.

N má rozsah od 1 do 255.

Chyby, které nastavují ENO = 0

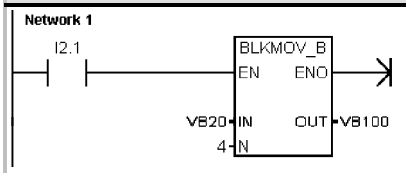
- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)



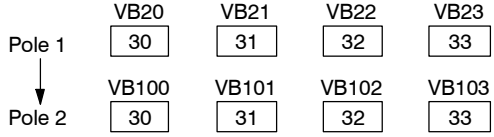
Tabulka 6-56 Operandy platné pro instrukce přesunu bloku dat

Vstupy/Výstupy	Typy dat	Operandy
IN	BYTE	IB, QB, VB, MB, SMB, SB, LB, *VD, *LD, *AC
	WORD, INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AIW, *VD, *LD, *AC
	DWORD, DINT	ID, QD, VD, MD, SMD, SD, LD, *VD, *LD, *AC
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, *VD, *LD, *AC
	WORD, INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AQW, *VD, *LD, *AC
	DWORD, DINT	ID, QD, VD, MD, SMD, SD, LD, *VD, *LD, *AC
N	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, konstanta,*VD, *LD, *AC

Příklad: Instrukce přesunu bloku dat



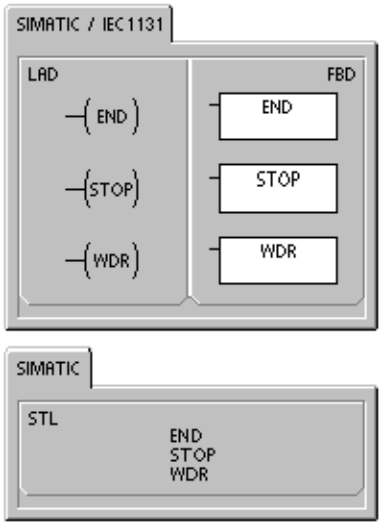

```
Network 1       //Přesune pole 1 (VB20 až VB23)
                //na pole 2 (VB100 až VB103)
LD      I2.1
BMB     VB20, VB100, 4
```



Řízení programu

Podmíněný konec programu

Instrukce ”Podmíněný konec programu” (END) ukončí současný programový cyklus na základě předcházející logiky. Instrukci ”Podmíněný konec programu” je možné použít v hlavním programu; není možné ji však použít v podprogramech nebo v přerušení.



Stop

Instrukce ”Stop” (STOP) ukončí provádění uživatelského programu tím, že způsobí přechod CPU S7-200 z režimu RUN do režimu STOP.

Pokud se instrukce ”Stop” provede v podprogramu přerušení, je tento podprogram okamžitě ukončen a všechna nevyřízená přerušení jsou ignorována. Zbývající činnosti současného programového cyklu jsou dokončeny včetně provedení hlavního uživatelského programu a přechod z režimu RUN do režimu STOP je proveden na konci aktuálního programového cyklu.

Reset Watchdogu

Instrukce ”Reset Watchdogu” (WDR) umožňuje opětovné spuštění obvodu sledování funkce (watchdog timer) CPU S7-200, čímž se prodlouží čas, po který může trvat programový cyklus, aniž by došlo k chybě watchdogu.

Instrukci ”Reset Watchdogu” je nutné používat opatrně. Pokud použijete instrukce k zabránění dokončení programového cyklu nebo pro výrazné prodloužení jeho trvání, jsou následující procesy zablokovány až do dokončení cyklu:

- Komunikace (s výjimkou režimu Freeport)
- Aktualizace vstupů/výstupů (s výjimkou bezprostředních I/O)
- Aktualizace force (vnucení hodnot)
- Aktualizace SM bitu (SM0, SM5 až SM29 nejsou aktualizovány)
- Diagnostika v době vykonávání programu (run-time)
- 10ms a 100ms časovače nebudou správně akumulovat čas u programových cyklů delších než 25 sekund
- Instrukce STOP použitá v podprogramu přerušení
- Rozšiřovací moduly s diskrétními výstupy mají také obvod sledování funkce (watchdog timer), který vypne výstupy, pokud automat S7-200 nezapisuje do modulu. V případě diskrétních výstupů použijte bezprostřední zapisování do každého rozšiřovacího modulu, aby se zachovaly zapnuté správné výstupy během prodlouženého trvání programového cyklu. Viz příklad, který následuje po tomto popisu.



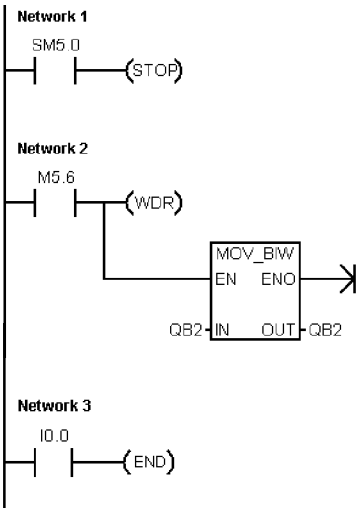
Tip

Pokud předpokládáte, že programový cyklus bude delší než 500 ms, nebo pokud očekáváte shluk přerušovacích událostí, které by mohly zabránit návratu do hlavního cyklu po dobu delší než 500 ms, měli byste použít instrukci ”Reset Watchdogu” pro opětovné spuštění obvodu sledování funkce.

Když použijete instrukci ”Reset Watchdogu”, měli byste v každém diskrétním rozšiřovacím modulu pro resetování watchdogu každého rozšiřovacího modulu pokaždé provést i bezprostřední zápis do jednoho výstupního bytu (QB).

Pokud použijete instrukci ”Reset Watchdogu”, abyste umožnili provádění programu, který vyžaduje dlouhý čas programového cyklu, přepnutí přepínače režimů do polohy STOP způsobí, že S7-200 přejde do režimu STOP do 1,4 sekundy.

Příklad: Instrukce ”Stop”, ”Konec” a ”Reset Watchdogu”

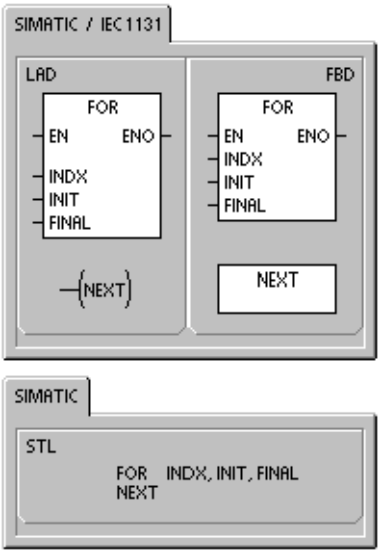



```
Network 1       //Když se detekuje chyba I/O:
                //vnutí se přechod do režimu STOP.
LD      SM5.0
STOP

Network 2       //Když je M5.6 v hodnotě logická 1, umožní, aby
                //byl programový cyklus prodloužen:
                //1.  Znovu spustí instrukci ”Reset Watchdog” pro
                    S7-200.
                //2. Znovu spustí watchdog pro
                    první výstupní modul.
LD      M5.6
WDR
BIW     QB2, QB2

Network 3       //Když je I0.0 v hodnotě logická 1, ukončí aktuální
                //programový cyklus.
LD      I0.0
END
```

Smyčka For–Next



Smyčky ”For” (FOR) a ”Next” (NEXT) se používají na definování smyčky, která má specifický počet opakování. Každá instrukce ”For” vyžaduje instrukci ”Next”. Smyčky For–Next je možné vnořit (umístit smyčku For–Next do jiné smyčky For–Next) až do hloubky osm.

Instrukce ”For” provede instrukce mezi instrukcemi ”For” a ”Next”. Je třeba specifikovat hodnotu indexu neboli počet opakování pro aktuální smyčku INDX, počáteční hodnotu INIT a konečnou hodnotu FINAL.

Instrukce ”Next” označuje konec smyčky FOR.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)

Pokud provedete smyčku For–Next, její práce pokračuje až do ukončení opakování, pokud nezměníte konečnou hodnotu zvnitřku smyčky samotné. Hodnoty je možné měnit i během práce smyčky For–Next. Pokud je smyčka spuštěna opakovaně, zkopíruje se počáteční hodnota do hodnoty indexu (číslo aktuální smyčky).

Instrukce ”For–Next” se sama vynuluje při příštím povolení.

Jestliže má například INIT hodnotu 1 a FINAL hodnotu 10, budou instrukce mezi instrukcí ”For” a instrukcí ”Next” provedeny desetkrát a hodnota INDX naroste:
1, 2, 3, ...10.

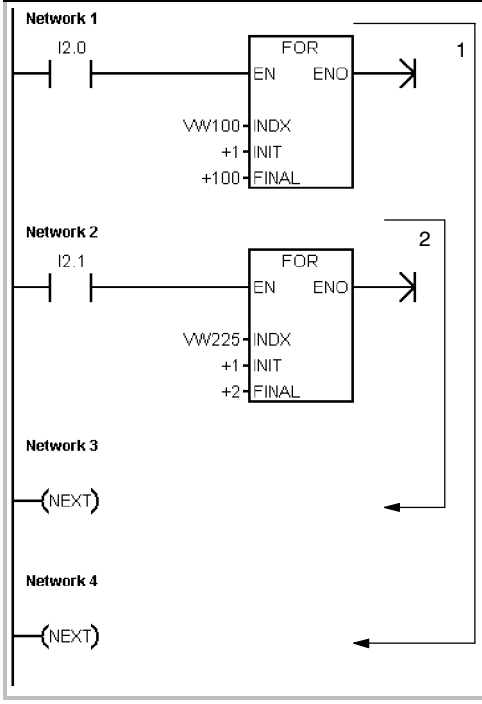
Pokud je počáteční hodnota větší než konečná, smyčka se neprovede. Po každém provedení instrukcí mezi instrukcemi ”For” a ”Next” je hodnota INDX inkrementována a výsledek je porovnán s konečnou hodnotou. Pokud je INDX větší než konečná hodnota, smyčka je ukončena.

Pokud je na vrcholu zásobníku hodnota log. 1, když program vstoupí do smyčky For–Next, bude hodnota na vrcholu zásobníku 1, když program ze smyčky For–Next vystoupí.

Tabulka 6-57 Operandy platné pro instrukce ”For” a ”Next”

Vstupy/Výstupy	Typy dat	Operandy
INDX	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, *VD, *LD, *AC
INIT, FINAL	INT	VW, IW, QW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta

Příklad: Instrukce smyčky For – Next




```
Network 1       //Když se zapne I2.0, je vnější smyčka
                //(šipka 1) provedena stokrát.
LD      I2.0
FOR     VW100, +1, +100

Network 2       //Vnitřní smyčka (šipka 2)
                //je provedena dvakrát pro každé
                //provedení vnější smyčky,
                //pokud je ovšem I2.1 zapnutý.
LD      I2.1
FOR     VW225, +1, +2

Network 3       //Konec smyčky 2.
NEXT

Network 4       //Konec smyčky 1.
NEXT
```

Skokové instrukce

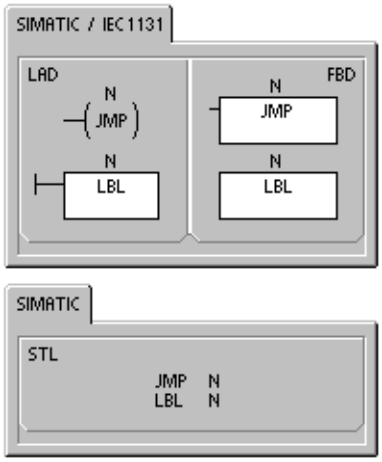
Instrukce ”Skok na návěští” uskutečňuje skok na specifikované návěští N programu.

Instrukce Návěští (LBL) označuje cílové místo pro skok N.

Skokové instrukce je možné používat v hlavním programu, v podprogramech nebo v přerušení. Instrukce ”Skok” a k ní příslušející instrukce ”Návěští” musí být vždy umístěny ve stejné části programu (v hlavním programu, podprogramu nebo přerušení).

Není možné skočit z hlavního programu na návěští v podprogramu nebo přerušení. Obdobně nemůžete skočit z podprogramu nebo z přerušení na návěští mimo tento podprogram nebo v přerušení.

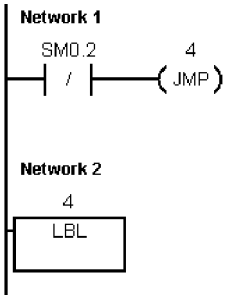
Instrukci ”Skok” můžete použít v segmentu SCR, ale příslušná instrukce ”Návěští” musí být umístěna ve stejném segmentu SCR.



Tabulka 6-58 Operandy platné pro skokové instrukce

Vstupy/Výstupy	Typy dat	Operandy
N	WORD	Konstanta (0 až 255)

Příklad: Instrukce Skok




```
Network 1      //Pokud nebyla ztracena remanentní data,
               //přejít na LBL4.
LDN    SM0.2
JMP    4
Network 2
LBL    4
```

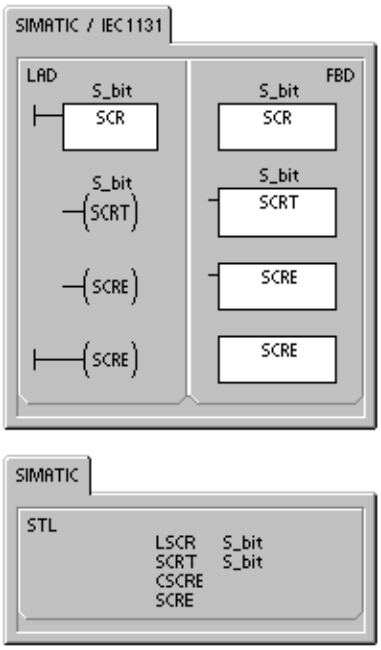
Sekvenční řídicí relé (SCR)

Instrukce SCR poskytují jednoduchou, ale účinnou programovací techniku stavového řízení, která se přirozeně hodí pro LAD, FBD nebo STL.

Kdykoliv se aplikace skládá ze sledu operací, které musí být prováděny opakovaně, je možné pomocí SCR strukturalizovat program tak, aby odpovídal přímo dané aplikaci. V důsledku toho je možné aplikaci rychleji a snáze programovat a odladit.

Instrukce "Načíst SCR" (LSCR) uloží do SCR a logických zásobníku hodnotu bitu S daného instrukcí N.

Segment SCR je aktivován nebo deaktivován výslednou hodnotou zásobníku SCR. Hodnota zásobníku SCR je zkopírována na vrchol logického zásobníku, takže bloky nebo výstupní kontakty mohou být připojeny přímo k levé napájecí liště bez vloženého kontaktu.



Omezení

Při používání SCR mějte na paměti následující omezení:

- Stejný bit S není možné použít ve více než jedné části podprogramu. Jestliže použijete například S0.1 v hlavním programu, nepoužívejte ho v podprogramu.
- Není možné provést skok do segmentu SCR nebo z něho ven; můžete ale použít instrukce "Skok" a "Návěští" pro skok kolem segmentů SCR nebo skok uvnitř segmentu SCR.
- V segmentu SCR nemůžete použít instrukci END.

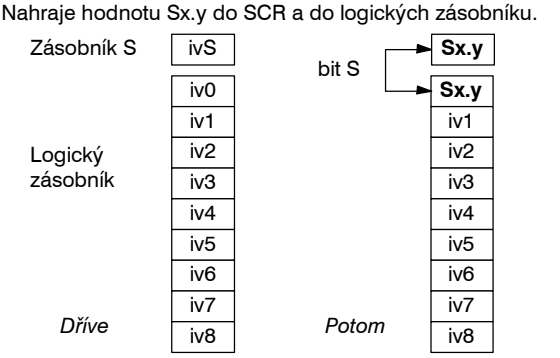
Tabulka 6-59 Operandy platné pro instrukce sekvenčních řídicích relé

Vstupy/Výstupy	Typy dat	Operandy
S_bit	BOOL	S

Obrázek 6-32 ukazuje zásobník S, logický zásobník a funkci instrukce ”Načíst SCR”. Pro instrukce sekvenčního řídicího relé platí:

- Instrukce ”Načíst SCR” (LSCR) označuje začátek segmentu SCR a instrukce ”Konec SCR” (SCRE) označuje konec segmentu SCR. Provedení veškeré logiky mezi instrukcemi ”Load SCR” a ”SCR End” závisí na hodnotě zásobníku S. Logika mezi ”Konec SCR ” a příští instrukcí ”Načíst SCR” na hodnotě zásobníku S nezávisí.
- Instrukce ”Přechod SCR” (SCRT) dává možnost přenést řízení z aktivního segmentu SCR na jiný segment SCR.

  Provedení instrukce ”Přechod SCR” při aktivním signálovém toku vynuluje bit S právě aktivního segmentu a nastaví bit S daného segmentu. Vynulování bitu S aktivního segmentu neovlivní zásobník S v době provedení instrukce ”Přechod SCR”. V důsledku toho zůstane segment SCR aktivní, dokud není opuštěn.

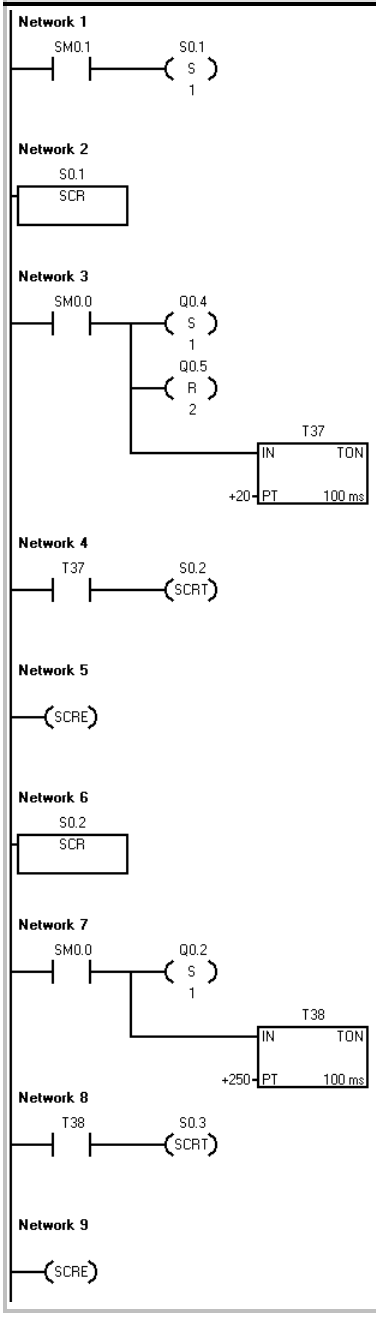


Obr. 6-32 Účinek LSCR na logický zásobník

- Instrukce ”Podmíněný konec SCR” (CSCRE) poskytuje možnost vystoupit z aktivního segmentu SCR bez provedení instrukcí mezi instrukcemi ”Podmíněný konec SCR” a ”Konec SCR”. Instrukce ”Podmíněný konec SCR” neovlivní žádný bit S ani zásobník S.

V následujícím příkladu bit prvního programového cyklu SM0.1 nastaví S0.1, což je aktivní Stav 1 při prvním programovém cyklu. Po zpoždění 2 sekund způsobí T37 přechod do Stavu 2. Tento přechod deaktivuje segment SCR (S0.1) Stavu 1 a aktivuje segment SCR (S0.2) Stavu 2.

Příklad: Instrukce sekvenčního řídicího relé




```
Network 1       //V prvním programovém cyklu povolí Stav 1.
LD      SM0.1
S       S0.1, 1

Network 2       //Začátek oblasti řízení Stavu 1.
LSCR    S0.1

Network 3       //Řídí signály pro ulici 1:
                //1. Nastavení: Rozsvítí červené světlo.
                //2. Nulování: Zhasne žluté a zelené světlo.
                //3. Spustí 2sekundový časovač.
LD      SM0.0
S       Q0.4, 1
R       Q0.5, 2
TON     T37, +20

Network 4       //Po zpoždení 2 sekundy přechod do Stavu 2.
LD      T37
SCRT    S0.2

Network 5       //Konec oblasti SCR pro Stav 1.
SCRE

Network 6       //Začátek oblasti řízení Stavu 2.
LSCR    S0.2

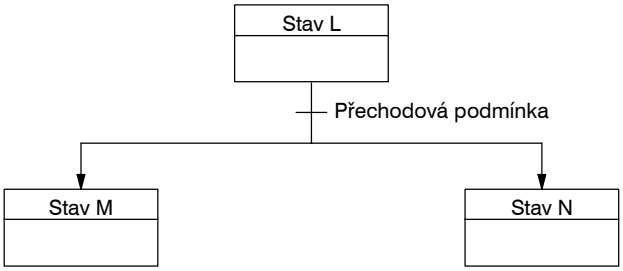
Network 7       //Řídí signály pro ulici 2:
                //1. Nastavení: Rozsvítí zelené světlo.
                //2. Spustí 25sekundový časovač.
LD      SM0.0
S       Q0.2, 1
TON     T38, +250

Network 8       //Po zpoždení 25 sekund přechod do Stavu 3.
LD      T38
SCRT    S0.3

Network 9       //Konec oblasti SCR pro Stav 2.
SCRE
```

Větvení programu

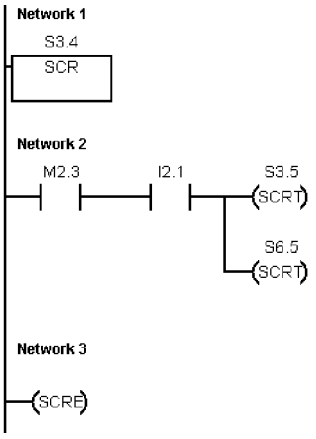
V mnoha aplikacích musí být jeden tok sekvenčních stavů rozdělen do dvou nebo více různých toků. Jestliže se tok řízení rozvětví, musí být všechny výstupní větve aktivovány současně. To ukazuje obrázek 6-33.



Obr. 6-33 Větvení toku řízení

Větvení toků řízení může být v programu SCR implementováno použitím více instrukcí SCRT povolených stejnou přechodovou podmínkou, jak ukazuje následující příklad.

Příklad: Divergence toků řízení




```
Network 1       //Začátek oblasti řízení Stavu L
LSCR    S3.4

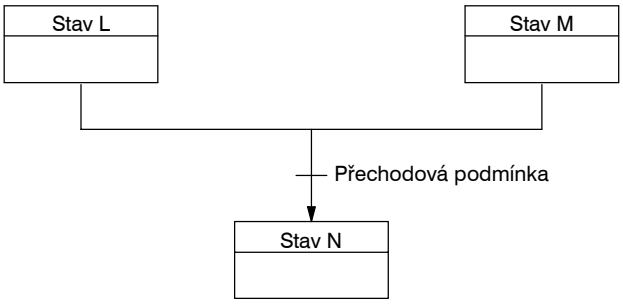
Network 2
LD      M2.3
A       I2.1
SCRT    S3.5            //Přechod do Stavu M
SCRT    S6.5            //Přechod do Stavu N

Network 3       //Konec oblasti stavu pro Stav L
SCRE
```

Sloučení větví

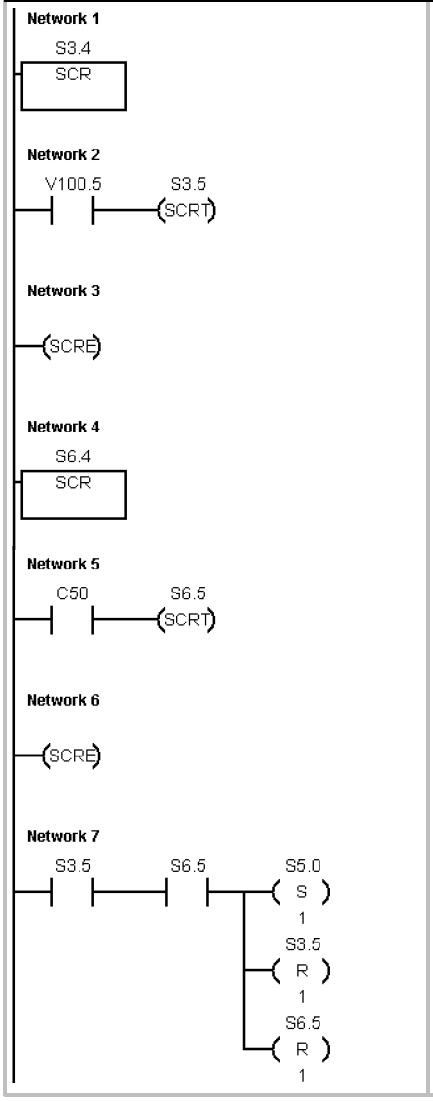
Obdobná situace nastane, když se dva nebo více toků sekvenčních stavů musí sloučit do toku jediného. Sloučení více toků do jediného se nazývá konvergence. Když toky konvergují, musí být před provedením následujícího stavu všechny vstupující toky ukončené. Obrázek 6-34 znázorňuje konvergenci dvou toků řízení.

Konvergenci toků řízení je možné v programu SCR implementovat provedením přechodu ze stavu L do stavu L' a přechodem ze stavu M do stavu M'. Když jsou oba bity SCR představující L' a M' pravdivé, je možné povolit stav N, jak ukazuje následující příklad.



Obr. 6-34 Konvergence toku řízení

Příklad: Konvergence toků řízení




```
Network 1       //Začátek oblasti řízení Stavu L
LSCR    S3.4

Network 2       //Přechod do stavu L'
LD      V100.5
SCRT    S3.5

Network 3       //Konec oblasti SCR pro Stav L
SCRE

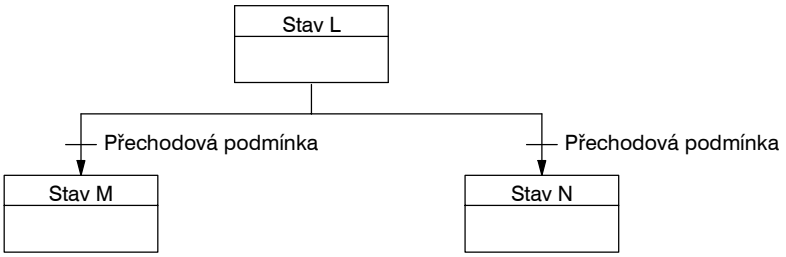
Network 4       //Začátek oblasti řízení Stavu M
LSCR    S6.4

Network 5       //Přechod do stavu M'
LD      C50
SCRT    S6.5

Network 6       //Konec oblasti SCR pro Stav M
SCRE

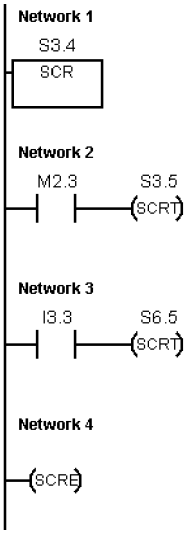
Network 7       //Když je stav L' i stav M' aktivován:
                //1. Povolí Stav N (S5.0)
                //2. Resetuje Stav L' (S3.5)
                //3. Resetuje Stav M' (S6.5)
LD      S3.5
A       S6.5
S       S5.0, 1
R       S3.5, 1
R       S6.5, 1
```

V ostatních situacích je možné tok řízení nasměrovat do jenoho z několika možných toků řízení podle toho, která přechodová podmínka se uskuteční jako první. Taková situace je ukázána na obrázku 6-35, který ukazuje shodný program SCR.



Obr. 6-35 Větvení toku řízení v závislosti na přechodové podmínce

Příklad: Podmíněné přechody




```
Network 1       //Začátek oblasti řízení Stavu L
LSCR    S3.4

Network 2       //Přechod do Stavu M
LD      M2.3
SCRT    S3.5

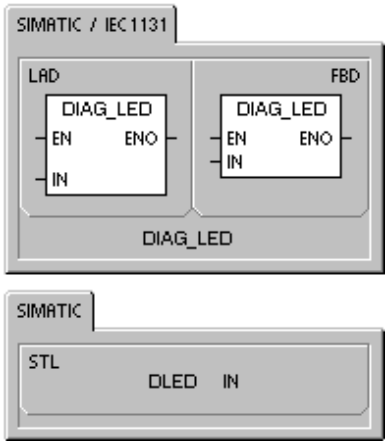
Network 3       //Přechod do Stavu N
LD      I3.3
SCRT    S6.5

Network 4       //Konec oblasti SCR pro Stav L
SCRE
```

Instrukce diagnostických kontrolek

Jestliže má vstupní parametr IN hodnotu nula, pak nastaví diagnostickou kontrolku OFF. Jestliže má vstupní parametr IN hodnotu větší než nula, pak nastaví diagnostickou kontrolku ON (žlutá).

Světelnou emisní diodu (LED) CPU označenou SF/ DIAG lze nakonfigurovat, aby svítila žlutě, když jsou podmínky specifikované v systémovém bloku pravdivé nebo když je instrukce DIAG_LED provedena s nenulovým parametrem IN.



Možnosti zaškrtávacích políček systémového bloku (konfigurace LED):

- ❏ LED SF/ DIAG svítí (žlutě), když je položka vnucena v CPU
- ❏ LED SF/ DIAG svítí (žlutě), když má modul chybu vstupu/výstupu

Zrušte zaškrtnutí obou možností konfigurace LED, aby instrukce DIAG_LED převzala výlučnou kontrolu nad žlutým světlem SF/ DIAG. Systémová chyba (SF) CPU je indikována červeným světlem.

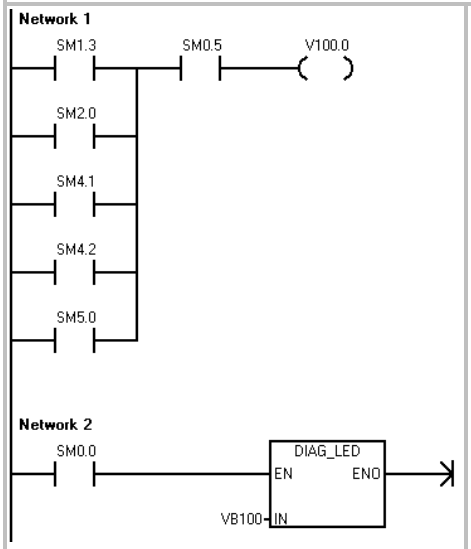
Tabulka 6-60 Operandy platné pro instrukci diagnostické LED

Vstupy/Výstupy	Typy dat	Operandy
IN	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, konstanta, *VD, *LD, *AC

Příklad 1 Instrukce diagnostické LED

Blikání diagnostické LED při zjištění chyby.

Blikání diagnostické LED při zjištění některé z 5 chybových podmínek.



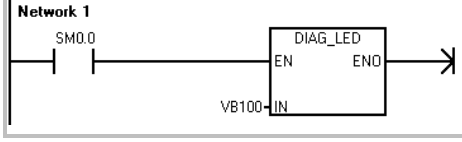

```
Network 1
LD      SM1.3
O       SM2.0
O       SM4.1
O       SM4.2
O       SM5.0
A       SM0.5
=       V100.0

Network 2
LD      SM0.0
DLED    VB100
```

Příklad 2 Instrukce diagnostické LED

Rozsvícení diagnostické LED při návratu chyby.

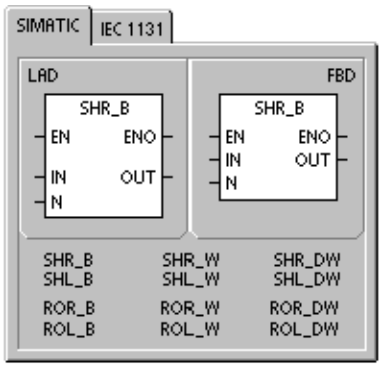
Když je hlášen chybový kód ve VB100, rozsvítit diagnostickou LED.




```
Network 1
LD      SM0.0
DLED    VB100
```

Instrukce posunu a rotace

Posun vpravo a vlevo



Instrukce posunu posunou vstupní hodnotu IN doprava nebo doleva o počet posunu N a výsledek nahrajou na výstup OUT.

Při posunu bitu ven doplňují instrukce posunu hodnotu nulami. Pokud je počet posunu (N) větší nebo pokud se rovná povolenému maximu (8 pro operace s byty, 16 pro operace s word a 32 pro operace s double word), je hodnota posunuta tolikrát, kolik je maximální počet pro tuto operaci. Pokud je počet posunu větší než 0, přebírá bit přetečení paměti (SM1.1) hodnotu posledního vysunutého bitu. Nulový bit paměti (SM1.0) je nastaven, jestliže je výsledkem operace posunu nula.

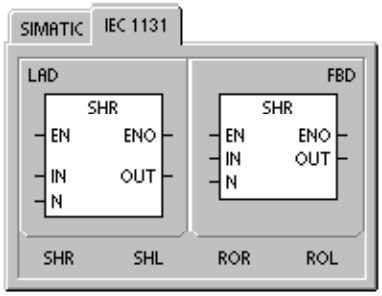
Operace s byty jsou bez znaménka. U operací s jednotkami word a double word je při použití datových typu se znaménky posunut i znaménkový bit.

Chyby, které nastavují ENO = 0

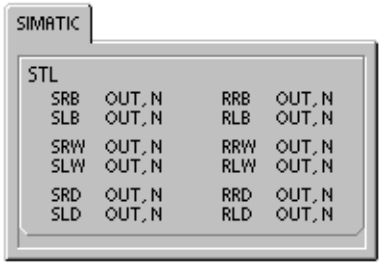
- 0006 (nepřímá adresa)

Ovlivněné bity SM:

- SM1.0 (nula)
- SM1.1 (přetečení)



Rotace vpravo a vlevo



Instrukce rotují vstupní hodnotu (IN) doprava nebo doleva o počet bitů (N) a výsledek uloží na paměťové místo (OUT). Rotace je kruhová.

Pokud je počet rotací větší nebo pokud se rovná maximu pro operaci (8 pro operaci s byty, 16 pro operaci s word nebo 32 pro operaci s double word), provede S7-200 operaci modulo s počtem posunu, aby získal před provedením rotace platný počet. Výsledkem je počet posunu 0 až 7 pro operaci s byty, 0 až 15 pro operaci s jednotkami word a 0 až 31 pro operaci s jednotkami double word.

Pokud je počet posunu 0, operace rotace není provedena. Po provedení operace rotace je hodnota posledního rotovaného bitu zkopírována do bitu přeplnění (SM1.1).

Pokud není počet posunu celočíselným násobkem 8 (pro operace s byty), 16 (pro operace s jednotkami word) nebo 32 (pro operace s jednotkami double word), zkopíruje se poslední rotovaný bit do paměťového bitu přeplnění (SM1.1). Nulový bit paměti (SM1.0) je nastaven, jestliže je hodnota otočena na nulu.

Operace s byty jsou bez znaménka. U operací s jednotkami word a double word je při použití datových typu se znaménky posunut i znaménkový bit.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)

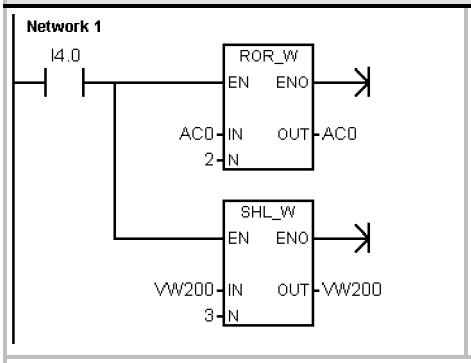
Ovlivněné bity SM:

- SM1.0 (nula)
- SM1.1 (přetečení)

Tabulka 6-61 Operandy platné pro instrukce posunu a rotace

Vstupy/Výstupy	Typy dat	Operandy
IN	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
	DWORD	ID, QD, VD, MD, SMD, SD, LD, AC, HC, *VD, *LD, *AC, konstanta
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC
	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, *VD, *LD, *AC
	DWORD	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *LD, *AC
N	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta

Příklad: Instrukce posunu a rotace



Network 1

```
LD      I4.0
RRW     AC0, 2
SLW     VW200, 3
```

Rotace

	Před rotací	Přetečení
AC0	0100 0000 0000 0001	x
	Po první rotaci	Přetečení
AC0	1010 0000 0000 0000	1
	Po druhé rotaci	Přetečení
AC0	0101 0000 0000 0000	0

Nulový bit paměti (SM1.0) = 0
Bit Přetečení paměti (SM1.1) = 0

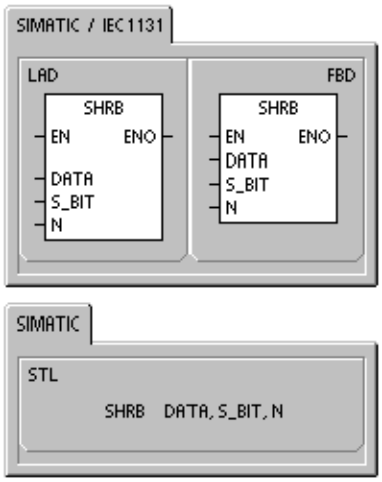
Posun

	Před posunem	Přetečení
VW200	1110 0010 1010 1101	x
	Po prvním posunu	Přetečení
VW200	1100 0101 0101 1010	1
	Po druhém posunu	Přetečení
VW200	1000 1010 1011 0100	1
	Po třetím posunu	Přetečení
VW200	0001 0101 0110 1000	1

Nulový bit paměti (SM1.0) = 0
Bit Přetečení paměti (SM1.1) = 1

Posunutí bitů registru

Instrukce ”Posunout bit registru” posune hodnotu do posuvného registru. Tato instrukce nabízí snadný způsob pro kontrolu následnosti a řízení toku výrobků nebo dat. Instrukci použijte pro posunutí celého registru o jeden bit jednou za programový cyklus (scan).



Instrukce ”Posunout bit registru” posune hodnotu DATA do posuvného registru. S_BIT specifikuje nejnižší platný bit posuvného registru. N specifikuje délku posuvného registru a směr posunu (kladné posunutí = N, záporné posunutí = – N).

Každý bit vysunutý instrukcí SHRB je umístěn do bitu přetečení paměti (SM1.1).

Tato instrukce je definována jak nejnižším platným bitem (S_BIT), tak počtem bitu specifikovaných délkou (N).

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)
- 0092 (chyba v poli počtu)

Ovlivněné bity SM:

- SM1.1 (přetečení)

Tabulka 6-62 Operandy platné pro instrukci ”Posunout bit registru”

Vstupy/Výstupy	Typy dat	Operandy
DATA, S_Bit	BOOL	I, Q, V, M, SM, S, T, C, L
N	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta

Pro výpočet adresy nejvyššího platného bitu posuvného registru (MSB.b) použijte následující rovnici:

MSB.b = *[(Byte S_BIT) + ([N] – 1 + (bit S_BIT)) / 8].[zbytek dělení 8]*

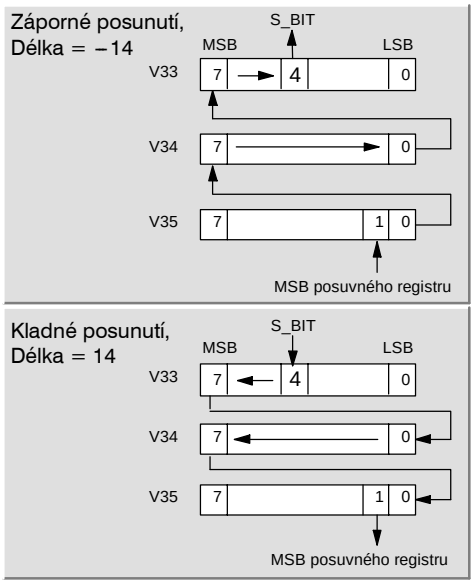
Například: jestliže S_BIT je V33.4 a N je 14, ukazuje následující výpočet, že MSB.b je V35.1.

MSB.b = V33 + ([14] – 1 +4)/8
= V33 + 17/8
= V33 + 2 se zbytkem 1
= V35.1

Při záporném posunutí indikovaném zápornou hodnotou délky (N) se vstupní data posunou do nejvyššího platného bitu posuvného registru a vysunou nejnižší platný bit (S_BIT). Vysunutá data jsou pak umístěna v bitu přetečení paměti (SM1.1).

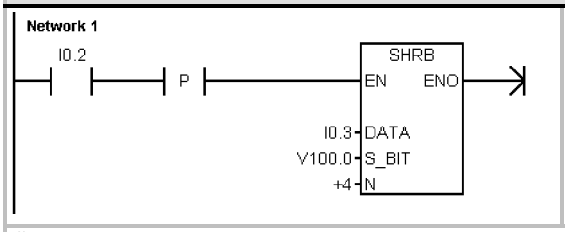
Při kladném posunutí indikovaném kladnou hodnotou délky (N) se vstupní data (DATA) vysunou ven z nejvyššího platného bitu posuvného registru a posunou se do nejnižšího platného bitu posuvného registru specifikovaného S_BIT. Vysunutá data jsou pak umístěna v bitu přetečení paměti (SM1.1).

Maximální délka posuvného registru je 64 bitů kladných nebo záporných. Obrázek 6-36 ukazuje posunutí bitů pro záporné a kladné hodnoty N.



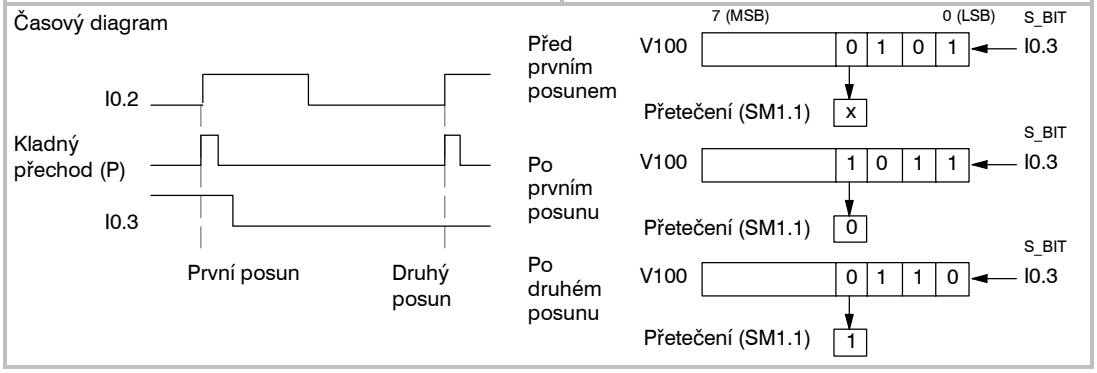
Obr. 6-36 Vstup a výstup posuvného registru

Příklad: Instrukce "Posunout bit registru"



Network 1

```
LD      I0.2
EU
SHRB    I0.3, V100.0, +4
```

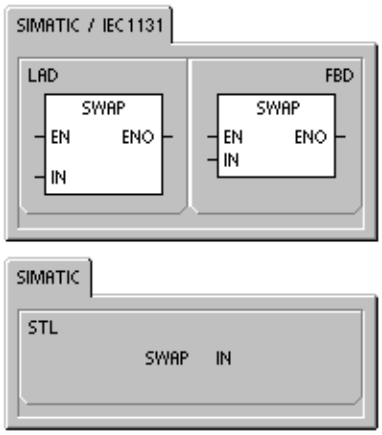


Výměna bytů

Instrukce ”Vyměnit byty” vymění nejvyšší platný byte slova (word) IN s jeho nejnižším platným byte.

Chyby, které nastavují ENO = 0

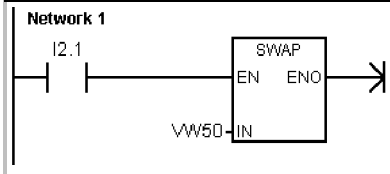
- 0006 (nepřímá adresa)



Tabulka 6-63 Operandy platné pro instrukci ”Vyměnit byty”

Vstupy/Výstupy	Typy dat	Operandy
IN	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC,*VD, *LD, *AC

Příklad: Instrukce výměny




```
Network 1
LD      I2.1
SWAP    VW50
```

Výměna VW50 [D6 C3] → VW50 [C3 D6]

Instrukce pro řetězce

Délka řetězce

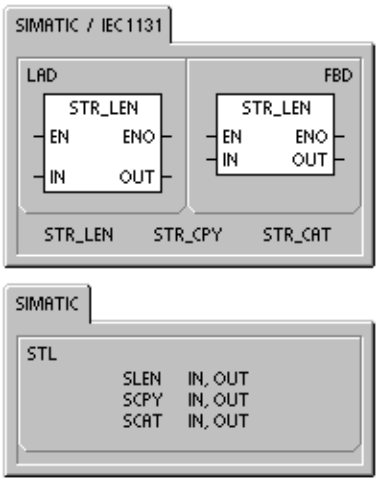
Instrukce ”Délka řetězce” (SLEN) vrací délku řetězce specifikovanou IN.

Kopírování řetězce

Instrukce ”Kopírovat řetězce” (SCPY) kopíruje řetězec specifikovaný IN do řetězce specifikovaného OUT.

Spojování řetězců

Instrukce ”Sloučit řetězce” (SCAT) připojí řetězec specifikovaný IN do řetězce specifikovaného OUT.



Bity SM a ENO

U instrukcí ”Délka řetězce”, ”Kopírovat řetězce” a ”Sloučit řetězce” ovlivní ENO následující podmínky.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (chyba rozsahu)

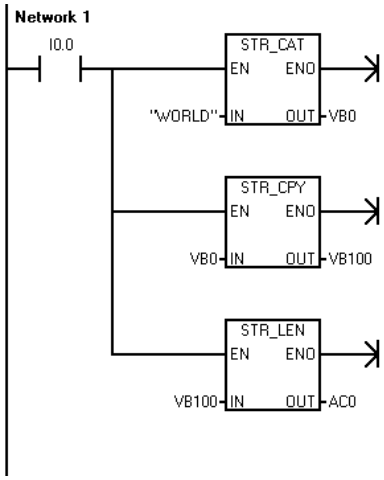
Tabulka 6-64 Operandy platné pro instrukci ”Délka řetězce”

Vstupy/Výstupy	Typy dat	Operandy
IN	ŘETĚZEC	VB, LB, *VD, *LD, *AC, řetězec konstant
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC

Tabulka 6-65 Operandy platné pro instrukce ”Kopírovat řetězec” a ”Sloučit řetězec”

Vstupy/Výstupy	Typy dat	Operandy
IN	ŘETĚZEC	VB, LB, *VD, *LD, *AC, řetězec konstant
OUT	ŘETĚZEC	VB, LB, *VD, *AC, *LD

Příklad: Instrukce ”Sloučit řetězce”, ”Kopírovat řetězec” a ”Délka řetězce”




```
Network 1      //1. Připojí řetězec na ”WORLD”
               //   k řetězci na VB0
               //2. Zkopíruje řetězec na VB0
               //   na nový řetězec na VB100
               //3. Získá délku řetězce,
               //   který začíná na VB100
LD      I0.0
SCAT    ”WORLD”, VB0
STRCPY  VB0, VB100
STRLEN  VB100, AC0
```

Před provedením programu

VB0						VB6
6	’H’	’E’	’L’	’L’	’O’	’ ’

Po provedení programu

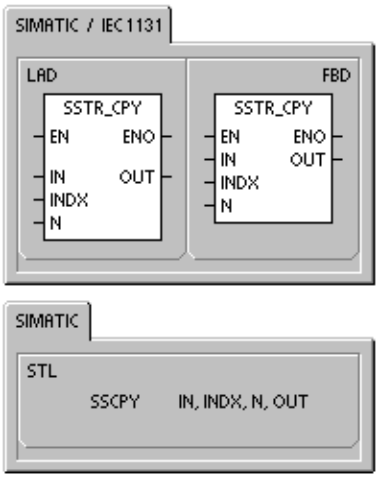
VB0											VB11
11	’H’	’E’	’L’	’L’	’O’	’ ’	’W’	’O’	’R’	’L’	’D’

VB100											VB111
11	’H’	’E’	’L’	’L’	’O’	’ ’	’W’	’O’	’R’	’L’	’D’

AC0
11

Kopírování části řetězce

Instrukce "Kopírovat část řetězce" (SSCPY) zkopíruje specifikovaný počet znaků N z řetězce specifikovaného IN, počínaje indexem INDX, do nového řetězce specifikovaného OUT.



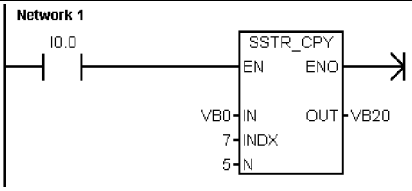
Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (chyba rozsahu)
- 009B (index=0)

Tabulka 6-66 Operandy platné pro instrukci "Kopírovat část řetězce"

Vstupy/Výstupy	Typy dat	Operandy
IN	ŘETĚZEC	VB, LB, *VD, *LD, *AC, řetězec konstant
OUT	ŘETĚZEC	VB, LB, *VD, *LD, *AC
INDX, N	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta

Příklad: Instrukce kopírování části řetězce




```
Network 1       //Počínaje sedmým znakem
                //z řetězce na VB0 zkopíruje 5 znaků
                //na nový řetězec na VB20
LD      I0.0
SSCPY   VB0, 7, 5, VB20
```

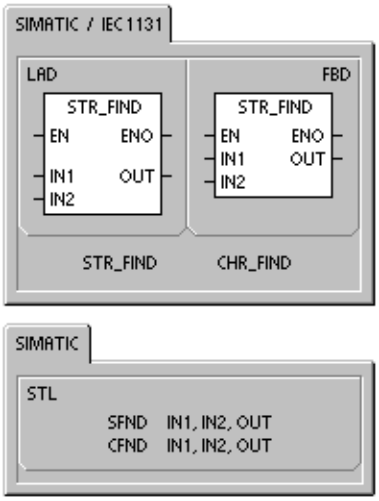
Před provedením programu

VB0											VB11
11	'H'	'E'	'L'	'L'	'O'	' '	'W'	'O'	'R'	'L'	'D'

Po provedení programu

VB20					VB25
5	'W'	'O'	'R'	'L'	'D'

Nalezení části řetězce



Instrukce ”Najít část řetězce” (SFND) hledá první výskyt řetězce IN2 uvnitř řetězce IN1. Vyhledávání začíná na počátečním místě specifikovaném OUT. Jestliže je nalezena sekvence znaků, která přesně odpovídá řetězci IN2, je umístění prvního znaku sekvence řetězce zapsáno do OUT. Pokud řetězec IN2 nebyl v řetězci IN1 nalezen, je instrukce OUT nastavena na 0.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (chyba rozsahu)
- 009B (index=0)

Nalezení znaku uvnitř řetězce

Instrukce ”Najít znak v řetězci” (CFND) prohledává řetězec IN1 na první výskyt kteréhokoliv znaku ze souboru znaku popsaného v řetězci IN2. Vyhledávání začíná na výchozím místě v OUT. Jestliže je nalezen odpovídající znak, je jeho umístění zapsáno do OUT. Není–li nalezen žádný odpovídající znak, je v OUT 0.

Chyby, které nastavují ENO = 0

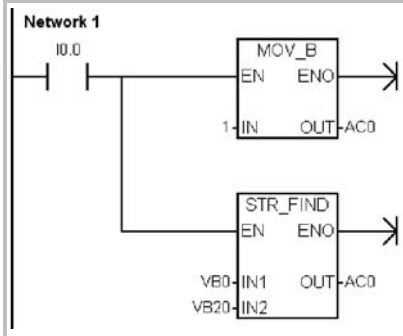
- 0006 (nepřímá adresa)
- 0091 (chyba rozsahu)
- 009B (index=0)

Tabulka 6-67 Operandy platné pro instrukce ”Najít část řetězce” a ”Najít znak v řetězci”

Vstupy/Výstupy	Typy dat	Operandy
IN1, IN2	ŘETĚZEC	VB, LB, *VD, *LD, *AC, řetězec konstant
OUT	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC

Příklad: Instrukce nalezení části řetězce

V následujícím příkladu je řetězec uložený na VB0 použit jako povel pro zapnutí nebo vypnutí čepadla. Řetězec 'On' (zapnutí) je uložen na VB20 a řetězec 'Off' (vypnutí) je uložen na VB30. Výsledek instrukce "Najít část řetězce" je uložen v AC0 (parametr OUT). Pokud výsledek není 0, byl řetězec 'On' nalezen v povelovém řetězci (VB12).




```
Network 1      //1. Nastaví AC0 na 1.
               //   (AC0 je použit jako parametr OUT.)
               //2. Prohledá řetězec na VB0 na vložený
               //   řetězec na VB20 ('On'), počínaje od
               //   prvního místa (AC0=1).
LD     I0.0
MOVB   1, AC0
SFND   VB0, VB20, AC0
```

VB0												VB12
12	'T'	'u'	'r'	'n'	'	'P'	'u'	'm'	'p'	'	'O'	'n'
					,					,		

VB20		VB22
2	'O'	'n'

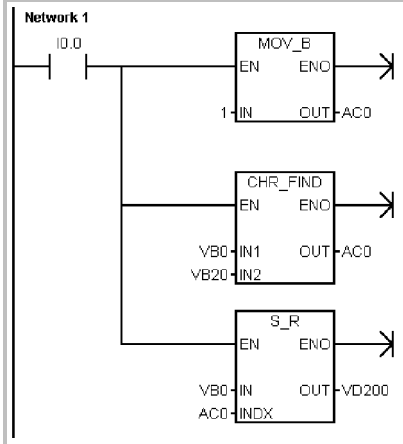
VB30			VB33
3	'O'	'f'	'f'

Je–li řetězec ve VB20 nalezen: AC0 = 11

Není–li řetězec ve VB20 nalezen: AC0 = 0

Příklad: Instrukce nalezení znaku uvnitř řetězce

V následujícím příkladu obsahuje řetězec uložený ve VB0 teplotu. Řetězec ve VB20 uchovává všechny číslice (a + a –), které mohou identifikovat v řetězci teplotu. Vzorový program nalezne počáteční umístění čísla v daném řetězci, a pak převede číslice na reálné číslo. VD200 uchovává hodnotu teploty danou reálným číslem.




```
Network 1      //1. Nastaví AC0 na 1.
               //   (AC0 je použit jako parametr OUT
               //   a ukazuje na první místo řetězce.)
               //2. Najde číslici
               //   v řetězci na VB0.
               //3. Konvertuje řetězec na reálné číslo.
LD     I0.0
MOVB   1, AC0
CFND   VB0, VB20, AC0
STR    VB0, AC0, VD200
```

VB0											VB11
11	'T'	'e'	'm'	'p'	'	'	'9'	'8'	'.'	'6'	'F'
					,	,					

VB20												VB32
12	'1'	'2'	'3'	'4'	'5'	'6'	'7'	'8'	'9'	'0'	'+'	'–'

Počáteční místo teploty uložené ve VB0: AC0 = 7

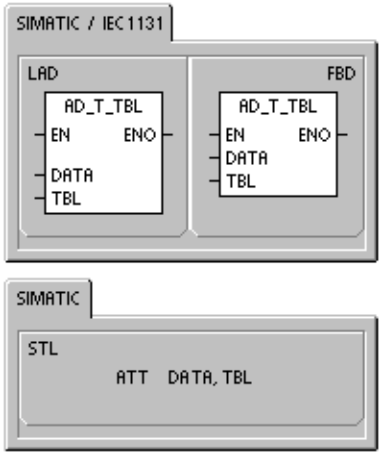
Hodnota teploty udaná reálným číslem: VD200 = 98,6

Instrukce pro práci s tabulkou

Přidávání do tabulky

Instrukce ”Přidat do tabulky” přidává hodnoty velikosti word (DATA) do tabulky (TBL). První hodnota v tabulce je její maximální délka (TL). Druhá hodota je počet položek (EC), která specifikuje počet položek v tabulce. Nové údaje se do tabulky přidávají za poslední položku. Pokaždé, když jsou do tabulky přidána nová data, se zvětší počet položek.

Tabulka může obsahovat až 100 datových položek.



Chyby, které nastavují ENO = 0

- SM1.4 (přetečení)
- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)

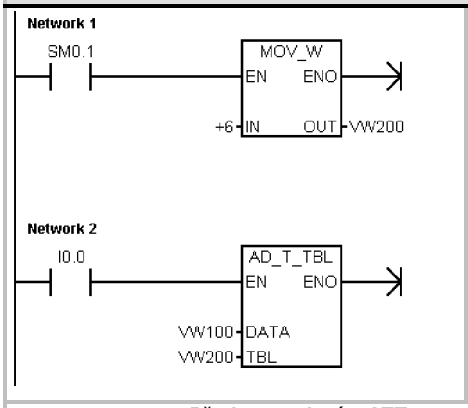
Ovlivněné bity SM:

- SM1.4 je nastaven na 1, jestliže se snažíte tabulku přeplnit

Tabulka 6-68 Operandy platné pro instrukce tabulky

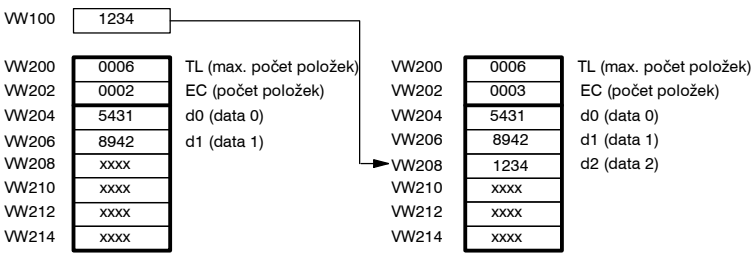
Vstupy/Výstupy	Typy dat	Operandy
DATA	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
TBL	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, *VD, *LD, *AC

Příklad: Instrukce ”Přidat do tabulky”



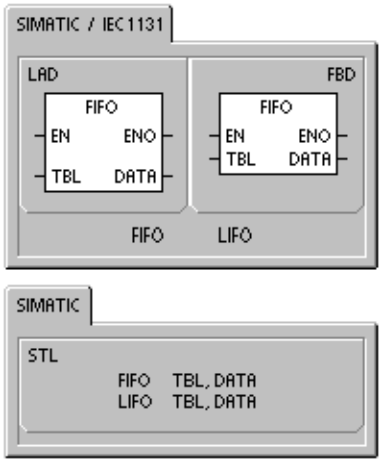

```
Network 1       //Nahraje maximální délku tabulky
LD      SM0.1
MOVW    +6, VW200

Network 2
LD      I0.0
ATT     VW100, VW200
```



Instrukce FIFO a LIFO

Tabulka může obsahovat až 100 datových položek.



První dovnitř, první ven (FIFO)

Instrukce "První dovnitř, první ven" (FIFO) posune nejstarší (neboli první) položku tabulky na adresu výstupní paměti tak, že odstraní první položku v tabulce (TBL) a posune ji na místo specifikované v DATA. Všechny ostatní položky tabulky se posunou o jedno místo. Počet položek v tabulce se po každém provedení instrukce snižuje.

Poslední dovnitř, první ven (LIFO)

Instrukce "Poslední dovnitř, první ven" (LIFO) posune nejnovější (neboli poslední) položku tabulky na adresu výstupní paměti tak, že odstraní poslední položku v tabulce (TBL) a posune ji na místo specifikované v DATA. Počet položek v tabulce se po každém provedení instrukce snižuje.

Chyby, které nastavují ENO = 0

- SM1.5 (prázdná tabulka)
- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)

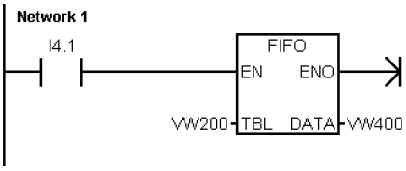
Ovlivněné bity SM:

- SM1.5 se nastaví na 1, jestliže se pokusíte odstranit položku z prázdné tabulky

Tabulka 6-69 Operandy platné pro instrukce "První dovnitř, první ven" a "Poslední dovnitř, první ven"

Vstupy/Výstupy	Typy dat	Operandy
TBL	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, *VD, *LD, *AC
DATA	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AQW, *VD, *LD, *AC

Příklad: Instrukce "První dovnitř, první ven"




```
Network 1
LD      I4.1
FIFO    VW200, VW400
```

Před provedením FIFO

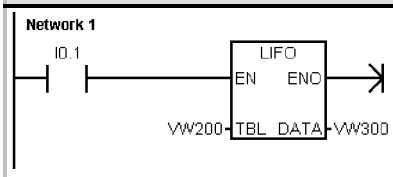
VW200	0006	TL (max. počet položek)
VW202	0003	EC (počet položek)
VW204	5431	d0 (data 0)
VW206	8942	d1 (data 1)
VW208	1234	d2 (data 2)
VW210	xxxx	
VW212	xxxx	
VW214	xxxx	

VW400 5431

Po provedení FIFO

VW200	0006	TL (max. počet položek)
VW202	0002	EC (počet položek)
VW204	8942	d0 (data 0)
VW206	1234	d1 (data 1)
VW208	xxxx	
VW210	xxxx	
VW212	xxxx	
VW214	xxxx	

Příklad: Instrukce "Poslední dovnitř, první ven"



Network 1

LD I0.1
LIFO VW200, VW300

Před provedením LIFO

VW200	0006	TL (max. počet položek)
VW202	0003	EC (počet položek)
VW204	5431	d0 (data 0)
VW206	8942	d1 (data 1)
VW208	1234	d2 (data 2)
VW210	xxxx	
VW212	xxxx	
VW214	xxxx	

VW300 1234

Po provedení LIFO

VW200	0006	TL (max. počet položek)
VW202	0002	EC (počet položek)
VW204	5431	d0 (data 0)
VW206	8942	d1 (data 1)
VW208	xxxx	
VW210	xxxx	
VW212	xxxx	
VW214	xxxx	

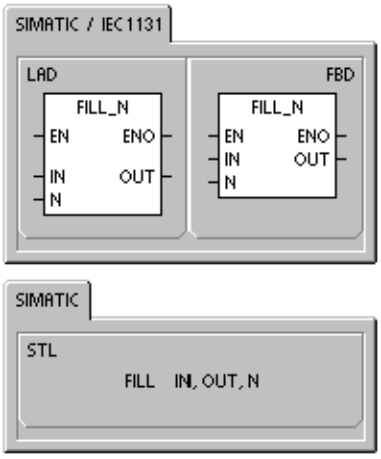
Vyplnění paměti

Instrukce ”Vyplnit paměť” (FILL) zapíše N za sebou jdoucích wordů, které začínají na adrese OUT, s hodnotou slova obsaženou na adrese IN.

N má rozsah od 1 do 255.

Chyby, které nastavují ENO = 0

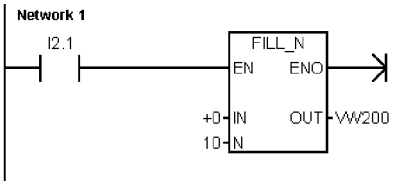
- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)



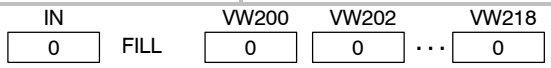
Tabulka 6-70 Operandy platné pro instrukci ”Vyplnit paměť”

Vstupy/Výstupy	Typy dat	Operandy
IN	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
N	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *LD, *AC, konstanta
OUT	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AQW, *VD, *LD, *AC

Příklad: Instrukce ”Vyplnit paměť”

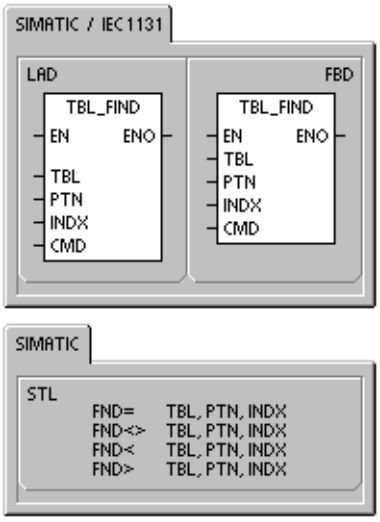



```
Network 1
LD      I2.1
FILL    +0, VW200, 10
```



Prohledávání tabulky

Instrukce ”Najít v tabulce” (FND) prohledává tabulku a hledá data, která odpovídají určitým kritériím. Instrukce ”Najít v tabulce” prohledává tabulku TBL počínaje položkou tabulky INDX a hledá datovou hodnotu nebo obrazec, které odpovídají vyhledávacím kritériím definovaným v CMD. Parametr povelu CMD je daný číselnou hodnotou 1 až 4, kde 1 znamená =, 2 znamená <>, 3 znamená < a 4 znamená>.



Když se najde vhodná hodnota, ukáže INDX na odpovídající položku tabulky. Chcete–li hledat další odpovídající položku, je třeba nejdříve inkrementovat INDX a teprve pak opět spustit instrukci ”Najít v tabulce”. Pokud se odpovídající hodnota nenajde, rovná se hodnota INDX počtu položek.

Tabulka může obsahovat až 100 datových položek. Datové položky (prohledávaný prostor) jsou číslovány od 0 do maximálně 99.

Chyby, které nastavují ENO = 0

- 0006 (nepřímá adresa)
- 0091 (operand mimo rozsah)

Tabulka 6-71 Operandy platné pro instrukci ”Najít v tabulce”

Vstupy/Výstupy	Typy dat	Operandy
TBL	WORD	IW, QW, VW, MW, SMW, T, C, LW, *VD, *LD, *AC
PTN	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta
INDX	WORD	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, *VD, *LD, *AC
CMD	BYTE	(Konstanta) 1: Rovná se (=), 2: Nerovná se (<>), 3: Menší než (<), 4: Větší než (>)

Tip

Pokud instrukci ”Najít v tabulce” použijete pro tabulky generované pomocí instrukcí ”Přidat do tabulky”, ”Poslední dovnitř, první ven” a ”První dovnitř, první ven”, počet položek přímo odpovídá datovým položkám. Parametr maximálního počtu položek, který je vyžadován pro instrukce”přidat do tabulky”, ”Poslední dovnitř, první ven” nebo ”První dovnitř, první ven”, instrukce ”Najív v tabulce” nevyžaduje. Viz obrázek 6-37.

Proto musíte nastavit operand TBL instrukce ”Najít v tabulce” na adresu o jedno slovo (dva byty) vyšší než operand TBL odpovídající instrukce ”Přidat do tabulky”, ”Poslední dovnitř, první ven” nebo ”První dovnitř, první ven”

Formát tabulky pro ATT, LIFO a FIFO

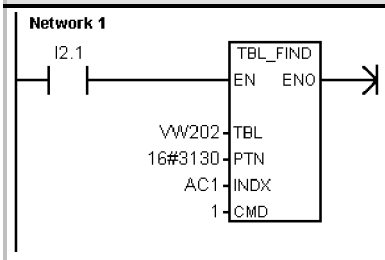
VW200	0006	TL (max. počet položek)
VW202	0006	EC (počet položek)
VW204	xxxx	d0 (data 0)
VW206	xxxx	d1 (data 1)
VW208	xxxx	d2 (data 2)
VW210	xxxx	d3 (data 3)
VW212	xxxx	d4 (data 4)
VW214	xxxx	d5 (data 5)

Formát tabulky pro TBL_FIND

VW202	0006	EC (počet položek)
VW204	xxxx	d0 (data 0)
VW206	xxxx	d1 (data 1)
VW208	xxxx	d2 (data 2)
VW210	xxxx	d3 (data 3)
VW212	xxxx	d4 (data 4)
VW214	xxxx	d5 (data 5)

Obr. 6-37 Rozdíl ve formátu tabulky instrukce ”Najít v tabulce” a tabulky instrukcí ATT, LIFO a FIFO

Příklad: Instrukce ”Najít v tabulce”




```
Network 1
LD      I2.1
FND=    VW202, 16#3130, AC1
```

Pokud má I2.1 hodnotu logické 1, prohledá tabulku a hledá hodnotu, která se rovná 3130 HEX.

VW202	0006	EC (počet položek)
VW204	3133	d0 (data 0)
VW206	4142	d1 (data 1)
VW208	3130	d2 (data 2)
VW210	3030	d3 (data 3)
VW212	3130	d4 (data 4)
VW214	4541	d5 (data 5)

Pokud byla tabulka vytvořena pomocí instrukcí ATT, LIFO a FIFO, obsahuje VW200 maximální počet položek a instrukce prohledávání ho nevyžadují.

AC1 0 — AC1 musí být nastaven na 0, aby prohledávání probíhalo v tabulce odshora dolů.

Provede prohledávání tabulky

AC1 2 — AC1 obsahuje číslo datové položky, která odpovídá první nalezené shodě v tabulce (d2).

AC1 3 — Inkrementuje INDX o jednu, než začne prohledávat zbývající položky tabulky.

Provede prohledávání tabulky

AC1 4 — AC1 obsahuje číslo datové položky odpovídající druhé nalezené shodě v tabulce (d4).

AC1 5 — Inkrementuje INDX o jednu, než začne prohledávat zbývající položky tabulky.

Provede prohledávání tabulky

AC1 6 — AC1 obsahuje hodnotu, která je rovna počtu položek. Celá tabulka byla prohledána a další shoda nebyla nalezena.

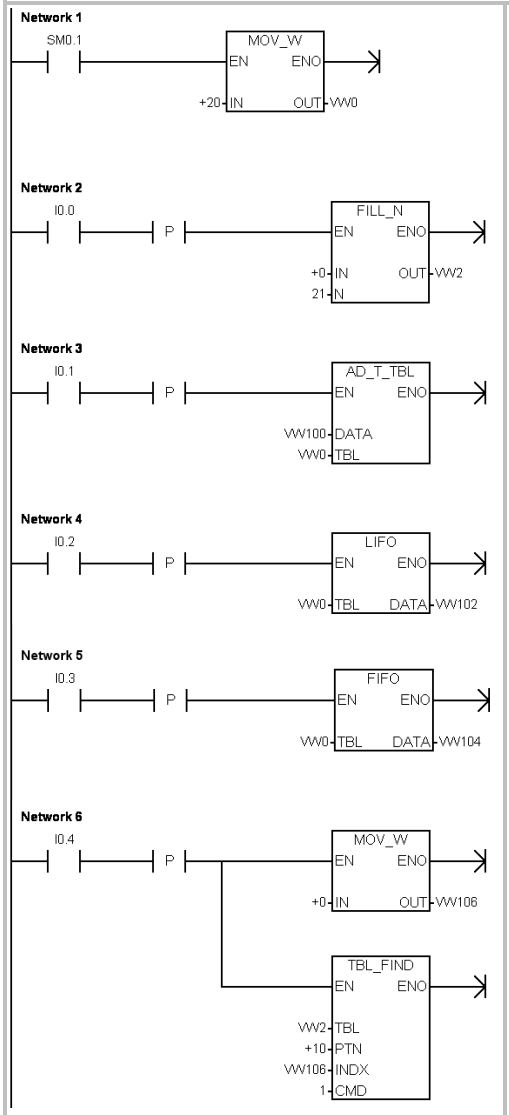
AC1 0 — Před opětovným prohledáním tabulky musí být hodnota INDX resetována na 0.

Příklad: Vytvoření tabulky

Následující program vytvoří tabulku se 20 položkami. První paměťové místo tabulky obsahuje délku tabulky (v tomto případu 20 položek). Druhé paměťové místo ukazuje aktuální počet položek tabulky. Ostatní místa obsahují jednotlivé položky. Tabulka může obsahovat až 100 položek. Příklad nezahrnuje parametry definující maximální délku tabulky nebo skutečný počet položek (v tomto případě VW0 a VW2). Při každém příkazu CPU automaticky zvyšuje nebo snižuje aktuální počet položek v tabulce (zde VW2).

Dříve než začnete pracovat s tabulkou, stanovte maximální počet položek tabulky. Jinak nemůžete položky do tabulky vkládat. Také zkontrolujte, že všechny příkazy pro čtení a zápis jsou aktivovány hranou.

Chcete–li tabulku prohledávat, musí být index (VW106) nastaven na 0 před spuštěním prohledávání. Jestliže je nalezena shoda, bude mít index číslo položky tabulky, ale jestliže se žádná shoda nenajde, bude se index rovnat současnému počtu položek v tabulce (VW2).




```
Network 1       //Vytvoří tabulku s 20 položkami, která
                //začíná na paměťovém místě 4.
                //1. Při prvním programovém cyklu
                //definuje maximální délku tabulky.
LD      SM0.1
MOVW    +20, VW0

Network 2       //Resetuje tabulku vstupem I0.0
                //Při náběžné hraně na I0.0 vyplní
                //paměťová místa od VW2 "+0" .
LD      I0.0
EU
FILL    +0, VW2, 21

Network 3       //Zapíše hodnotu do tabulky vstupem
                //I0.1. Při náběžné hraně na I0.1
                //zkopíruje hodnotu paměťového místa
                //VW100 do tabulky.
LD      I0.1
EU
ATT     VW100, VW0

Network 4       //Načte první hodnotu tabulky
                //vstupem I0.2. Přesune poslední
                //hodnotu v tabulce na místo VW102.
                //Tím se sníží počet položek. Při
                //náběžné hraně na I0.2. Přesune
                //poslední hodnotu v tabulce na místo
                //VW102.
LD      I0.2
EU
LIFO    VW0, VW102

Network 5       //Načte poslední hodnotu tabulky
                //vstupem I0.3. Přesune první hodnotu
                //v tabulce na místo VW102. Tím se
                //sníží počet položek. Při náběžné
                //hraně na I0.0. Přesune první hodnotu
                //v tabulce na pW104
LD      I0.3
EU
FIFO    VW0, VW104

Network 6       //Vyhledá v tabulce na první místo,
                //které má hodnotu 10.
                //1. Při náběžné hraně na I0.4
                //   resetuje ukazatel indexu.
                //2. Najde položku tabulky, která se
                //   rovná 10.
LD      I0.4
EU
MOVW    +0, VW106
FND=    VW2, +10, VW106
```

Instrukce časovače

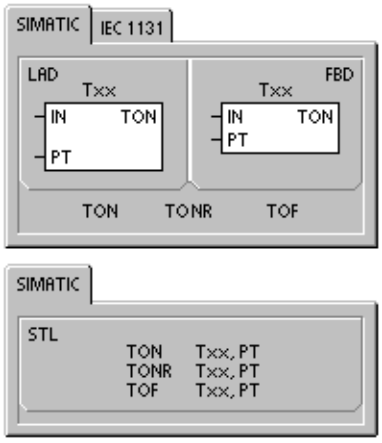
Časovač SIMATIC

Časovač zpožděného zapnutí
Remanentní časovač zapnutí

Instrukce ”Časovač zapnutí” (TON)” a ”Remanentní časovač zapnutí” (TONR) odpočítávají čas od sepnutí vstupu. Číslo časovače (Txx) určuje jednotlivé časovače a rozlišení se zobrazí v blokové instrukci.

Časovač zpožděného vypnutí

Instrukce ”Časovač vypnutí” (TOF) se používá na zpoždění rozepnutí výstupu o pevný čas po rozepnutí vstupu. Číslo časovače (Txx) určuje jednotlivé časovače a rozlišení se zobrazí v blokové instrukci.



Tabulka 6-72 Operandy platné pro SIMATIC instrukce časovače

Vstupy/Výstupy	Typy dat	Operandy
Txx	WORD	Konstanta (T0 až T255)
IN	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
PT	INT	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *LD, *AC, konstanta



Tip

Není možné mít stejné číslo časovače (Txx) pro časovač vypnutí (TOF) a časovač zapnutí (TON). Nemůžete mít například TON T32 i TOF T32.

Jak ukazuje tabulka 6-73, uvedené tři typy časovačů provádějí různé typy časovacích úloh:

- TON je možné použít pro intervalové časování.
- TONR je možné použít pro akumulaci více časovaných intervalů.
- TOF je možné použít pro prodloužení času po vypnutí, například pro vychlazení motoru po jeho vypnutí.

Tabulka 6-73 Operace instrukcí časovače

Typ	Aktuální >= Nastavená	Stav povolovacího vstupu (IN)	Výpadek napájení nebo první cyklus
TON	Časovací bit zapnut Pokračuje počítání aktuální hodnoty do 32.767	ZAPNUTO: Aktuální hodnota odpočítává čas VYPNUTO: Bit časovače vypnutý, aktuální hodnota = 0	Bit časovače vypnutý Aktuální hodnota = 0
TONR	Časovací bit zapnut Pokračuje počítání aktuální hodnoty do 32.767	ZAPNUTO: Aktuální hodnota odpočítává čas VYPNUTO: Bit časovače a aktuální hodnota udržují poslední stav	Bit časovače vypnutý Je možné udržet aktuální hodnotu[1]
TOF	Bit časovače vypnutý Aktuální = Nastavená, konec počítání	ZAPNUTO: Bit časovače zapnutý, aktuální hodnota = 0 VYPNUTO: Časovač počítá po přechodu ze zapnutého do vypnutého stavu	Bit časovače vypnutý Aktuální hodnota = 0

[1] Pro uchování hodnoty časovače i po výpadku napájení je možné nastavit remanenci aktuální hodnoty. Více informací o remanentní paměti CPU S7-200 najdete v kapitole 4.


Tipy a triky

Vzorový program s použitím časovače zpožděního zapnutí (TON) najdete v programovacích tipech na CD s dokumentací. Viz tip 31.

Instrukce TON a TONR odpočítávají čas, když je jejich vstup zapnutý. Pokud je aktuální hodnota rovna nebo větší než nastavený čas, zapne se bit časovače.

- ❏ Když se rozepne jeho vstup IN, aktuální hodnota časovače TON je smazána, zatímco aktuální hodnota časovače TONR se při rozepnutí vstupu uchová.
- ❏ Časovač TONR je možné použít pro akumulaci času při sepnutí a rozeputí vstupu IN. Aktuální hodnotu TONR lze vymazat pomocí instrukce ”Reset” (R).
- ❏ Jak časovač TON, tak časovač TONR pokračují v počítání i po dosažení nastavené hodnoty a přestanou počítat na maximální hodnotě 32.767.

Instrukce TOF se používá na zpoždění rozepnutí výstupu o pevný čas po rozepnutí vstupu. Když se sepne povolovací vstup, zapne se okamžitě bit časovače a aktuální hodnota je nastavena na 0. Když se vstup rozepne, časovač počítá, dokud uplynulý čas nedosáhne nastavené hodnoty.

- ❏ Když je dosažena nastavená hodnota, vypne se bit časovače a přestane se zvětšovat aktuální hodnota; pokud se však vstup opět sepne dříve, než TOF dosáhne nastavené hodnoty, zůstává bit časovače zapnutý.
- ❏ Aby začal TOF počítat časové intervaly, musí povolovací vstup provést přechod ze sepnutého do rozepnutého stavu.
- ❏ Pokud je časovač TOF uvnitř oblasti SCR a oblast SCR je neaktivní, je aktuální hodnota nastavena na 0, bit časovače se vypne a aktuální hodnota se nezvyšuje.



Tip

TONR je možné resetovat pouze pomocí instrukce Reset (R). Instrukci ”Reset” můžete také použít pro resetování kteréhokoliv časovače TON nebo TOF. Instrukce ”Reset” provádí následující operace:

- Bit časovače = vypnutý
- Aktuální hodnota časovače = 0

Mají–li časovače TOF opět začít počítat, potřebují po resetování přechod vstupu IN ze sepnutého do rozepnutého stavu.

Stanovení rozlišení časovače

Časovače počítají časové intervaly. Rozlišení (neboli časová základna) časovače určuje délku nejmenšího časového intervalu. Například časovač TON s rozlišením 10 ms počítá, kolik 10ms intervalů uplyne po tom, co byl časovač TON spuštěn: počet 50 představuje na 10ms časovači 500 ms. Časovače SIMATIC nabízí tři rozlišení: 1 ms, 10 ms a 100 ms. Jak je patrné z tabulky 6-74, číslo časovače ukazuje jeho rozlišení.

Tip

Pro zaručení minimálního časového intervalu zvyšte požadovanou hodnotu (PV) o 1; například pro zajištění minimálního časovaného intervalu nejméně 2100 ms u 100ms časovače nastavte PV na 22.

Tabulka 6-74 Čísla a rozlišení časovačů

Typ časovače	Odstranění	Maximální hodnota	Číslo časovače
TONR (remanentní)	1 ms	32,767 s (0,546 min)	T0, T64
	10 ms	327,67 s (5,46 min)	T1 až T4, T65 až T68
	100 ms	3276,7 s (54,6 min)	T5 až T31, T69 až T95
TON, TOF (bez remanence)	1 ms	32,767 s (0,546 min)	T32, T96
	10 ms	327,67 s (5,46 min)	T33 až T36, T97 až T100
	100 ms	3276,7 s (54,6 min)	T37 až T63, T101 až T255

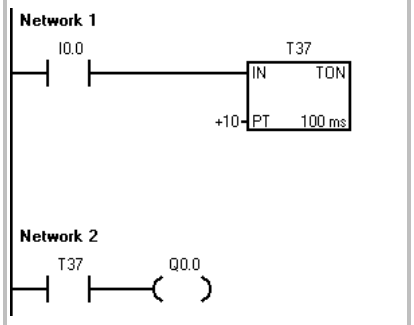
Vysvětlení, jak rozlišení ovlivňuje činnost časovače

U časovače s rozlišením 1 ms jsou bit časovače a aktuální hodnota aktualizovány asynchronně s programovým cyklem. U programových cyklů delších než 1 ms jsou bit časovače a aktuální hodnota aktualizovány během cyklu několikrát.

U časovače s rozlišením 10 ms jsou bit časovače a aktuální hodnota aktualizovány na začátku každého programového cyklu. Bit časovače a aktuální hodnota zůstávají během programového cyklu konstantní; časové intervaly, které se během cyklu nakumulují, se na začátku každého programového cyklu přičtou k aktuální hodnotě.

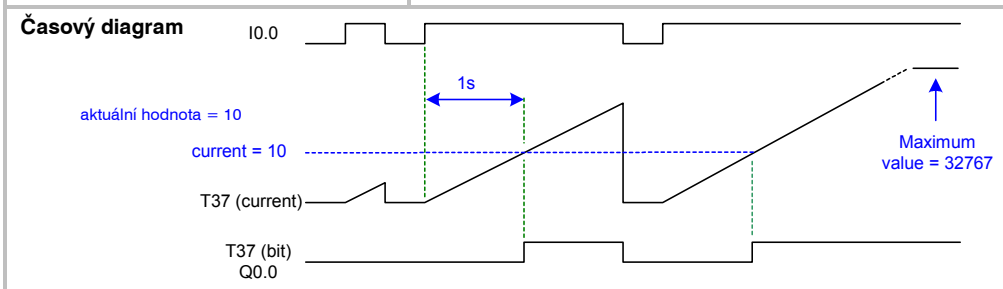
U časovače s rozlišením 100 ms jsou bit časovače a aktuální hodnota aktualizovány při provedení instrukce; proto zajistěte, aby program vykonával instrukci pro 100ms časovač pouze jedenou za programový cyklus, aby si časovač uchoval správné časování.

Příklad: Časovač zpoždění náběžné hrany (On – Delay) SIMATIC



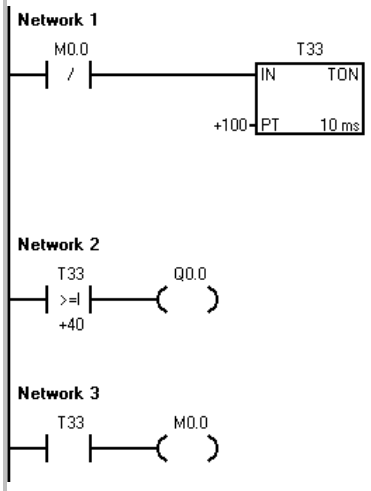

```
Network 1      //100ms časovač T37 časově odpojuje poté, co
               //(10 x 100 ms = 1s)
               //I0.0 ON=T37 zapnut,
               //I0.0 OFF=vypnut a reset T37
LD     I0.0
TON    T37, +10

Network 2      //Bit T37 je řízen časovačem T37
LD     T37
=      Q0.0
```



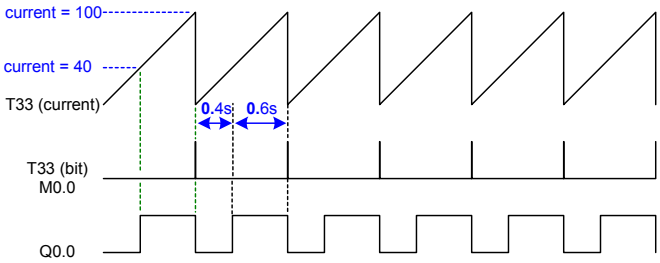
Tip
Aby bylo zaručeno, že se výstup časovače s automatickým opakováním sepne v jednom programovém cyklu pokaždé, když časovač dosáhne požadované hodnoty, použijte jako povolovací vstup časovače kontakt, který je v klidu zavřen (namísto bitu časovače).

Příklad: Časovač zpožděného zapnutí SIMATIC s automatickým opakováním

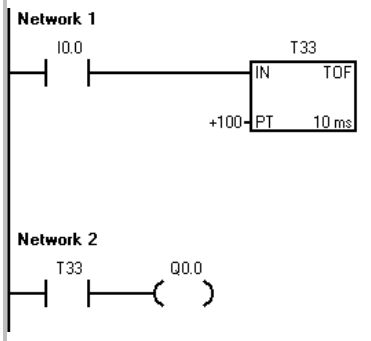



```
Network 1     //10ms časovač T33 časově odpojuje poté, co
              //(100 x 10 ms = 1s)
              //M0.0 impulz je příliš rychlý k monitorování
              //stavu programu.
LDN     M0.0
TON     T33, +100
Network 2     //Porovnání je pravdivé při
              //rychlosti, která je vidět na stavu programu.
              //Q0.0 se zapíná v intervalu (40 x 10 ms)
              //pro časový průběh 40 % vypnuto / 60 % zapnuto
LDW>=   T33, +40
=       Q0.0
Network 3     //Pulz T33 (bit) je příliš rychlý k monitorování
              //stavu programu.
              //Reset časovače přes M0.0 po uplynutí
              //doby (100 x 10 ms)
LD      T33
=       M0.0
```

Časový diagram



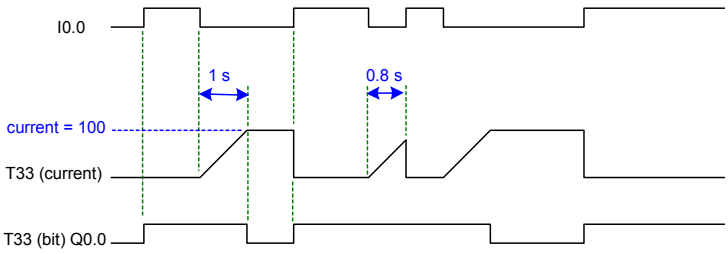
Příklad: Časovač zpožděného vypnutí SIMATIC-



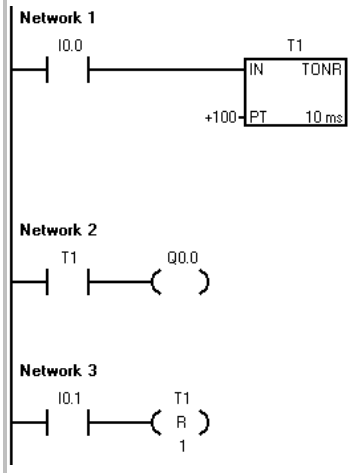

```
Network 1     //10ms časovač T33 časuje po (100 x 10 ms = 1s)
              //I0.0 přejde ze sepnutého do rozepnutého stavu
              //(ON-to-OFF) = T33 povolen
              //I0.0 přejde z rozepnutého do sepnutého stavu
              //(OFF-to-ON) = zablokování a reset T33
LD      I0.0
TOF     T33, +100

Network 2     //Časovač T33 řídí Q0.0 přes kontakt časovače T33
LD      T33
=       Q0.0
```

Časový diagram



Příklad: Remanentní časovač zpožděného zapnutí SIMATIC

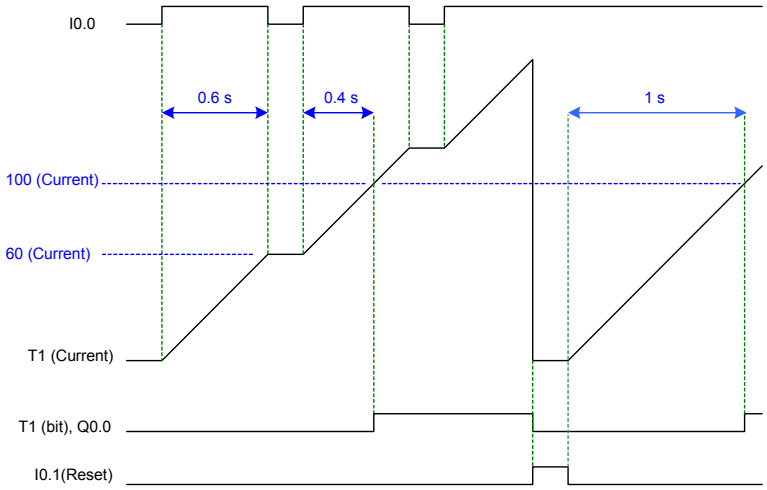



```
Network 1       //10ms časovač TONR T1 časově odpojuje na
                //PT=(100 x 10 ms=1s)
LD      I0.0
TONR    T1, +100

Network 2       //Bit T1 je řízen časovačem T1.
                //Sepne Q0.0 po tom, co časovač
                //souhrnně napočítá 1 sekundu.
LD      T1
=       Q0.0

Network 3       //Časovače TONR musí být resetovány pomocí
                //instrukce "Reset"
                //Když je sepnutý I0.1, resetuje časovač T1 (aktuální
hodnotu a bit).
LD      I0.1
R       T1, 1
```

Časový diagram



Instrukce časovače IEC

Časovač zpožděného zapnutí

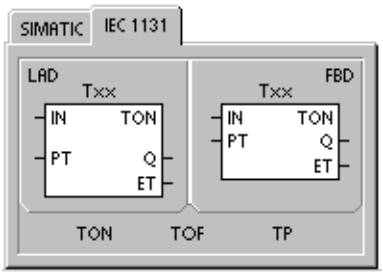
Instrukce "Časovač zapnutí" (TON) odpočítává čas, když je sepnut povolovací vstup.

Časovač zpožděného vypnutí

Instrukce "Časovač vypnutí" (TOF) zpožďuje rozepnutí výstupu o pevný čas po rozepnutí vstupu.

Pulzní časovač

Instrukce "Pulzní časovač" (TP) generuje pulzy specifické délky.



Tabulka 6-75 Operandy platné pro IEC instrukce časovače

Vstupy/Výstupy	Typy dat	Operandy
Txx	TON, TOF, TP	Konstanta (T32 až T63, T96 až T255)
IN	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
PT	INT	IW, QW, VW, MW, SMW, SW, LW, AC, AIW, *VD, *LD, *AC, konstanta
Q	BOOL	I, Q, V, M, SM, S, L
ET	INT	IW, QW, VW, MW, SMW, SW, LW, AC, AQW, *VD, *LD, *AC

Tip
Časovače TOF, TON a TP nemohou být označeny stejnými čísly. Nemůžete mít například TON T32 i TOF T32.

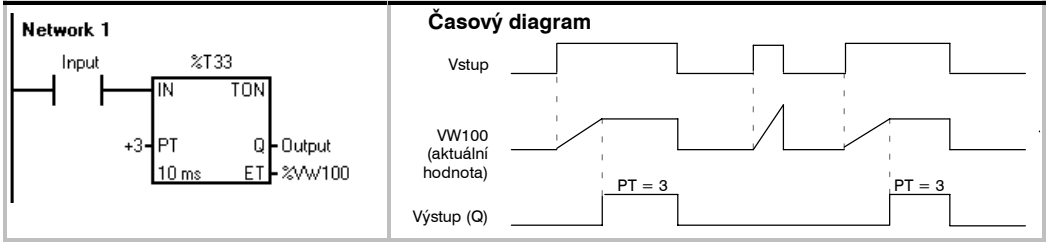
- Instrukce TON počítá časové intervaly až do požadované hodnoty, když je vstup (IN) v log. 1. Když se uplynulý čas (ET) rovná nastavenému času (PT), nastaví se výstupní bit (Q) časovače na log. 1. Výstupní bit je resetován po deaktivaci vstupu IN. Když je dosaženo požadované hodnoty, časování se zastaví a časovač je zablokován.
- Instrukce TOF zpožďuje rozepnutí o pevný čas po nastavení vstupu na log. 0. Když se rozepne vstup IN, začne TOF počítat čas až do požadované hodnoty. Když se uplynulý čas (ET) rovná nastavenému času (PT), nastaví se výstupní bit (Q) časovače na log. 0. Když je dosaženo požadované hodnoty, nastaví se výstupní bit časovače na log. 0 a uplynulý čas je uchováván, dokud vstup IN nepřejde do stavu log. 1. Jestliže vstup IN nastaví přechod do log. 0 na dobu kratší, než je nastavený čas, zůstane výstupní bit v log. 1.
- Instrukce TP generuje pulzy specifické délky. Když se sepne vstup (IN), nastaví se výstupní bit (Q). Výstupní bit zůstane v log. 1 po dobu trvání pulzu v rámci nastaveného času (PT). Dosáhne–li uplynulý čas (ET) požadované hodnoty (PT), výstupní bit se nastaví na log. 0. Uplynulý čas je uchováván až do doby, než se rozepne vstup IN. Když se nastaví výstupní bit, zůstane zapnutý, dokud neuplyne čas pulzu.

Každá aktuální hodnota je násobkem časové základny. Například počet 50 představuje na 10ms časovači 500 ms. Časovače IEC (TON, TOF a TP) jsou k dispozici se třemi rozlišeními: Rozlišení je dáno číslem časovače, jak ukazuje tabulka 6-76.

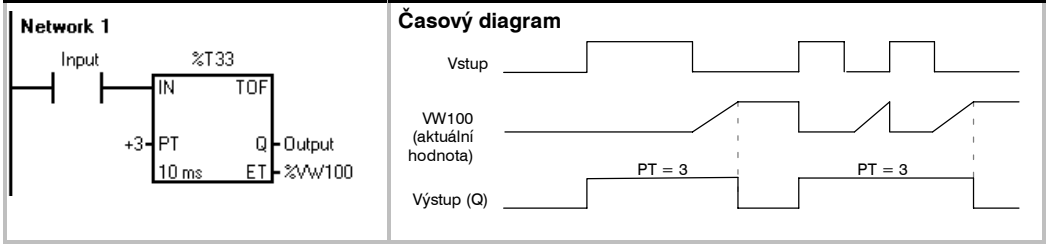
Tabulka 6-76 Rozlišení časovačů IEC

Rozlišení	Maximální hodnota		Číslo časovače
1 ms	32,767 s	(0,546 minut)	T32, T96
10 ms	327,67 s	(5,46 minut)	T33 až T36, T97 až T100
100 ms	3276,7 s	(54,6 minut)	T37 až T63, T101 až T255

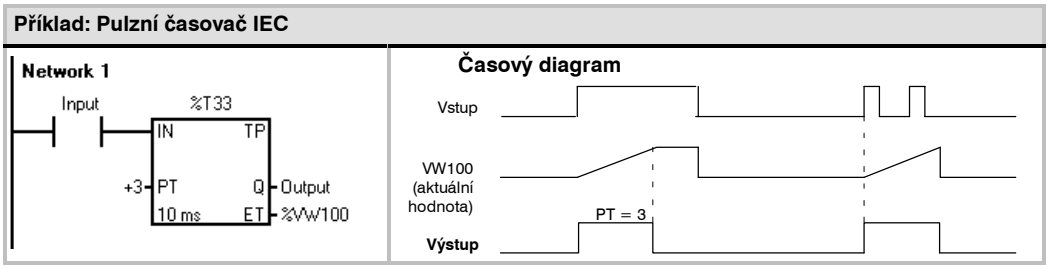
Příklad: Časovače zpožděného zapnutí IEC



Příklad: Časovače zpožděného vypnutí IEC



Příklad: Pulzní časovač IEC



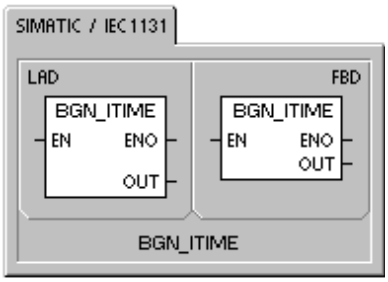
Intervalové hodiny

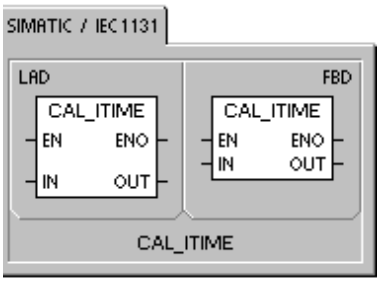
Začátek časového intervalu

Instrukce “Začátek časového intervalu” (BITIM) čte aktuální hodnotu vestavěného 1ms čítače a ukládá hodnotu do OUT. Maximální časový interval pro hodnotu DWORD v ms je 2 na 32 nebo 49,7 dní.

Výpočet časového intervalu

Instrukce “Výpočet časového intervalu” (CITIM) vypočítá časový rozdíl mezi aktuálním časem a časem uvedeným v IN. Rozdíl se ukládá do OUT. Maximální časový interval pro hodnotu DWORD v ms je 2 na 32 nebo 49,7 dní. CITIM automaticky řídí 1ms prodloužení, které se objevuje v rámci maximálního intervalu podle toho, kdy byla provedena instrukce BITIM.

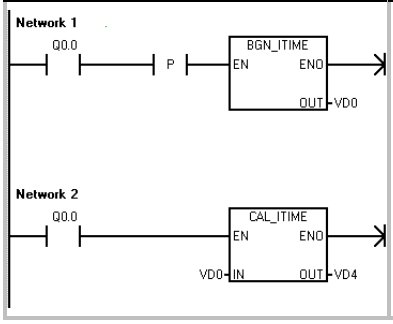




Tabulka 6-77 Operandy platné pro instrukce časovače intervalů

Vstupy/Výstupy	Typy dat	Operandy
IN	DWORD	VD, ID, QD, MD, SMD, SD, LD, HC, AC, *VD, *LD, *AC
OUT	DWORD	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

Příklad: Počáteční časový interval SIMATIC a výpočet časového intervalu




```
Network 1       //Zachytí čas, který zapnul Q0.0.

LD      Q0,0
EU
BITIM   VD0

Network 2       //Vypočítá čas, po který byl zapnut Q0.0.

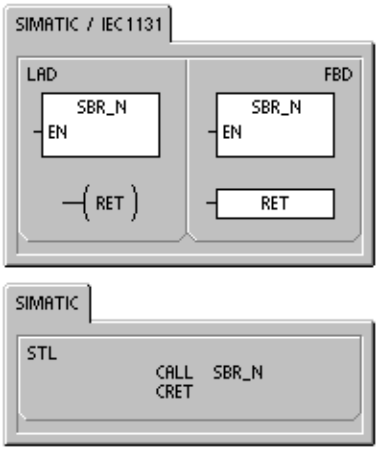
LD      Q0,0
CITIM   VD0, VD4
```

Instrukce podprogramů

Instrukce “Volání podprogramu” (CALL) přenáší řízení do podprogramu SBR_N. Instrukci ”Volání podprogramu” můžete použít s parametry nebo bez nich. Po skončení provádění podprogramu se řízení vrací na instrukci, která následuje po instrukci ”Volání podprogramu”.

Instrukce ”Podmíněný návrat z podprogramů” (CRET) ukončuje podprogram na základě předcházející logiky.

Chcete–li přidat podprogram, vyberte povel menu **Úpravy > Vložit > Podprogram**.



Chyby, které nastavují ENO = 0

- 0008 (překročen maximální povolený počet vnoření podprogramů)
- 0006 (nepřímá adresa)

Z hlavního programu můžete vnořovat podprogramy (umístit volání podprogramu do jiného podprogramu) až do hloubky osm. Z přerušení podprogramy vnořovat nemůžete.

Volání podprogramu nemůže být umístěno v podprogramu, který je volán z přerušení. Rekurze (podprogram, který volá sám sebe) není zakázána; s jejím použitím byste však měli u podprogramů zacházet opatrně.

Tabulka 6-78 Operandy platné pro instrukci podprogramu

Vstupy/Výstupy	Typy dat	Operandy
SBR_N	WORD	Konstanta *pro CPU 221, CPU 222, CPU 224*: 0 až 63 *pro CPU 224XP a CPU 226* 0 až 127
IN	BOOL	V, I, Q, M, SM, S, T, C, L, signálový tok
	BYTE	VB, IB, QB, MB, SMB, SB, LB, AC, *VD, *LD, *AC[1], konstanta
	WORD, INT	VW, T, C, IW, QW, MW, SMW, SW, LW, AC, AIW, *VD, *LD, *AC[1], konstanta
	DWORD, DINT	VD, ID, QD, MD, SMD, SD, LD, AC, HC, *VD, *LD, *AC[1], &VB, &IB, &QB, &MB, &T, &C, &SB, &AI, &AQ, &SMB, konstanta
	ŘETĚZEC	*VD, *LD, *AC, konstanta
IN/OUT	BOOL	V, I, Q, M, SM[2], S, T, C, L
	BYTE	VB, IB, QB, MB, SMB[2], SB, LB, AC, *VD, *LD, *AC[1]
	WORD, INT	VW, T, C, IW, QW, MW, SMW[2], SW, LW, AC, *VD, *LD, *AC[1]
	DWORD, DINT	VD, ID, QD, MD, SMD[2], SD, LD, AC, *VD, *LD, *AC[1]
OUT	BOOL	V, I, Q, M, SM[2], S, T, C, L
	BYTE	VB, IB, QB, MB, SMB[2], SB, LB, AC, *VD, *LD, *AC[1]
	WORD, INT	VW, T, C, IW, QW, MW, SMW[2], SW, LW, AC, AQW, *VD, *LD, *AC[1]
	DWORD, DINT	VD, ID, QD, MD, SMD[2], SD, LD, AC, *VD, *LD, *AC[1]

1 Musí mít posunutí 1 nebo vyšší
2 Musí mít posunutí 30 nebo vyšší



Tip
STEP 7–Micro/WIN automaticky doplní nepodmíněný návrat z každého podprogramu.

Když je volán podprogram, uloží se celý logický zásobník, vrchol zásobníku se nastaví na jedna, všechna ostatní místa zásobníku se nastaví na nulu a řízení se přenese do volaného podprogramu. Pokud je tento podprogram ukončen, zásobník se obnoví hodnotami uloženými v místě volání a řízení se vrátí do volajícího programu.

Akumulátory jsou společné pro podprogramy a volající program. Při použití podprogramu se s nimi neprovádí operace uložení nebo obnovení.

Když je podprogram volán více než jednou ve stejném cyklu, neměly by se používat instrukce vzestupná hrana, sestupná hrana, časovač a čítač.

Volání podprogramu s parametry

Podprogramy mohou obsahovat odevzdávané parametry. Ty jsou definovány v tabulce lokálních proměnných podprogramu. Parametry musí mít symbolický název (maximálně 23 znaků), typ proměnné a typ dat. Do podprogramu nebo z podprogramu je možné odevzdávat až šestnáct parametrů.

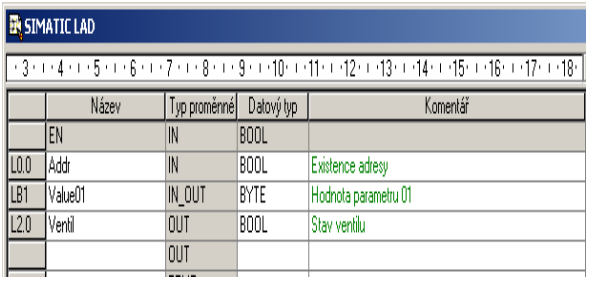
Pole typu proměnné v tabulce lokálních proměnných definuje, zda je proměnná odevzdávána do podprogramu (IN), odevzdávána do podprogramu i z něho (IN_OUT) nebo pouze odevzdávána z podprogramu (OUT). Tabulka 6-79 popisuje typy parametrů podprogramu. Chcete–li přidat další parametr, umístěte kurzor na pole typu proměnné (IN, IN_OUT nebo OUT), kterou chcete přidat. Klikněte pravým tlačítkem myši a zobrazí se menu voleb. Vyberte volbu ”Vložit” a pak ”Řádek pod”. Pod aktuální položkou se objeví další položka požadovaného typu.

Tabulka 6-79 Typy parametrů podprogramu

Parametr	Popis
IN	Parametry se odevzdávají do podprogramu. Pokud je parametrem přímá adresa (např. VB10), odevzdá se hodnota na daném místě odevzdá do podprogramu. Pokud je parametr nepřímá adresa (např. *AC1), odevzdá se hodnota do podprogramu na místo, na které ukazuje. Pokud je parametrem datová konstanta (16#1234) nebo adresa (&VB100), je do podprogramu odevzdána hodnota konstanty nebo adresy.
IN_OUT	Hodnota na dané adrese parametru se odevzdá do podprogramu, zatímco výsledná hodnota se z podprogramu vrátí na stejnou adresu. Konstanty (např. 16#1234) a adresy (např. &VB100) nejsou jako vstupní/výstupní parametry povoleny.
OUT	Výsledná hodnota se z podprogramu zapíše na danou adresu parametru. Konstanty (např. 16#1234) a adresy (např. &VB100) nejsou jako výstupní parametry povoleny. Protože výstupní parametry neuchovají hodnotu přiřazenou posledním provedením podprogramu, musíte výstupům přiřadit hodnoty po každém vyvolání podprogramu. Všimněte si, že instrukce SET a RESET ovlivňují pouze hodnotu booleovského operandu/operandů při zapnutém signálovém toku.
TEMP	Kterákoliv lokální paměť, která se nepoužívá pro předávání parametrů, může být použita pro dočasné uložení v rámci podprogramu.

Jak ukazuje obrázek 6-38, pole typu dat v tabulce lokálních proměnných definuje velikost a formát parametru. Typy parametrů je následující:

- BOOL: Tento typ dat se používá pro vstupy a výstupy o velikosti jednoho bitu. IN3 v následujícím příkladu je booleovský (bitový) vstup.
- BYTE, WORD, DWORD: Tyto typy dat označují vstupní nebo výstupní parametr bez znaménka o velikosti 1, 2 nebo 4 byty.
- INT, DINT: Tyto typy dat označují vstupní nebo výstupní parametr se znaménkem o velikosti 2 a 4 byty.



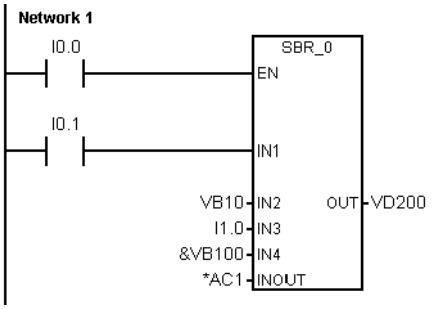
Obr. 6-38 Tabulka lokálních proměnných

- REAL: Tento typ dat označuje hodnotu s plovoucí desetinou čárkou a jednoduchou přesností (4 byty) podle IEEE.
- ŘETĚZEC: Tento typ dat se používá jako čtyřbytový ukazatel na řetězec.
- Signálový tok: Booleovský signálový tok je povolen pouze pro bitové (booleovské) vstupy. Z této deklarace vyplývá pro STEP 7-Micro/WIN, že tento vstupní parametr je výsledkem signálového toku založeného na kombinaci bitových logických instrukcí. Booleovské vstupy signálového toku se musí v tabulce lokálních proměnných objevit jako první před jakýmkoliv jiným typem vstupu. Tímto způsobem je možné použít pouze vstupní parametry. Povolovací vstup (EN) a vstup IN1 v následujícím příkladu používají booleovské typy parametrů.

Příklad: Volání podprogramu

Následují dva příklady v STL. První příklad může být zobrazen pouze editorem STL, protože parametry BOOL použité jako vstupy signálového toku nejsou uloženy do paměti L.

Druhý soubor instrukcí STL může být zobrazen také editory LAD a FBD, protože je použita paměť L pro uchování stavu vstupních BOOL parametrů, které jsou zobrazeny jako vstupy signálového toku v LAD a FBD.



Pouze STL:

Network 1

```
LD      I0.0
CALL    SBR_0, I0.1, VB10, I1.0, &VB100, *AC1, VD200
```

Pro zprávné zobrazení v LAD a FBD:

Network 1

```
LD      I0.0
=       L60,0
LD      I0.1
=       L63,7
LD      L60.0
CALL    SBR_0, L63.7, VB10, I1.0, &VB100, *AC1, VD200
```

Adresy jako parametry, například IN4 (&VB100), se do podprogramu odevzdávají jako hodnota DWORD (bez znaménka). Typ konstantního parametru musí být ještě před jeho použitím specifikován jako parametr ve volajícím podprogramu deskriptorem konstanty. Aby byla například konstanta double word bez znaménka o hodnotě 12.345 předána jako parametr, musí být konstantní parametr specifikován jako DW#12345. Pokud by se z parametru deskriptor konstanty vynechal, mohlo by se poedpokládat, že konstanta je jiný typ.

Se vstupními a výstupními parametry se neprovádějí žádné automatické konverze typu dat. Jestliže například tabulka lokálních proměnných specifikuje, že parametr má typ dat REAL, a ve volajícím podprogramu je pro tento parametr specifikováno double word (DWORD), bude hodnota v podprogramu double word.

Když se hodnoty odevzdávají do podprogramu, ukládají se do jeho lokální paměti. První sloupec tabulky lokálních proměnných uvádí adresu v lokální paměti pro každý předaný parametr. Když je podprogram volán, jsou hodnoty vstupních parametrů zkopírovány do jeho lokální paměti. Hodnoty výstupních parametrů jsou zkopírovány z lokální paměti podprogramu na specifikované adresy výstupních parametrů.

Velikost a typ datového prvku jsou dány parametrem. Přiřazení hodnot parametrů k lokální paměti v podprogramu je následující:

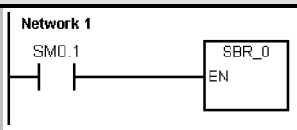
- Hodnoty parametrů jsou přiřazeny do lokální paměti v pořadí daném instrukcí volaného podprogramu s parametry začínajícími na L.0.
- Jedna až osm za sebou jdoucích hodnot bitových parametrů jsou přiřazeny jednomu bytu na adrese Lx.0 až Lx.7.
- Hodnoty byte, word a double word jsou zapsány do lokální paměti podle bytu (LBx, LWx nebo LDx).

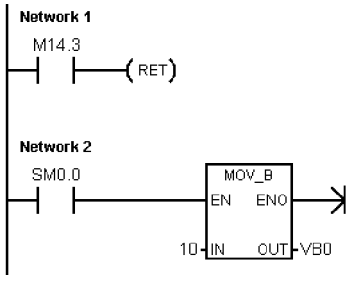
V případě instrukce ”Volání podprogramu” s parametry musí být parametry uspořádány tak, aby byly na prvním místě vstupní parametry, za nimi vstupní/výstupní parametry a po nich parametry výstupní.

Pokud programujete v STL, je formát instrukce CALL:

```
CALL    číslo podprogramu, parametr 1, parametr 2, ... , parametr
```

Příklad: Volání podprogramu a návratu z podprogramu

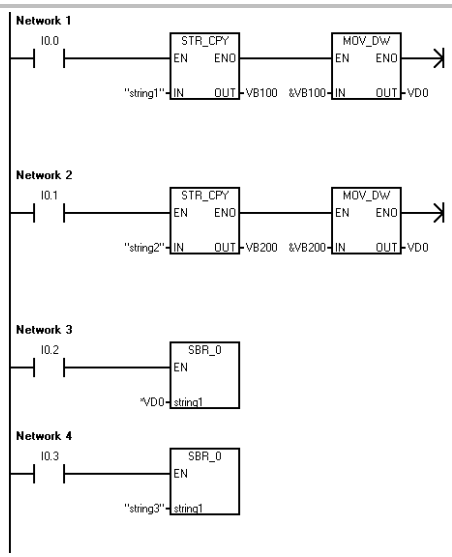
MAIN		Network 1 //V prvním programovém cyklu volá //podprogram 0 kvůli inicializaci. LD SM0.1 CALL SBR_0

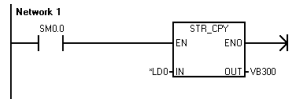
SBR0		Network 1 //Před posledním networkem můžete použít
		//podmíněný návrat pro opuštění
		//podprogramu.
		LD M14.3
		CRET
		Network 2 //Pokud je M14.3 zapnutý, bude tento network
		//přeskočen.
		LD SM0.0
		MOVB 10, VB0

Příklad: Volání podprogramu s řetězci

Tento příklad kopíruje různý řetězcový literál na jedinečnou adresu podle daného vstupu. Jedinečná adresa tohoto řetězce se uloží. Adresa řetězce se pak přenese do podprogramu pomocí nepřímé adresy. Typem dat vstupního parametru podprogramu je řetězec. Podprogram pak přesune řetězec na jiné místo.

Řetězcový literál lze také přesunout do podprogramu. Odkaz na řetězec v podprogramu je vždy stejný.

MAIN		Network 1 //
		LD I0.0
		SSCPY ”string1”, VB100
		AENO
		MOVD &VB100, VD0
		Network2 //
		LD I0.1
		SSCPY ”string2”, VB200
		AENO
		MOVD &VB200, VD0
		Network3 //
		LD I0.2
		CALL SBR_0, *VD0

SBR0		Network 1 // LD SM0.0 SSCPY *LD0, VB300

7

Síťová komunikace

Programovatelný automat S7-200 je určen k řešení potřeb uživatele v oblasti komunikace a sítí, a to nejen těch nejjednodušších, ale i složitějších. S7-200 je vybaven funkcemi, které umožňují komunikovat s jinými zařízeními, jako jsou například tiskárny a váhy, které užívají své vlastní komunikační protokoly.

Díky STEP 7–Micro/WIN je nastavení a konfigurace sítě jednoduché a snadné.

Přehled kapitoly

Vysvětlení základů síťové komunikace S7-200 210
Výběr síťového komunikačního protokolu 214
Instalace a odstranění komunikačních rozhraní 220
Návrh sítě 221
Vytvoření protokolů definovaných uživatelem v režimu Freeport 225
Použití modemu a STEP 7–Micro/WIN 228
Témata pro pokročilé 233
Konfigurace RS-232/PPI Multi-Master kabelu pro dálkové ovládání 239

Vysvětlení základů síťové komunikace S7-200

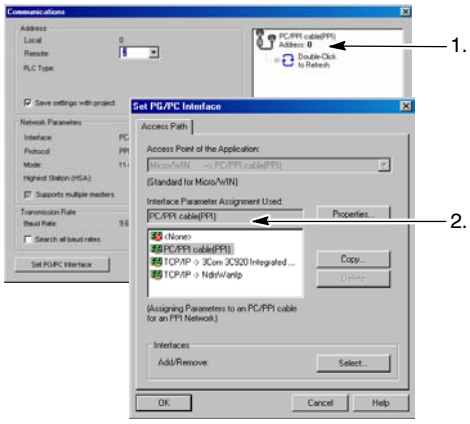
Výběr síťového komunikačního rozhraní

Automat S7-200 podporuje mnoho různých typů komunikačních sítí. Výběr sítě se provádí v dialogovém okně "Nastavit vlastnosti rozhraní PG/PC". Vybraná síť je označována jako rozhraní. Různé typy rozhraní, které jsou k dispozici pro přístup k těmto komunikačním sítím, jsou:

- PPI Multi-Master kabely
- CP komunikační karty
- Ethernetové komunikační karty

Chcete–li vybrat komunikační rozhraní pro STEP 7–Micro/WIN, proveďte následující kroky. Viz obrázek 7-1.

1. Dvakrát klikněte na ikonu v okně "Nastavení komunikace".
2. Vyberte parametr rozhraní pro STEP 7–Micro/WIN.



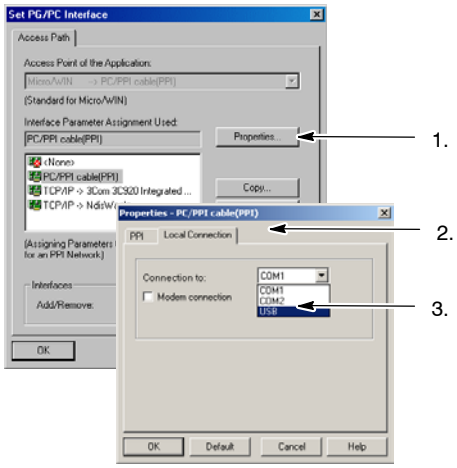
Obr. 7-1 Komunikační rozhraní STEP 7–Micro/WIN

PPI Multi-Master kabely

Automat S7-200 podporuje komunikaci pomocí dvou různých typů PPI Multi-Master kabelů. Tyto typy kabelů umožňují komunikaci přes rozhraní RS-232 nebo rozhraní USB.

Jak je vidět na obrázku 7-2, výběr typu PPI Multi-Master kabelu je jednoduchý. Proveďte následující kroky:

1. Na stránce "Nastavení vlastností rozhraní PG/PC" klikněte na tlačítko "Vlastnosti".
2. Na stránce "Vlastnosti" klikněte na záložku "Místní připojení".
3. Vyberte USB nebo požadovaný port COM.



Obr. 7-2 Výběr PPI Multi-Master kabelu

Tip
Vezměte prosím na vědomí, že v jednom případě je možné použít pouze jeden USB kabel.

Tip
Příklady v tomto manuálu používají RS-232/PPI Multi-Master kabel. RS-232/PPI Multi-Master kabel nahrazuje dřívější PC/PPI kabel. Nabízíme také USB/PPI Multi-Master kabel. Objednací čísla – viz přílohu NO TAG.

Použití zařízení typu master a slave

Síť S7-200 pracuje v režimu master-slave, což znamená, že může fungovat jako master i jako slave v síti PROFIBUS, přičemž STEP 7 – Micro/WIN je vždy master.

Zařízení typu master

Zařízení, které v síti pracuje jako master, může generovat požadavek na jiné síťové zařízení. Master může také reagovat na požadavky jiných masterů, zapojených v síti. Typickými představiteli zařízení typu master jsou STEP 7 – Micro/WIN, zařízení pro rozhraní člověk – stroj, jako je např. TD 200 a programovatelné automaty S7-300 nebo S7-400. Programovatelný automat S7-200 funguje jako master tehdy, požaduje – li informace od jiného S7-200 (komunikace peer-to-peer).



Tip

TP070 nebude v síti spolupracovat s jiným zařízením typu master.

Zařízení typu slave

Zařízení, které je konfigurováno jako slave, může odpovídat pouze na požadavky masteru. Slave žádost nikdy negeneruje. Ve většině sítí funguje automat S7-200 jako slave. Ve funkci slave odpovídá S7-200 na požadavky masteru, jako je například operátorský panel nebo STEP 7 – Micro/WIN.

Nastavení přenosové rychlosti a síťové adresy

Rychlost, kterou jsou data přenášena po síti, se nazývá přenosová rychlost; měří se v kilobaudech (kbaud) nebo megabaudech (Mbaud). Přenosová rychlost udává, jaké množství dat je možné přenést za daný časový interval. Například přenosová rychlost 19,2 kbaud znamená přenos 19.200 bitů za sekundu.

Všechna zařízení, která komunikují prostřednictvím dané sítě, musí být konfigurována tak, aby přenášela data stejnou přenosovou rychlostí. Proto je nejvyšší přenosová rychlost dána nejpomalejším zařízením, připojeným do síti.

Tabulka 7-1 uvádí přenosové rychlosti podporované automatem S7-200.

Síťová adresa je jedinečné číslo, které přidělíte každému zařízení v síti. Adresa zajišťuje, aby byla data přenášena nebo vyhledávána ve správném zařízení. Programovatelný automat S7-200 podporuje síťové adresy od 0 do 126. V případě S7-200 se dvěma porty má každý z nich vlastní síťovou adresu. Tabulka 7-2 uvádí implicitní nastavení adres pro automaty S7-200.

Tabulka 7-1 Přenosové rychlosti podporované S7-200

Síť	Přenosová rychlost
Standardní síť	9,6 kbaud až 187,5 kbaud
S použitím modulu EM 277	9,6 kbaud až 12 Mbaud
Režim Freeport	1200 baud až 115,2 kbaud

Tabulka 7-2 Implicitně nastavené adresy automatů S7-200

Programovatelný automat S7-200	Implicitně nastavená adresa
STEP 7 – Micro/WIN	0
HMI (TD 200, TP nebo OP)	1
S7-200 CPU	2

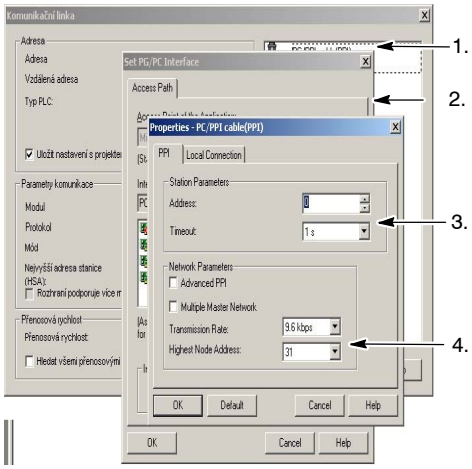
Nastavení přenosové rychlosti a síťové adresy pro STEP 7–Micro/WIN

Pro STEP 7–Micro/WIN je třeba nastavit adresu a přenosovou adresu. Přenosová rychlost musí být stejná, jako mají ostatní zařízení v síti, síťová adresa musí být jedinečná.

Běžně není nutné síťovou adresu STEP 7 Micro/WIN (0) měnit. Pokud ale síť obsahuje jiný programovací SW, může být nutné změnit síťovou adresu STEP 7–Micro/WIN.

Jak je vidět z obrázku 7-3, nastavení přenosové rychlosti a síťové adresy pro STEP 7–Micro/WIN je jednoduché. Nejprve klikněte na ikonu ”Komunikace” v menu ”Navigace” a proveďte následující kroky:

1. Dvakrát klikněte na ikonu v okně ”Nastavení komunikace”.
2. V dialogovém okně ”Nastavení rozhraní PG/PC” klikněte na tlačítko ”Vlastnosti”.
3. Vyberte síťovou adresu pro STEP 7–Micro/WIN.
4. Vyberte přenosovou rychlost pro STEP 7–Micro/WIN.



Obr. 7-3 Nastavení STEP 7–Micro/WIN

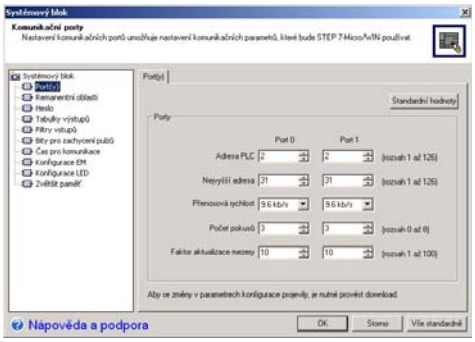
Nastavení přenosové rychlosti a síťové adresy pro S7-200

Přenosovou rychlost a síťovou adresu je nutné nastavit také pro S7-200. Rychlost i adresa jsou uloženy v systémovém bloku S7-200. Po nastavení parametrů pro S7-200 musíte systémový blok do S7-200 downloadovat.

Implicitně nastavená rychlost portů S7-200 je 9,6 kbaud, implicitní síťová adresa je 2.

Jak je vidět na obrázku 7-4, pro nastavení přenosové rychlosti a síťové adresy S7-200 se používá STEP 7–Micro/WIN. Nejprve vyberte ikonu ”Systémový blok” nebo vyberte příkaz menu **Zobrazit > Komponenty > Systémový Blok** a proveďte následující kroky:

1. Vyberte síťovou adresu.
2. Vyberte přenosovou rychlost.
3. Proveďte download systémového bloku.



Obr. 7-4 Nastavení CPU S7-200

Tip

Je povolen výběr všech volitelných přenosových rychlostí. STEP 7–Micro/WIN potvrdí platnost tohoto výběru během downloadu systémového bloku. Download výběru přenosových rychlostí, které by bránily komunikaci STEP 7–Micro/WIN s automatem S7-200, není proveden.

Nastavení vzdálené adresy

Před downloadem nového nastavení do S7-200 musíte nastavit jak komunikační (COM) port STEP 7–Micro/WIN (lokální), tak adresu S7-200 (vzdálenou) tak, aby odpovídaly aktuálnímu nastavení vzdáleného S7-200. Viz obrázek 7-5.

Po downloadu aktualizovaných nastavení může být nutné rekonfigurovat nastavení přenosové rychlosti rozhraní PG/PC (pokud se liší od nastavení použitého při downloadu vzdáleného S7-200). Konfiguraci přenosové rychlosti najdete na obr. 7-3.



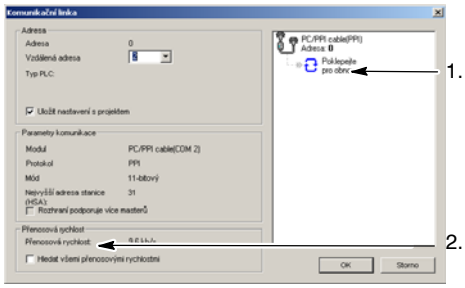
Obr. 7-5 Nastavení STEP 7–Micro/WIN

Vyhledávání CPU S7-200 v síti

CPU S7-200, které jsou k síti připojeny, je možné vyhledávat a identifikovat. Při vyhledávání je možné prohledávat síť určenou buď přenosovou rychlostí, nebo všemi přenosovými rychlostmi.

Pouze PPI Multi-Master kabely umožňují vyhledávání všech přenosových rychlostí. Tato funkce není k dispozici, jestliže komunikujete prostřednictvím CP karty. Vyhledávání začíná aktuálně zvolenou přenosovou rychlostí.

1. Chcete–li zahájit vyhledávání, otevřete dialogové okno ”Komunikace” a dvakrát klikněte na ikonu ”Obnovit”.
2. Pro vyhledávání ve všech přenosových rychlostech vyberte zaškrtávací políčko ”Hledat všemi přenosovými rychlostmi”.



Obr. 7-6 Vyhledávání CPU v síti

Výběr síťového komunikačního protokolu

Následuje přehled protokolů podporovaných CPU S7-200.

- ❏ Dvoubodové spojení (PPI)
- ❏ Mnohobodové spojení (MPI)
- ❏ PROFIBUS

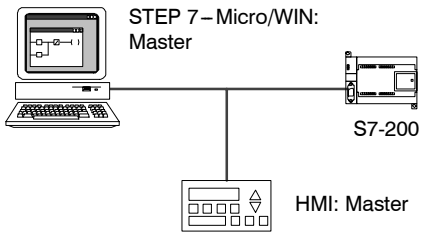
Tyto protokoly jsou založeny na sedmivrstvové architektuře (OSI). Jsou implementovány v síti s kruhovou topologií, která odpovídá standardu PROFIBUS tak, jak je definován evropskou normou EN 50170. Tyto protokoly jsou asynchronní, znakově orientované s jedním start bitem, osmi datovými bity, sudou paritou a jedním stop bitem. Přenosové rámce obsahují speciální znaky start a stop, adresy zdrojové a cílové stanice, délky u rámce a kontrolním součet. Pokud mají všechny protokoly stejnou přenosovou rychlost, mohou v síti běžet, aniž by se vzájemně rušily.

Pro CPU S7-200 s rozšiřovacími moduly CP243–1 a CP243–1IT je k dispozici i Ethernet.

Protokol PPI

PPI je protokol typu master-slave: zařízení master posílají požadavky zařízením slave, které jim odpovídají. Viz obrázek 7-7. Zařízení slave negenerují zprávy, ale čekají, dokud master nepošle požadavek nebo je nevyzve k odpovědi.

Zařízení master komunikují se zařízeními slave prostřednictvím sdíleného spojení, které je řízeno protokolem PPI. Tento protokol neomezuje počet masterů, které mohou komunikovat s kterýmkoliv zařízením slave; není však možné instalovat více než 32 masterů.



Obr. 7-7 Síť PPI

CPU S7-200 se mohou v režimu RUN chovat jako master, pokud v uživatelském programu povolíte režim PPI master. (Viz popis SMB30 v příloze D.) Po povolení režimu PPI master můžete použít instrukce "Čtení ze sítě" pro čtení nebo "Čtení do sítě" pro zápis do jiných jednotek S7-200. I když pracuje automat jako PPI master, stále reaguje jako zařízení typu slave na požadavky jiných masterů.

PPI Advanced umožňuje navázat vzájemné logické spojení mezi zařízeními. V případě PPI Advanced má každé zařízení omezený počet připojení. Tabulka 7-3 uvádí počet připojení podporovaných S7-200.

Všechny CPU S7-200 podporují protokol PPI i PPI Advanced, ale PPI Advanced je jediný protokol PPI podporovaný modulem EM 277.

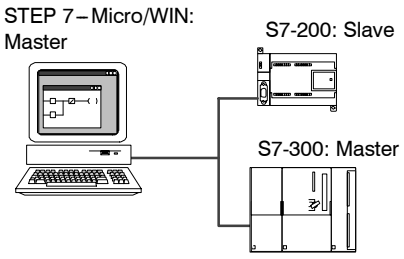
Tabulka 7-3 Počet spojení CPU S7-200 a modulů EM 277

Modul		Přenosová rychlost	Počet spojení
CPU S7-200	Port 0	9,6 kbaud, 19,2 kbaud nebo 187,5 kbaud	4
	Port 1	9,6 kbaud, 19,2 kbaud nebo 187,5 kbaud	4
Modul EM 277		9,6 kbaud až 12 Mbaud	6 na modul

Protokol MPI

Protokol MPI umožňuje komunikaci master – master i master – slave. Viz obrázek 7-8. Pro komunikaci s CPU S7-200 naváže STEP 7 – Micro/WIN spojení master – slave. Protokol MPI nepodporuje CPU S7-200 ve funkci master.

Síťová zařízení spolu komunikují prostřednictvím oddělených spojení (řízených protokolem MPI) mezi libovolnými dvěma zařízeními. Komunikace mezi zařízeními je omezena na počet spojení podporovaných CPU S7-200 nebo moduly EM 277. Počet spojení, podporovaných S7-200, uvádí tabulka 7-3.



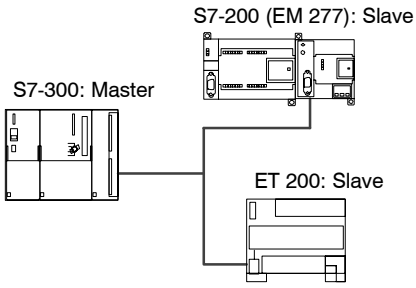
Obr. 7-8 Síť MPI

Pro čtení a zápis dat používají programovatelné automaty S7-300 a S7-400 v protokolu MPI instrukce XGET a XPUT. Informace o těchto instrukcích najdete v příručce pro programování S7-300 nebo S7-400.

Protokol PROFIBUS

Protokol PROFIBUS je určen pro vysokorychlostní komunikaci s distribuovanými vstupními/výstupními zařízeními (vzdálené vstupy/výstupy). Mnoho různých výrobců nabízí řadu zařízení PROFIBUS. Tato zařízení se pohybují od jednoduchých vstupních nebo výstupních modulů až po regulátory motorů a programovatelné automaty.

Sítě PROFIBUS běžně obsahují jedno zařízení typu master a několik vstupních/výstupních zařízení typu slave. Viz obrázek 7-9. Zařízení typu master je nakonfigurováno tak, že zná, jaké typy vstupních/výstupních zařízení typu slave jsou k němu připojeny a jaké mají adresy. Master inicializuje síť a ověřuje, zda zařízení typu slave, zapojená do sítě, odpovídají konfiguraci. Master nepřetržitě zapisuje výstupní data do zařízení typu slave a čte z nich data vstupní.



Obr. 7-9 Síť PROFIBUS

Po úspěšné konfiguraci zařízení typu slave DP masterem je slave přiřazen určitému masteru. Pokud je v síti druhý master, má jen velmi omezený přístup k zařízením typu slave přiřazeným prvnímu masteru.

Protokol TCP/IP

Automat S7-200 může podporovat ethernetovou komunikaci TCP/IP použitím rozšiřovacích modulů pro Ethernet (CP 243 – 1) nebo Internet (CP 243 – 1 IT). Tabulka 7-4 ukazuje přenosové rychlosti a počet připojení podporovaných těmito moduly.

Tabulka 7-4 Počet spojení ethernetových (CP 243 – 1) a internetových (CP 243 – 1 IT) modulů

Modul	Přenosová rychlost	Počet spojení
Ethernetový (CP 243 – 1) modul	10 až 100 Mbaud	8 univerzálních připojení
Internetový (CP 243 – 1 IT) modul		1 připojení STEP7 – Micro/WIN1

Další informace najdete v *Manuálu komunikačního procesoru SIMATIC NET CP 243 – 1 pro průmyslový Ethernet* nebo v *Manuálu komunikačního procesoru SIMATIC NETCP 243 – 1 IT pro průmyslový Ethernet a informační technologie.*

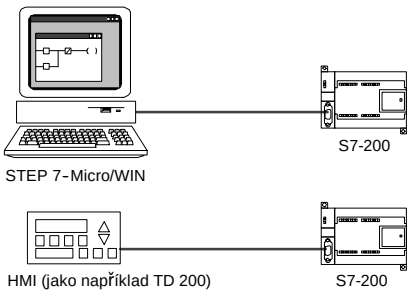
Vzorové konfigurace sítě s použitím pouze automatů S7-200

Síť PPI s jedním masterem

V síti s jedním masterem jsou počítač a CPU S7-200 spojeny buď kabelem PPI Multi-Master, nebo kartou komunikačního procesoru (CP karta) instalovanou v počítači.

Na obrázku 7-10 nahoře je počítač (STEP 7–Micro/WIN) master této sítě. Na obrázku 7-10 dole funguje jako master zařízení pro rozhraní člověk – stroj (HMI) (jako například TD 200, TP nebo OP).

V obou vzorových příkladech pracuje CPU S7-200 jako slave, který reaguje na požadavky masteru.



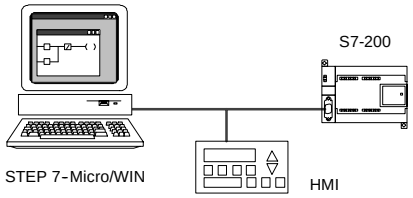
Obr. 7-10 Síť PPI s jedním masterem

V případě použití sítě PPI s jedním masterem konfigurujete STEP 7–Micro/WIN pro používání protokolu PPI. Pokud jsou dostupná, zrušte zaškrtnutí políček "Síť s více mastery" a "PPI Advanced".

Síť PPI s více mastery

Na obrázku 7-11 je příklad sítě obsahující více zařízení typu master a jedno typu slave. Počítač (STEP 7–Micro/WIN) používá CP kartu nebo PPI Multi-Master kabel. STEP 7–Micro/WIN a zařízení HMI síť sdílejí.

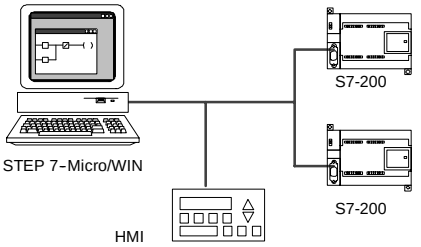
Jak STEP 7–Micro/WIN, tak zařízení HMI jsou zařízení typu master a musí mít svoje vlastní síťové adresy. Použitý PPI Multi-Master kabel je master a používá síťovou adresu poskytnutou STEP 7–Micro/WIN. CPU S7-200 je zařízení typu slave.



Obr. 7-11 Více masterů s jedním slave

Na obrázku 7-12 je síť PPI, v níž více masterů komunikuje s více zařízeními typu slave. V tomto příkladu může jak STEP 7–Micro/WIN, tak HMI požadovat data od jakékoliv CPU S7-200 ve funkci slave. STEP 7–Micro/WIN a zařízení HMI síť sdílejí.

Všechna zařízení (typu master i slave) mají různé adresy. Pokud použijete kabel PPI Multi-Master, tak kabel je master a používá síťovou adresu poskytnutou STEP 7–Micro/WIN. CPU S7-200 jsou zařízení typu slave.



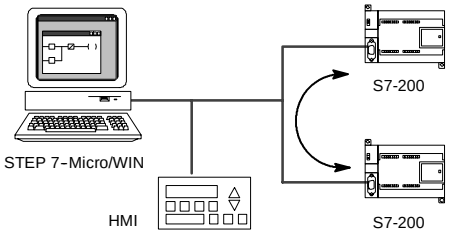
Obr. 7-12 Více zařízení typu master i slave

V síti s více mastery a jedním nebo více zařízeními typu slave konfigurujte STEP 7–Micro/WIN na užívání protokolu PPI a zaškrtněte políčka "Síť s více mastery" a "PPI Advanced", pokud jsou dostupná. Pokud užíváte PPI Multi-Master kabel, políčka "Síť s více mastery" a "PPI Advanced" se ignorují.

Složité sítě PPI

Na obrázku 7-13 je vidět příklad sítě, která používá více zařízení typu master pro dvoubodovou komunikaci (peer-to-peer) .

STEP 7–Micro/WIN a zařízení HMI čtou a zapisují data do CPU S7-200; CPU S7-200 používají instrukce "Čtení ze sítě" a "Zápis do sítě" pro vzájemné čtení a zápis (dvoubodová komunikace peer-to-peer).

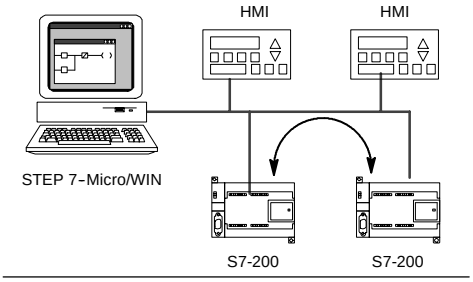


Obr. 7-13 Komunikace peer-to-peer

Na obrázku 7-14 vidíme další příklad složité sítě PPI, která používá více masterů pro dvoubodovou komunikaci (peer-to-peer). V tomto příkladu každé HMI monitoruje jedno CPU S7-200.

CPU S7-200 používá pro vzájemné čtení a zápis instrukce NETR a NETW (komunikace peer-to-peer).

Ve složitých sítích PPI konfigurujte STEP 7–Micro/WIN na užívání protokolu PPI a zaškrtněte políčka "Síť s více mastery" a "PPI Advanced", pokud jsou dostupná. Pokud užíváte PPI Multi-Master kabel, políčka "Síť s více mastery" a "PPI Advanced" se ignorují.



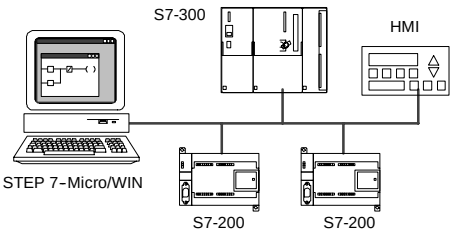
Obr. 7-14 Zařízení HMI a komunikace peer-to-peer

Příklady konfigurací sítě s použitím S7-200, S7-300 a S7-400

Sítě s přenosovou rychlostí až do 187,5 kbaud

U příkladu sítě na obrázku 7-15 používá S7-300 pro komunikaci s CPU S7-200 instrukce XPUT a XGET. S7-300 nemůže komunikovat s CPU S7-200 v režimu master.

Pro komunikaci s CPU S7 konfigurujte STEP 7–Micro/WIN na užívání protokolu PPI a zaškrtněte políčka "Síť s více mastery" a "PPI Advanced", pokud jsou dostupná. Pokud užíváte PPI Multi-Master kabel, políčka "Síť s více mastery" a "PPI Advanced" se ignorují.



Obr. 7-15 Rychlost přenosu až do 187,5 kbaud

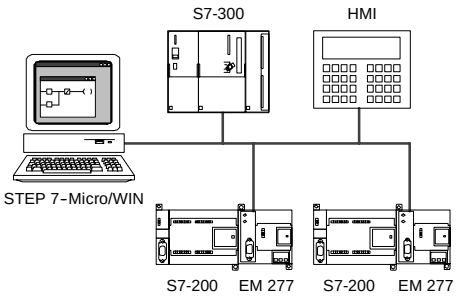
Sítě s přenosovou rychlostí nad 187,5 kbaud

Pro přenosové rychlosti vyšší než 187,5 kbaud musí CPU S7-200 použít pro připojení k síti modul EM 277. Viz obrázek 7-16. STEP 7–Micro/WIN musí být připojen kartou komunikačního procesoru (CP).

V této konfiguraci může S7-300 komunikovat s automaty S7-200 pomocí instrukcí XPUT a XGET a HMI může monitorovat buď automaty S7-200, nebo S7-300.

EM 277 je vždy zařízení typu slave.

STEP 7–Micro/WIN může programovat nebo monitorovat kterýkoliv CPU S7-200 prostřednictvím připojeného EM 277. Chcete–li s EM 277 komunikovat rychlostí vyšší než 187,5 kbaud, konfigurujte STEP 7–Micro/WIN na užívání protokolu MPI s CP kartou. Maximální přenosová rychlost PPI Multi-Master kabelů je 187,5 kbaud.



Obr. 7-16 Rychlost přenosu nad 187,5 kbaud

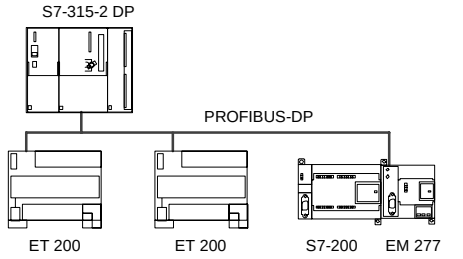
Příklady konfigurací sítě PROFIBUS-DP

Sítě s S7-315–2 DP jako zařízením PROFIBUS master a EM 277 jako PROFIBUS slave

Na obrázku 7-17 je příklad sítě PROFIBUS, která používá S7-315–2 DP jako PROFIBUS master. Modul EM 277 je PROFIBUS slave.

S7-315–2 DP může z EM 277 data číst nebo zapisovat v rozsahu od 1bytu až do 128 bytů. S7-315–2 DP používá pro čtení a zápis paměti V automatu S7-200.

Tato síť podporuje přenosové rychlosti od 9600 baud do 12 Mbaud.

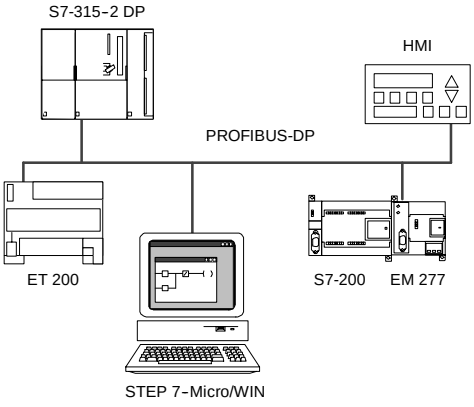


Obr. 7-17 Síť s S7-315–2 DP

Sítě se STEP 7–Micro/WIN a HMI

Na obrázku 7-18 je příklad sítě s S7-315–2 DP jako PROFIBUS master a EM 277 jako PROFIBUS slave. V této konfiguraci monitoruje HMI automat S7-200 přes EM 277. STEP 7–Micro/WIN programuje S7-200 přes EM 277.

Tato síť podporuje přenosové rychlosti od 9600 baud do 12 Mbaud. STEP 7–Micro/WIN vyžaduje pro přenosové rychlosti vyšší než 187,5 kbaud kartu CP.



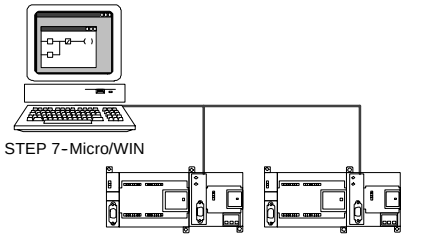
Obr. 7-18 Síť PROFIBUS

STEP7–Micro/WIN nakonfigurujte tak, aby pro kartu CP používal protokol PROFIBUS. Pokud jsou v síti pouze zařízení DP, vyberte profil DP nebo Standard. Vyberte pro všechny mastery profil Univerzální (DP/FMS), pokud jsou v síti jakákoliv zařízení jiného typu než DP, jako například TD 200. Aby síť fungovala, musí být všechny mastery v síti nastaveny na užívání stejného profilu PROFIBUS (DP, Standard nebo Univerzální).

PPI Multi-master kabely budou fungovat v sítích s přenosovou rychlostí do 187,5 kbaud pouze tehdy, když všechny mastery používají profil Univerzální (DP/FMS).

Vzorové konfigurace sítě s použitím ethernetových a/nebo internetových zařízení

V konfiguraci zobrazené na obrázku 7-19 je ethernetové spojení použito, aby umožnilo STEP 7–Micro/WIN komunikovat s kterýmkoliv CPU S7-200, který používá ethernetový (CP 243–1) nebo internetový (CP 243–1 IT) modul. CPU S7-200 si mohou prostřednictvím ethernetového spojení vzájemně vyměňovat data. Pro přístup na domovskou stránku internetového modulu (CP 243–1 IT) je možné použít standardní prohlížeč běžící na PC se STEP 7–Micro/WIN.



Obr. 7-19 Ethernetová síť 10/100 Mbaud

V ethernetových sítích konfigurujte STEP 7–Micro/WIN na užívání protokolu TCP/IP.



Tip

V dialogovém okně ”Nastavit rozhraní PG/PC” jsou nejméně dvě volby TCP/IP. Volbu TCP/IP –> NdisWanIp S7-200 nepodporuje.

- ❑ V dialogovém okně ”Nastavit rozhraní PG/PC” závisí volba (volby) na typu ethernetového rozhraní ve vašem PC. Zvolte to, které spojuje váš počítač s ethernetovou sítí, ve které je zapojen modul CP 243–1 nebo CP 243–1 IT.
- ❑ V dialogovém okně ”Komunikace” musíte vložit vzdálenou IP adresu (Vzdálené IP adresy) každého ethernetového/internetového modulu, se kterým si přejete komunikovat.

Instalace a odstranění komunikačních rozhraní

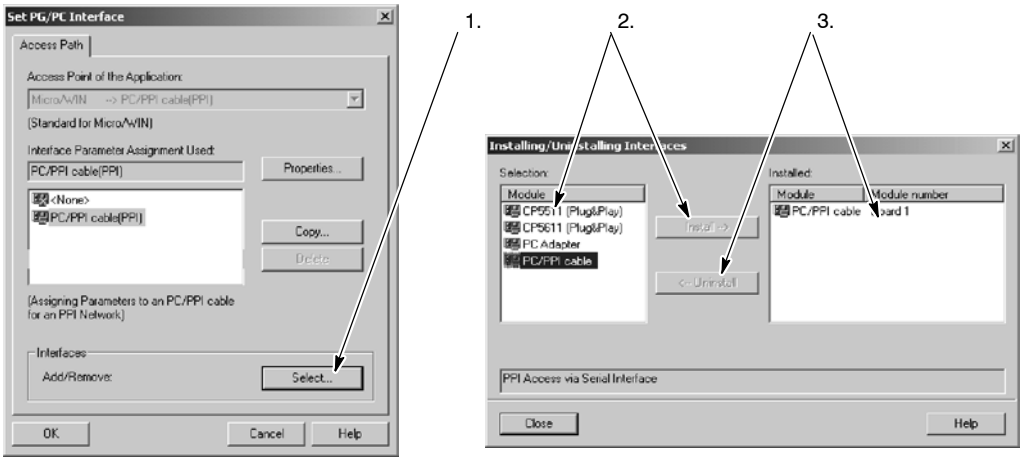
Z dialogového okna ”Nastavení rozhraní PG/PC” použijte pro instalaci nebo odstranění komunikačních rozhraní dialogové okno ”Instalace/odstranění rozhraní”

1. V dialogovém okně ”Nastavení rozhraní PG/PC” klikněte na ”Vybrat”; objeví se dialogové okno ”Instalace/odstranění rozhraní”.

   V poli ”Výběr” je seznam dostupných rozhraní, v poli ”Instalováno” jsou uvedena rozhraní, která už jsou v počítači instalována.

2. *Chcete–li přidat komunikační rozhraní:* Vyberte komunikační hardware a klikněte na ”Instalovat”. Po zavření dialogového okna ”Instalace/odstranění rozhraní” se zobrazí dialogové okno ”Nastavení rozhraní PG/PC” daného rozhraní v poli ”Použité přiřazení parametru rozhraní”.

3. *Chcete–li odstranit komunikační rozhraní:* Vyberte rozhraní, které má být odstraněno, a klikněte na ”Odstranit”. Po zavření dialogového okna ”Instalace/odstranění rozhraní”se dané rozhraní odstraní z pole ”Použité přiřazení parametru rozhraní” v dialogovém okně ”Nastavení rozhraní PG/PC”.



Obr. 7-20 Nastavení rozhraní PG/PC a instalace/odstranění dialogových oken pro rozhraní

Úprava nastavení portů počítače pro PPI Multi-Master

Pokud používáte USB/PPI Multi-Master kabel nebo RS-232/PPI Multi-Master kabel v režimu PPI, není nutné upravovat nastavení portu na vašem počítači a je možný provoz sítě s více mastery s operačním systémem Windows NT.

Pokud použijete RS-232/PPI Multi-Master kabel v režimu PPI/Freeport pro komunikaci mezi CPU S7-200 a STEP 7–Micro/WIN v operačním systému, který podporuje konfiguraci PPI Multi-Master (Windows NT PPI Multi-Master nepodporuje), může být zapotřebí upravit nastavení portu na počítači:

1. Klikněte pravým tlačítkem myši na ikonu ”Tento počítač” na ploše a vyberte ”Vlastnosti”.
2. Vyberte záložku ”Správce zařízení”. U Windows 2000 nejprve vyberte záložku ”Hardware”, pak tlačítko ”Správce zařízení”.
3. Dvakrát klikněte na ”Porty” (COM & LPT).
4. Vyberte komunikační port, který aktuálně používáte (například COM1).
5. V záložce ”Nastavení portu” klikněte na tlačítko ”Upřesnit”.
6. Nastavte ovládací prvky pro ”Přijímací vyrovnávací paměť” a ”Přenosová vyrovnávací paměť” na nejnižší hodnotu (1).
7. Klikněte na ”OK”, zavřete všechna okna a počítač restartujte, aby se nové nastavení projevilo.

Návrh sítě

Obecné pokyny

Vždy instalujte přepěťovou ochranu pro všechny okruhy, které by mohly být zasaženy bleskem.

Neumísťujte nízkonapěťové signálové vodiče a komunikační kabely do stejné lišty s napájecími kabely střídavého napětí a vysokonapěťovými vodiči stejnosměrného napětí, které jsou rychle spínány. Vždy pokládejte vodiče v párech s nulovým nebo společným vodičem, spárovaným s živým nebo signálovým vodičem.

Komunikační port CPU S7-200 není napěťově oddělen. Abyste zajistili oddělení sítě, zvažte použití repeateru RS-485 nebo modulu EM 277.

Upozornění

Jestliže propojíte zařízení s různými referenčními potenciály, mohou spojovacím kabelem protékat nežádoucí proudy.

Tyto nežádoucí proudy mohou způsobit chyby při komunikaci nebo poškodit zařízení.

Aby se zabránilo průtoku nežádoucích proudů, zkontrolujte, zda všechny části zařízení, které chcete propojit komunikačním kabelem, mají buď společný referenční potenciál, nebo jsou napěťově odděleny, aby se zabránilo průtoku nežádoucích proudů. Více informací o uzemnění a referenčních bodech oddělených obvodů najdete v kapitole 3.

Stanovení vzdáleností, přenosové rychlosti a kabelů

Jak je vidět z tabulky 7-5, maximální délka segmentu sítě je dána dvěma faktory: napěťovým oddělením (použití repeateru RS-485) a přenosovou rychlostí.

Jestliže spojujete zařízení s různými potenciály země, je nutné je galvanicky oddělit. Tyto různé potenciály se mohou vyskytnout v případě, kdy jsou země fyzicky odděleny dlouhou vzdáleností. Na krátkých vzdálenostech mohou rozdíl potenciálů způsobit také napájecí proudy velkých strojů.

Tabulka 7-5 Maximální délka kabelu v síti

Přenosová rychlost	Neizolovaný port CPU[1]	Port CPU s repeaterem nebo modulem EM 277
9,6 kbaud až 187,5 kbaud	50 m	1 000 m
500 kbaud	Nepodporuje	400 m
1 Mbaud až 1,5 Mbaud	Nepodporuje	200 m
3 Mbaud až 12 Mbaud	Nepodporuje	100 m

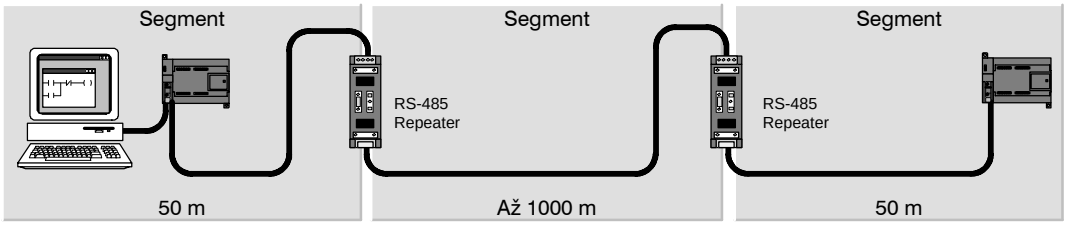
[1] Maximální povolená vzdálenost bez použití oddělení nebo repeateru je 50 m. Tato vzdálenost se měří od prvního k poslednímu uzlu v segmentu.

Použití repeaterů

Repeater RS-485 zajišťuje symetrizaci a ukončení segmentů sítě. Je možné ho použít k následujícím účelům:

- *Pro prodloužení délky sítě:* Přidání repeateru do sítě umožní její prodloužení o dalších 50 m. Pokud propojíte dva repeatery, mezi nimiž neleží žádné jiné uzly (jak je to znázorněno na obrázku 7-21), můžete síť prodloužit na maximální délku kabelu pro danou přenosovou rychlost. Do sítě můžete sériově zapojit až 9 repeaterů, ale celková délka sítě nesmí přesáhnout 9600 m.
- *Pro přidání zařízení do sítě:* Každý segment může mít maximálně 32 zařízení, propojených do vzdálenosti 50 m při rychlosti 9600 baud. Použití repeateru umožní přidat do sítě další segment (32 zařízení).
- *Pro oddělení různých segmentů sítě:* Oddělení sítě zlepšuje kvalitu přenosu tím, že oddělí síťové segmenty, které mohou mít různé zemní potenciály.

Repeater se v síti počítá za jeden z uzlů v segmentu, i když nemá přiřazenou síťovou adresu.



Obr. 7-21 Příklad sítě s repeaterem

Výběr síťového kabelu

Sítě S7-200 používají pro standard RS-485 kroucenou dvojlinku. Tabulka 7-6 uvádí požadované vlastnosti kabelu. V síťovém segmentu může být zapojeno až 32 zařízení.

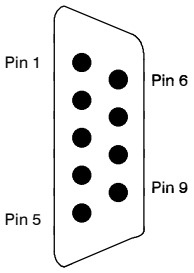
Tabulka 7-6 Obecné vlastnosti kabelů

Vlastnosti	Popis
Typ kabelu	Stíněná kroucená dvojlinka
Odpor smyčky	≤115 Ω/km
Kapacita	30 pF/m
Jmenovitá impedance	Přibližně 135 Ω až 160 Ω *(kmitočet = 3 MHz až 20 MHz)*
Útlum	0,9 dB/100 m *(kmitočet = 200 kHz)*
Průřez kabelu	0,3 mm^2 až 0,5 mm^2
Průměr kabelu	8 mm ∂0,5 mm

Rozmístění vývodů konektoru

Komunikační porty na CPU S7-200 jsou kompatibilní s RS-485 a jsou realizovány jako miniaturní konektory typu D s devíti vývody podle standardu PROFIBUS tak, jak je definován v evropské normě EN 50170. Tabulka 7-7 ukazuje konektor, který zajišťuje fyzické spojení komunikačního portu, a popisuje rozmístění vývodů komunikačního portu.

Tabulka 7-7 Rozmístění vývodů v komunikačním portu automatu S7-200

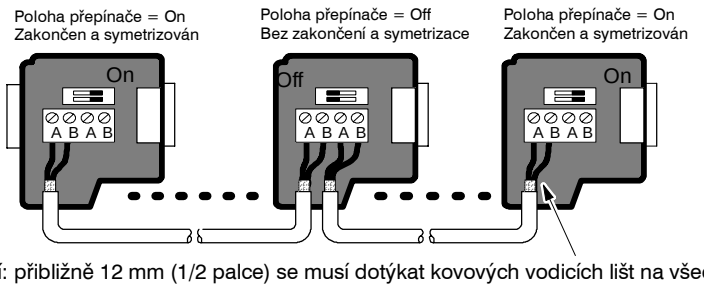
Konektor	Číslo vývodu	PROFIBUS Signal	Port 0/Port 1
	1	Stínění	Uzemnění rozvaděče
	2	24 V zpětný	Logický společný
	3	RS-485 signál B	RS-485 signál B
	4	Požadavek na vysílání	RTS (TTL)
	5	5 V zpětný	Logický společný
	6	+5 V	+5 V, 100 Ω sériový odpor
	7	+24 V	+24 V
	8	RS-485 signál A	RS-485 signál A
	9	Nepoužitý	10bitová volba protokolu (vstup)
	Pouzdro konektoru	Stínění	Uzemnění rozvaděče

Symetrizace a zakončení síťového kabelu

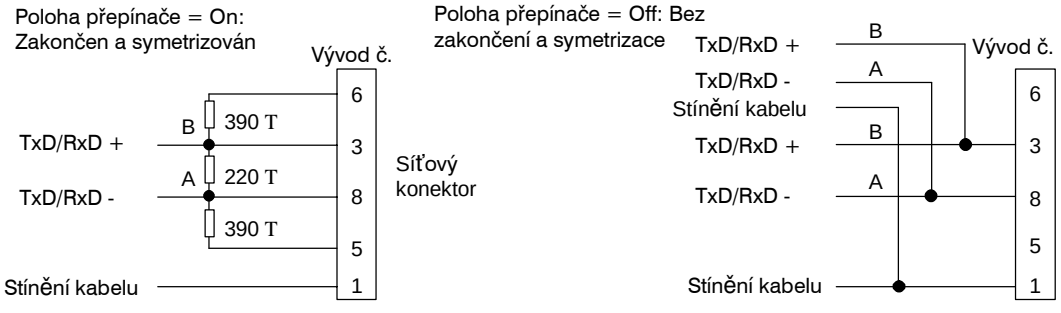
Siemens nabízí dva typy síťových konektorů, které lze použít pro snadné připojení více zařízení k síti: standardní síťový konektor (rozmístění vývodů najdete v tabulce 7-7) a konektor, který má programovací port umožňující připojení programovací stanice nebo zařízení HMI do sítě, aniž by se musela rozebírat stávající kabeláž sítě. Konektor s programovacím portem předává všechny signály (včetně výkonových vývodů) z S7-200 přes programovací port, což je obzvlášť užitečné pro připojení zařízení, která jsou napájena z S7-200 (jako například TD 200).

Oba konektory mají dvě sady šroubovacích svorkovnic, které umožňují připojení vstupních a výstupních síťových kabelů. Oba konektory také mají přepínače pro nastavení symetrizace a zakončení sítě. Obrázek 7-22 ukazuje typickou symetrizaci a zakončení kabelových konektorů.

Kabel musí být ukončen a symetrizován na obou koncích.



Stínění: přibližně 12 mm (1/2 palce) se musí dotýkat kovových vodicích lišt na všech místech.



Obr. 7-22 Symetrizace a zakončení síťového kabelu

Výběr PPI Multi-Master kabelu nebo CP karty pro vaši síť

Jak ukazuje tabulka 7-8, STEP 7–Micro/WIN podporuje RS-232/PPI Multi-Master kabel a USB/PPI Multi-Master kabel i několik CP karet, které umožňují, aby se počítač (nebo programovací zařízení SIMATIC) choval v síti jako master.

Pro přenosové rychlosti do 187,5 kbaud poskytují PPI Multi-Master kabely to nejjednodušší a nejrentabilnější spojení mezi STEP 7–Micro/WIN a jednou CPU S7-200 nebo sítí S7-200. Nabízíme dva typy kabelů PPI Multi-Master; oba je možné použít pro lokální spojení mezi STEP 7–Micro/WIN a sítí S7-200.

USB/PPI Multi-Master kabel patří mezi díly ”plug and play”, které je možné použít s PC podporujícími USB verze 1.1. Zajišťuje oddělení PC a sítě S7-200 a přitom podporuje komunikaci PPI s přenosovými rychlostmi do 187,5 kbaud. Není třeba nastavovat žádné přepínače; stačí zapojit kabel, vybrat kabel PC/PPI jako rozhraní, vybrat protokol PPI a nastavit port na USB na záložce ”Připojení PC”. Pro užívání STEP 7–Micro/WIN je možné k PC připojit naráz pouze jeden USB/PPI Multi-Master kabel.

RS-232/PPI Multi-Master kabel má osm DIP přepínačů: dva z těchto přepínačů se používají na konfiguraci kabelu pro práci se STEP 7–Micro/WIN.

- ❏ Pokud kabel připojujete k PC, vyberte režim PPI (přepínač 5 = 1) a lokální provoz (přepínač 6 = 0).
- ❏ Pokud kabel připojujete k modemu, vyberte režim PPI (přepínač 5 = 1) a dálkový provoz (přepínač 6 = 1).

Kabel zajišťuje oddělení PC a sítě S7-200. Zvolte PC/PPI kabel jako rozhraní a vyberte port RS-232, který chcete použít, na záložce ”Připojení PC”. Na záložce ”PPI” vyberte adresu stanice a přenosovou rychlost sítě. Žádné jiné volby nejsou nutné, protože u RS-232/PPI Multi-Master kabelu je volba protokolu automatická.

Oba kabely, USB/PPI i RS-232/PPI Multi-Master, jsou vybaveny LED, které indikují probíhající komunikaci s PC i probíhající komunikaci v síti.

- ❏ LED Tx indikuje, že kabel přenáší informace do PC.
- ❏ LED Rx indikuje, že kabel přijímá informace z PC.
- ❏ LED PPI indikuje, že kabel přenáší data v síti. Protože Multi-Master kabely jsou držiteli tokenu, je LED PPI rozsvícena trvale, jakmile byla STEP 7–Micro/WIN zahájena komunikace. LED PPI zhasne, když je ukončeno spojení se STEP 7–Micro/WIN. LED PPI bude také blikat rychlostí 1 Hz, když čeká na připojení k síti.

Karty CP obsahují specializovaný hardware, který pomáhá počítači při řízení sítě s více mastery a který podporuje různé protokoly s různými přenosovými rychlostmi.

Každá karta CP má jeden port RS-485 pro připojení k síti. Karta CP 5511 PCMCIA má adaptér pro D konektor s devíti vývody. Jeden konec kabelu se připojí do portu RS-485 komunikační karty, druhý konec k programovacímu portu síťového konektoru.

Jestliže používáte CP kartu s komunikací PPI, nebude STEP 7–Micro/WIN podporovat dvě různé aplikace běžící zároveň na stejné CP kartě. Musíte druhou aplikaci zavřít dříve, než připojíte STEP 7–Micro/WIN k síti prostřednictvím CP karty. Pokud používáte komunikaci MPI nebo PROFIBUS, je povolena komunikace více aplikací STEP 7–Micro/WIN v síti ve stejnou dobu.

Upozornění

Použití neoddděleného převodníku RS-485 na RS-232 může poškodit port RS-232 počítače.

RS-232/PPI a USB/PPI Multi-Master kabely společnosti Siemens (objednací čísla 6ES7 901–3CB30–0XA0 a 6ES7 901–3DB30–0XA0) zajišťují elektrické oddělení portu RS-485 na CPU S7-200 a portu RS-232 nebo USB, který připojuje počítač. Pokud nepoužijte Multi-Master kabel společnosti Siemens, musíte zajistit oddělení portu RS-232 počítače.

Tabulka 7-8 Karty CP a protokoly podporované STEP 7– Micro/WIN

Konfigurace	Přenosová rychlost	Protokol
RS-232/PPI Multi-Master nebo USB/PPI Multi-Master kabel[1] Připojen do portu na počítači	9,6 kbaud až 187,5 kbaud	PPI
CP 5511 Typ II, karta PCMCIA (pro notebook)	9,6 kbaud až 12 Mbaud	PPI, MPI a PROFIBUS
CP 5512 Typ II, karta PCMCIA (pro notebook)	9,6 kbaud až 12 Mbaud	PPI, MPI a PROFIBUS
CP 5611 (verze 3 nebo vyšší) Karta PCI	9,6 kbaud až 12 Mbaud	PPI, MPI a PROFIBUS
CP 1613, S7-1613 Karta PCI	10 Mbaud nebo 100 Mbaud	TCP/IP
CP 1612, SoftNet-S7 Karta PCI	10 Mbaud nebo 100 Mbaud	TCP/IP
CP 1512, SoftNet-S7 Karta PCMCIA (pro notebook)	10 Mbaud nebo 100 Mbaud	TCP/IP

[1] Multi-Master kabely zajišťují elektrické oddělení portu RS-485 (na CPU S7-200) a portu na vašem počítači. Použití neodděleného převodníku RS-485 na RS-232 může poškodit port RS-232 počítače.

Použití zařízení HMI v síti

CPU S7-200 podporuje mnoho typů zařízení HMI jak od firmy Siemens, tak i od jiných výrobců. Některá zařízení HMI (jako například TD 200 nebo TP070) neumožňují výběr komunikačního protokolu, který používají, ale jiná zařízení (jako například OP7 a TP170) ano.

Pokud zařízení HMI výběr komunikačního protokolu umožňuje, vezměte v úvahu následující pokyny:

- Pro zařízení HMI připojené ke komunikačnímu portu CPU S7-200 v síti, která neobsahuje žádná jiná zařízení, zvolte protokol PPI nebo MPI.
- Pro zařízení HMI připojené k modulu EM 277 PROFIBUS vyberte buď protokol MPI, nebo PROFIBUS.
  - Pokud jsou v síti spolu se zařízením HMI programovatelné automaty S7-300 nebo S7-400, zvolte pro zařízení HMI protokol MPI.
  - Pokud je zařízení HMI součástí sítě PROFIBUS, zvolte pro toto zařízení protokol PROFIBUS a zvolte stejný profil, jaký mají ostatní mastery v dané síti PROFIBUS.
- Pro zařízení HMI připojené na komunikační port CPU S7-200, které je nakonfigurováno jako zařízení typu master, zvolte protokol PPI. Optimální je PPI Advanced. Protokoly MPI a PROFIBUS nepodporují CPU S7-200 jako master.

Vytvoření protokolů definovaných uživatelem v režimu Freeport

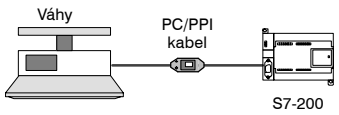
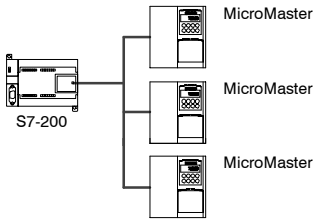
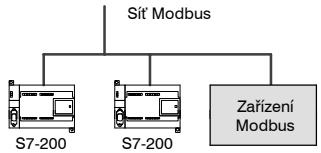
Režim Freeport umožňuje programově řídit komunikační port CPU S7-200. Je možné ho použít pro implementaci komunikačních protokolů, definovaných uživatelem při komunikaci s mnoha typy inteligentních zařízení. Režim Freeport podporuje ASCII protokol i binární protokoly.

Pro aktivaci režimu Freeport se používají byty speciální paměti SMB30 (pro port 0) a SMB130 (pro port 1). Uživatelský program používá pro řízení komunikačního portu následující:

- Instrukce a přerušení (”Vysílání”, XMT) pro vysílání: Instrukce ”Vysílání” umožňuje přenést až 255 znaků. Přerušení informuje program v S7-200 o tom, že byl přenos ukončen.
- Přerušení od příjmu znaku: Přerušení od příjmu znaku: přerušení od příjmu znaku informuje uživatelský program o tom, že byl na portu COM přijat znak. Program pak může tento znak zpracovat podle implementovaného protokolu.
- Instrukce příjmu (”Příjem”, RCV): Instrukce ”Příjem” přijme celou zprávu z portu COM; po jejím přijetí generuje přerušení uživatelského programu. Pro konfiguraci začátku a konce přijímání zpráv na základě definovaných podmínek se používá paměť SM automatu S7-200. Instrukce ”Příjem” umožňuje uživatelskému programu zahájit nebo zastavit zprávu na základě konkrétních znaků nebo časových intervalů. Pomocí instrukce ”Příjem” může být implementována většina protokolů.

Režim Freeport je aktivní pouze tehdy, je–li S7-200 v režimu RUN. Převedení automatu S7-200 do režimu STOP zastaví veškerou komunikaci Freeport a komunikační port se pak vrací do protokolu PPI s nastaveními, která byla konfigurována v systémovém bloku automatu S7-200.

Tabulka 7-9 Použití režimu Freeport

Konfigurace sítě		**Popis**
Použití Freeportu při připojení přes RS-232		Příklad: Použití S7-200 s elektronickými vahami, které mají port RS-232. • RS-232/PPI Multi-Master kabel spojuje port RS-232 na vahách s portem RS-485 CPU S7-200. (Kabel nastavte na režim PPI/Freeport, přepínač 5=0.) • CPU S7-200 používá pro komunikaci s vahami Freeport. • Přenosová rychlost se může pohybovat od 1200 baud do 115,2 kbaud. • Protokol je definován uživatelským programem.
Použití USS protokolu		Příklad: Použití S7-200 s pohony SIMODRIVE MicroMaster. • STEP 7–Micro/WIN s USS knihovnou. • CPU S7-200 funguje jako master, pohony jako slave. Tipy a triky *Příklad programu USS najdete v Programovacích tipech na CD s dokumentací. Viz tip 28.*
Vytvoření uživatelského programu, který emuluje zařízení typu slave v jiné síti		Příklad: Připojení CPU S7-200 k síti Modbus. • Uživatelský program v S7-200 emuluje zařízení Modbus slave. • STEP 7–Micro/WIN má knihovnu Modbus. Tipy a triky *Příklad programu Modbus najdete v Programovacích tipech na CD s dokumentací. Viz tip 41.*

Použití RS-232/PPI Multi-Master kabelu a režimu Freeport s rozhraním RS-232

RS-232/PPI Multi-Master kabel a komunikační funkce Freeport můžete použít pro připojení CPU S7-200 k zařízením, která splňují standard RS-232. Pro práci v režimu Freeport musí být kabel nastaven na PPI/Freeport (přepínač 5 = 0). Přepínač 6 volí buď lokální režim (DCE) (přepínač 6 = 0), nebo dálkový režim (DTE) (přepínač 6 = 1).

RS-232/PPI Multi-Master kabel je ve vysílacím režimu, pokud dochází k přenosu dat z portu RS-232 na port RS-485. Kabel je v přijímacím režimu , pokud je nečinný nebo pokud přenáší data z portu RS-485 na port RS-232. Kabel okamžitě přejde z přijímacího do vysílacího režimu, když na lince RS-232 detekuje znaky.

RS-232/PPI Multi-Master kabel podporuje přenosové rychlosti od 1200 baud do 115,2 kbaud. Pro nastavení RS-232/PPI Multi-Master kabelu na správnou přenosovou rychlost použijte DIP přepínače na kabelu. Tabulka 7-10 ukazuje přenosové rychlosti a polohy přepínače.

Kabel se přepíná zpět do přijímacího režimu, je–li vysílací linka RS-232 v nečinném stavu po dobu definovanou jako doba přepnutí. Volba přenosové rychlosti kabelu určuje dobu přepnutí, jak je patrné z tabulky 7-10.

Pokud jste užili RS-232/PPI Multi-Master kabel v systému, kde se používá komunikace Freeport, musí program v S7-200 počítat s dobou přepnutí v následujících situacích:

Tabulka 7-10 Doba obrátky a nastavení

Přenosová rychlost	Doba přepnutí	Nastavení (1 = nahoru)
115200	0,15 ms	110
57600	0,3 ms	111
38400	0,5 ms	000
19200	1,0 ms	001
9600	2,0 ms	010
4800	4,0 ms	011
2400	7,0 ms	100
1200	14,0 ms	101

- Automat S7-200 odpovídá na zprávy vysílané ze zařízení s rozhraním RS-232.

  Po přijetí požadavku z RS-232 musí S7-200 pozastavit vyslání odpovědi po dobu delší nebo rovnou době přepnutí kabelu.

- Zařízení s rozhraním RS-232 odpovídá na zprávy vysílané z automatu S7-200.

  Po přijetí odpovědi z RS-232 musí S7-200 pozastavit vyslání dalšího požadavku po dobu delší nebo rovnou době přepnutí kabelu.

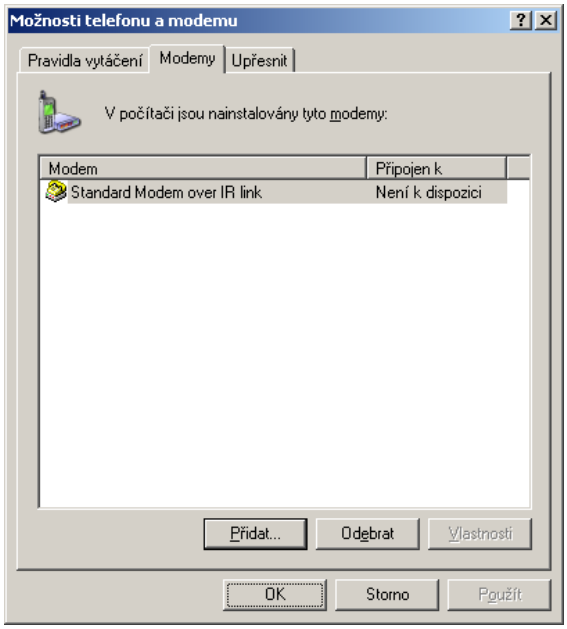
V obou situacích poskytne zpoždění RS-232/PPI Multi-Master kabelu dost času na přepnutí z vysílacího do přijímacího režimu, takže je možné přenášet data z portu RS-485 na port RS-232.

Použití modemu a STEP 7–Micro/WIN

STEP 7–Micro/WIN, verze 3.2 nebo vyšší, používá pro konfiguraci telefonních modemů standardní telefonní připojení a modemy Windows. Konfigurace telefonního připojení a modemu se provádí pomocí ovládacího panelu Windows. Nastavení modemů pod Windows umožňuje:

- Použít většinu interních a externích modemů, podporovaných Windows.
- Použít standardní konfigurace pro většinu modemů, podporovaných Windows.
- Použít standardní pravidla vytáčení Windows pro výběr měst, čísla zemí a směrovacího čísla, pulzní nebo tónové volby a volací karty.
- Použít vyšší přenosové rychlosti při komunikaci s modemovým modulem EM 241.

Pro zobrazení dialogového okna ”Možnosti telefonu a modemu” použijte ovládací panely Windows. Toto dialogové okno umožňuje konfigurovat modem; ten vyberete ze seznamu modemů podporovaných Windows. Pokud není váš typ uveden v seznamu modemů v dialogovém okně, vyberte typ, který vašemu nejblíže odpovídá, nebo kontaktujte výrobce modemu a získejte od něj ovladače modemu pro Windows.



Obr. 7-23 Konfigurace modemu

STEP 7–Micro/WIN umožňuje také použití rádiových a GSM modemů. Tyto typy modemů se neobjevují v dialogovém okně ”Vlastnosti” ve Windows, ale jsou dostupné při konfiguraci připojení ve STEP 7–Micro/WIN.

Konfigurace připojení modemu

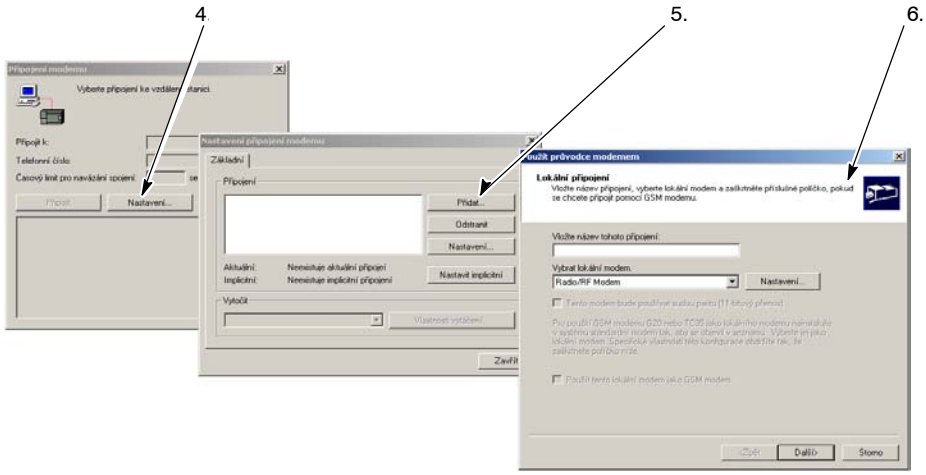
Připojení sdružuje identifikační jméno a fyzické vlastnosti připojení. U telefonního modemu k těmto vlastnostem patří typ modemu, výběr 10 nebo 11bitového protokolu a časové prodlevy.
U celulárních modemů připojení umožňuje nastavení PIN a dalších parametrů. K vlastnostem rádiového modemu patří výběr přenosové rychlosti, parity, řízení toku a další parametry.


"Průvodce"

Vytvoření připojení

Pro vytvoření nového připojení, jeho odstranění nebo editaci připojení použijte "Průvodce", jak ukazuje obrázek 7-24.

1. Dvakrát klikněte na ikonu v okně "Nastavení komunikace".
2. Abyste otevřeli nastavení rozhraní PG/PC, dvakrát klikněte na PC/PPI kabel. Vyberte kabel PPI a klikněte na tlačítko "Vlastnosti". V záložce "Místní připojení" zaškrtněte políčko "Připojení modemu".
3. V komunikačním dialogovém okně klikněte dvakrát na ikonu připojení modemu.
4. Abyste otevřeli dialogové okno "Nastavení připojení modemu", klikněte na tlačítko "Nastavení".
5. Pro spuštění "Použít průvodce modemem" klikněte na tlačítko "Přidat".
6. Nakonfigurujte připojení podle toho, jak průvodce napovídá.

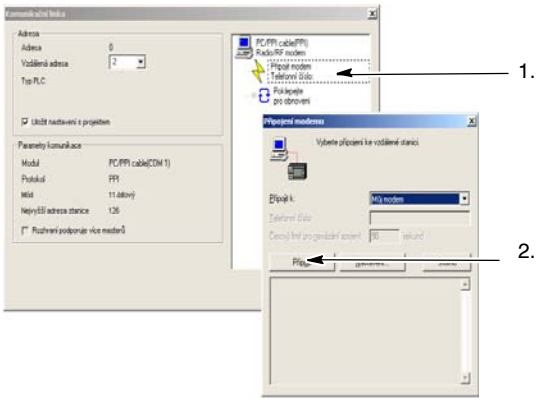


Obr. 7-24 Vytvoření připojení modemu

Připojení modemu k S7-200

Po vytvoření připojení se můžete připojit k CPU S7-200.

1. Abyste otevřete dialogové okno "Komunikační linka", klikněte dvakrát na ikonu "Připojit".
2. V dialogovém okně "Připojení modemu" klikněte na "Připojit", čímž modem vytvoříte.



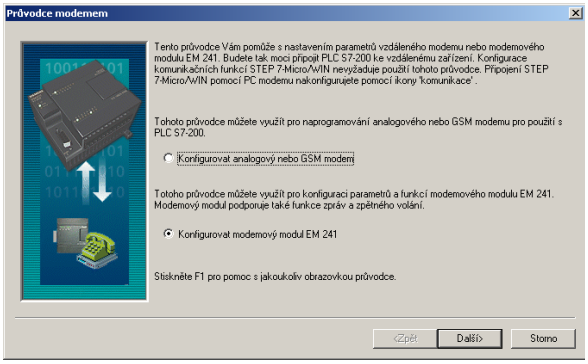
Obr. 7-25 Připojení k S7-200

Konfigurace vzdáleného modemu


"Průvodce modemem"

Vzdálený modem je takový modem, který je připojený k S7-200. Pokud je vzdáleným modemem modul EM 241, není nutná žádná konfigurace. Pokud připojujete externí modem nebo GSM modem, musíte připojení nakonfigurovat.

"Průvodce modemem" konfiguruje vzdálený modem, který je připojený k CPU S7-200. Pro správnou komunikaci s polovičním duplexem RS-485 CPU S7-200 je potřeba speciální konfigurace modemu. Jednoduše vyberte typ modemu a vložte informace podle průvodce. Více informací najdete v online nápovědě.



Obr. 7-26 "Průvodce"

Konfigurace PPI Multi-Master kabelu pro práci se vzdáleným modemem

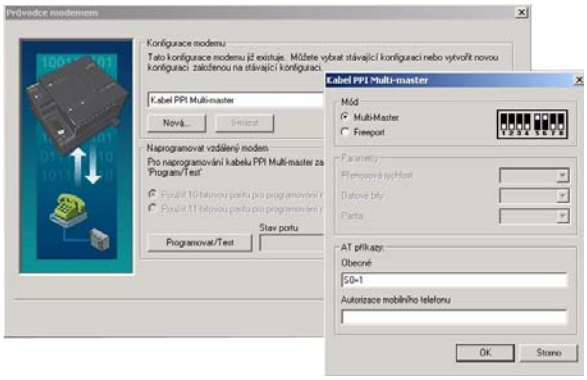
RS-232 PPI Multi-Master kabel dává možnost posílat povelové řetězce AT modemu po zapnutí kabelu. Vezměte prosím na vědomí, že tato konfigurace je nutná pouze tehdy, když musí být změněna implicitní nastavení modemu. Viz obrázek 7-27.

Příkazy modemu je možné specifikovat v "Obecných příkazech". Jediným implicitním nastavením bude příkaz automatického odpovídání.

Příkazy k oprávnění pro mobilní telefon a čísla PIN je možné specifikovat v poli "Oprávnění pro mobilní telefon", například +CPIN=1234.

Každý příkazový řetězec bude modemu odeslán odděleně. Každý řetězec bude předcházet příkaz k přípravě AT modemu.

Tyto příkazy budou v kabelu inicializovány vybráním tlačítka "Program/Test".



Obr. 7-27 "Průvodce modemem" – posílání příkazů modemu

Vezměte prosím na vědomí, že bitová mapa bude uvádět doporučená nastavení přepínačů v závislosti na vybraných parametrech.

Při konfigurování RS-232/PPI Multi-Master kabelu se STEP 7–Micro/WIN musíte zapojit konektor RS-485 do CPU S7-200. Je to zdroj napájení 24 V potřebného pro práci kabelu. Zkontrolujte, že je napájena CPU S7-200.

Když opustíte konfiguraci RS-232/PPI Multi-Master kabelu ve STEP 7–Micro/WIN, odpojte kabel od PC a připojte ho k modemu. Vypněte a obnovte napájení modemu i kabelu. Teď jste připraveni používat kabel pro dálkový provoz v síti PPI s více zařízeními typu master.



Tip
Modem musí mít implicitní nastavení pro použití s PPI Multi-Master kabelem z výrobního závodu.

Konfigurace PPI Multi-Master kabelu pro práci v režimu Freeport

RS-232 PPI Multi-Master kabel nabízí stejnou možnost posílat příkazové řetězce AT modemu, když je kabel nakonfigurován pro režim Freeport. Vezměte prosím na vědomí, že tato konfigurace je nutná pouze tehdy, když musí být změněna implicitní nastavení modemu.

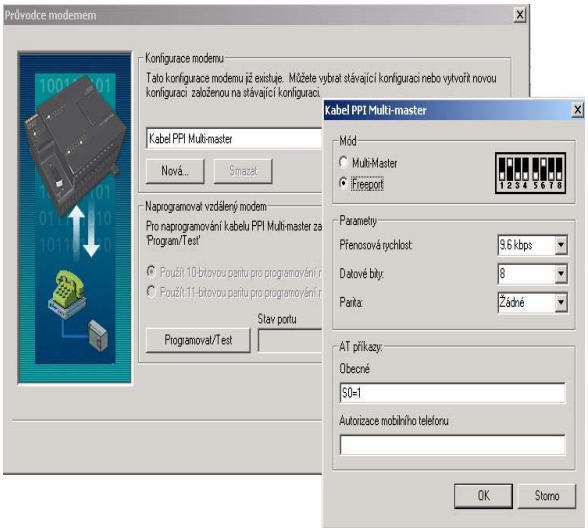
Kabel ale musí být také nakonfigurován tak, aby odpovídal přenosové rychlosti, paritě a počtu datových bitů portu S7-200. To je nutné, protože program aplikace v S7-200 bude kontrolovat konfiguraci těchto parametrů.

Přenosové rychlosti můžete zvolit z rozsahu od 1,2 kbaud do 115,2 kbaud.

Je možné vybrat sedm nebo osm datových bitů.

Lze vybrat sudou, lichou nebo žádnou paritu.

Vezměte prosím na vědomí, že bitová mapa bude uvádět doporučená nastavení přepínačů v závislosti na vybraných parametrech.



Obr. 7-28 "Průvodce modemem" – Odeslání příkazu modemu v režimu Freeport

Při konfigurování RS-232/PPI Multi-Master kabelu se STEP 7–Micro/WIN musíte zapojit konektor RS-485 do CPU S7-200. Je to zdroj napájení 24 V potřebného pro práci kabelu. Zkontrolujte, že je napájena CPU S7-200.

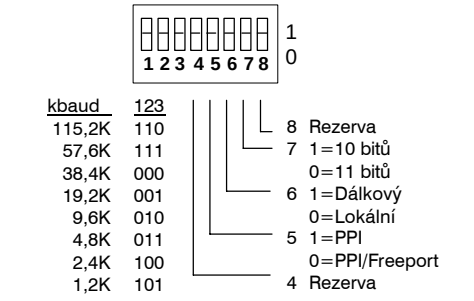
Když opustíte konfiguraci RS-232/PPI Multi-Master kabelu ve STEP 7–Micro/WIN, odpojte kabel od PC a připojte ho k modemu. Vypněte a obnovte napájení modemu i kabelu. Teď jste připraveni používat kabel pro dálkový provoz v síti PPI s více zařízeními typu master.

Tip
Modem musí mít implicitní nastavení pro použití s PPI Multi-Master kabelem z výrobního závodu.

Použití telefonního modemu s RS-232/PPI Multi-Master kabelem

Pro připojení komunikačního portu RS-232 modemu k CPU S7-200 můžete použít RS-232/PPI Multi-Master kabel. Viz obrázek 7-29.

- ❑ Přepínače 1, 2 a 3 nastavují přenosovou rychlost.
- ❑ Přepínačem 5 se volí režim PPI nebo PPI/Freeport.
- ❑ Přepínačem 6 se volí buď režim lokální (ekvivalentní "Komunikačnímu zařízení" – DCE), nebo dálkový (ekvivalentní "Koncovému datovému zařízení" – DTE).
- ❑ Přepínač 7 volí buď 10bitový nebo 11bitový PPI protokol.



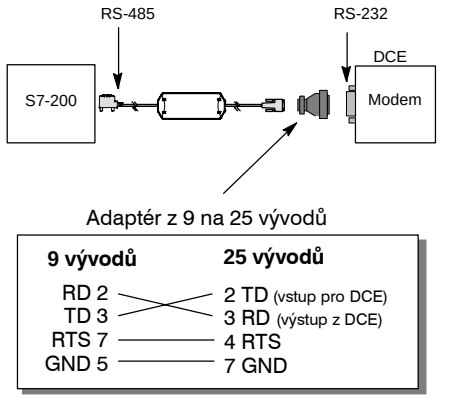
Obr. 7-29 Nastavení pro RS-232/PPI Multi-Masterkabel

Přepínačem 5 se volí provoz v režimu PPI nebo v režimu PPI/Freeport. Pokud pro komunikaci s S7-200 prostřednictvím modemů používáte STEP 7–Micro/WIN, vyberte režim PPI (přepínač 5 = 1). Jinak vyberte režim PPI/Freeport (přepínač 5 = 0).

Přepínač 7 RS-232/PPI Multi-Master kabelu volí buď 10bitový nebo 11bitový mód v režimu PPI/Freeport. Přepínač 7 přepněte pouze tehdy, je–li automat S7-200 připojen ke STEP 7–Micro/WIN s modemem v režimu PPI/Freeport. Jinak nastavte přepínač 7 na 11bitový režim, abyste zajistili správnou součinnost s jinými přístroji.

Přepínač 6 RS-232/PPI Multi-Master kabelu umožňuje nastavit port RS-232 kabelu buď na lokální (DCE), nebo na dálkový (DTE) režim.

- Pokud RS-232/PPI Multi-Master kabel používáte se STEP 7–Micro/WIN, nebo když je RS-232/PPI Multi-Master kabel připojen k počítači, nastavte RS-232/PPI Multi-Master kabel na lokální (DCE) režim.
- Pokud RS-232/PPI Multi-Master kabel užíváte s modemem (což je zařízení DCE), nastavte RS-232/PPI Multi-Master kabel na dálkový (DTE) režim.



Obr. 7-30 Obr. 7–25 Rozmístění vývodů pro adaptéry

Díky tomu není nutné instalovat mezi RS-232/PPI Multi-Master kabel a modem "nulový modem" . Podle typu konektoru modemu může ještě potřebovat adaptér z 9 vývodů na 25.

Obrázek 7-30 znázorňuje rozmístění vývodů běžného modemu.

Více informací o RS-232/PPI Multi-Master kabelu najdete v příloze A. Čísla a funkce vývodů portů RS-485 a RS-232 RS-232/PPI Multi-Master kabelu v lokálním režimu (DCE) jsou uvedeny v tabulce A-66. Tabulka A-67 ukazuje čísla a funkce vývodů portů RS-485 a RS-232 RS-232/PPI Multi-Master kabelu v dálkovém režimu (DTE). RS-232/PPI Multi-Master kabel napájí RTS pouze v dálkovém režimu (DTE).

Použití rádiového modemu s RS-232/PPI Multi-Master kabelem

Pro připojení komunikačního portu RS-232 rádiového modemu k CPU S7-200 můžete použít RS-232/PPI Multi-Master kabel. Provoz s rádiovými modemy ale není stejný jako s telefonními modemy.

Režim PPI

Když je RS-232/PPI Multi-Master kabel nastaven na režim PPI (přepínač 5 = 1), vybrali byste normálně pro provoz s modemem dálkový režim (přepínač 6 = 1). Jestliže však vyberete dálkový režim, způsobí to, že kabel odešle řetězec znaků 'AT' a čeká, až modem odpoví 'OK' při každém zapnutí. Telefonní modemy používají tuto sekvenci pro stanovení přenosové rychlosti, ale rádiové modemy běžně příkazy AT nepřijímají.

Proto musíte pro provoz s rádiovými modemy vybrat lokální režim (přepínač 6 = 0) a použít "nulový modem" mezi konektorem RS-232 kabelu a portem RS-232 na rádiovém modemu. "Nulové modemy" nabízíme s konfigurací buď 9 vývodů na 9 vývodů nebo 9 vývodů na 25 vývodů.

Rádiový modem nakonfigurujte na práci s přenosovou rychlostí 9,6, 19,2, 38,4, 57,6 nebo 115,2 kbaud. RS-232/PPI Multi-Master kabel se automaticky nastaví na kteroukoliv z těchto přenosových rychlostí při přenosu prvního znaku rádiovým modemem.

Režim PPI/Freeport

Když je RS-232/PPI Multi-Master kabel nastaven na režim PPI/Freeport (přepínač 5 = 0), vyberte pro provoz s rádiovým modemem dálkový režim (přepínač 6 = 1). Kabel konfigurujte tak, že nebude posílat žádné příkazy AT pro nastavení modemu.

Přepínače 1, 2 a 3 na RS-232/PPI Multi-Master kabelu nastavují přenosovou rychlost. Viz obrázek 7–29. Vyberte nastavení přenosové rychlosti, která odpovídá přenosové rychlosti programovatelného automatu a rádiového modemu.

Témata pro pokročilé

Optimalizace výkonu sítě

Výkon sítě ovlivňují následující faktory (s tím, že největší vliv má přenosová rychlost a počet masterů):

- Přenosová rychlost: Provoz sítě při nejvyšší přenosové rychlosti podporované všemi částmi zařízení má na výkon sítě největší vliv.
- Počet masterů v síti: Minimalizace počtu masterů také zvyšuje výkon sítě. Každý master zvyšuje požadavky režie sítě; jestliže síť obsahuje méně masterů, snižuje to režii.
- Volba adres zařízení typu master a slave: Adresy masterů by měly být nastaveny tak, aby následovaly za sebou bez mezer. Jakmile je mezi adresami mezera, kontroluje master nepřetržitě adresy v mezeře, zda se tam nenachází další master žádající o připojení do sítě. Tato kontrola vyžaduje čas a zvyšuje režii sítě. Pokud mezi adresami masterů mezera není, kontrola se neprovádí a režie je tak minimalizována. Adresy zařízení typu slave můžete nastavit libovolně a výkon sítě tím neovlivníte, pokud ovšem nejsou tato zařízení mezi zařízeními typu master. Slave umístěný mezi mastery zvyšuje režii sítě stejným způsobem jako mezery mezi adresami jednotlivých masterů.
- Faktor obnovení mezery (Gap update factor, GUF): GUF se používá pouze tehdy, pracuje–li CPU S7-200 jako PPI master a říká S7-200, jak často má hledat jiné mastery v mezeře mezi adresami. Pro nastavení GUF při konfiguraci CPU a jejího rozhraní se používá STEP 7–Micro/WIN. Automat S7-200 se tak konfiguruje na periodickou kontrolu mezery mezi adresami. Při GUF = 1 kontroluje S7-200 mezeru mezi adresami pokaždé, když dostane token; při GUF = 2 kontroluje S7-200 tuto mezeru jednou na každé dvě obdržení tokenu. Pokud jsou mezi adresami masterů mezery, snižuje vyšší GUF režii sítě. Pokud mezi nimi mezery nejsou, nemá GUF na výkon žádný vliv. Nastavení velkého GUF způsobuje velká zpoždění v připojování masterů do sítě, protože jsou adresy méně často kontrolovány. Implicitní nastavení GUF je 10.
- Adresa nejvyšší stanice (HSA): HSA se používá pouze tehdy, pracuje–li CPU S7-200 jako PPI master a definuje nejvyšší adresu, na které master hledá další mastery. Pro nastavení HSA se při konfiguraci portu CPU používá STEP 7–Micro/WIN. Nastavením HSA můžete eliminovat mezeru mezi adresami, kterou musí kontrolovat poslední master (nejvyšší adresa) v síti. Omezení velikosti mezery mezi adresami minimalizuje dobu potřebnou pro nalezení jiného mastera a jeho připojení do sítě. Adresa nejvyšší stanice nemá žádný vliv na adresy zařízení typu slave: master může komunikovat se zařízeními typu slave, která mají adresu vyšší než HSA. Obecně platí, že se má adresa nejvyšší stanice nastavit u všech zařízení typu master na stejnou hodnotu. Tato adresa má být vyšší nebo se rovnat nejvyšší adrese zařízení typu master. Implicitně nastavená hodnota HSA je 31.

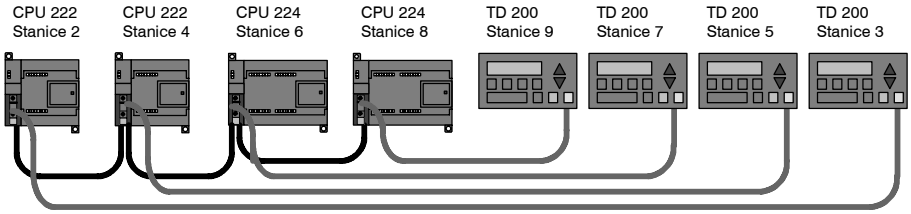
Výpočet doby oběhu tokenu

V síti, kde se předává token, může iniciovat komunikaci pouze ta stanice, která obdržela token. Doba oběhu tokenu (doba, za kterou token oběhne všechny mastery v logickém okruhu) je úměrná výkonu dané sítě.

Na obrázku 7-31je vzorová síť jako příklad pro výpočet doby oběhu tokenu v síti s více mastery. V tomto příkladu komunikuje TD 200 (stanice 3) s CPU 222 (stanice 2), TD 200 (stanice 5) komunikuje s CPU 222 (stanice 4) a tak dále. Dva moduly CPU 224 používají instrukce ”Čtení ze sítě” a ”Zápis do sítě” pro shromažďování dat z ostatních automatů S7-200: CPU 224 (stanice 6) posílá zprávy stanicím 2, 4 a 8 a CPU 224 (stanice 8) posílá zprávy stanicím 2, 4 a 6. V této síti je šest stanic master (čtyři displeje TD 200 a dva moduly CPU 224) a dvě stanice slave (dva moduly CPU 222).



Podrobnosti o oběhu tokenu najdete v Tipech a tricích na CD s dokumentací. Viz tip 42.



Obr. 7-31 Příklad sítě s oběhem tokenu

Aby master mohl vyslat zprávu, musí mít token. Když má například token stanici 3, generuje požadavek pro stanici 2, a pak předá token stanici 5. Stanice 5 pak generuje požadavek pro stanici 4 a následně předá token stanici 6. Stanice 6 pak generuje zprávu pro stanici 2, 4 nebo 8 a předá token stanici 7. Proces generování zpráv a předávání tokenu pokračuje kolem celého logického kruhu od stanice 3 do stanice 5, stanice 6, stanice 7, stanice 8, stanice 9 a nakonec zpátky do stanice 3. Token musí oběhnout celý logický okruh, aby mohl master vyslat žádost o informace. Pro logický okruh šesti stanic, z nichž každá vysílá na jedno obdržení tokenu jeden požadavek na zápis nebo čtení hodnoty jednoho double word (čtyři byty dat), je doba oběhu tokenu přibližně 900 ms při 9600 baud. Zvýšení počtu bytů dat na požadavek nebo zvýšení počtu stanic prodlužuje dobu oběhu tokenu.

Doba oběhu tokenu je určena tím, jak dlouho ho každá stanice drží. Dobu oběhu tokenu v síti s více mastery můžete stanovit sečtením časů, po které každý master token drží. Pokud byl aktivován režim PPI master (pod protokolem PPI), můžete posílat zprávy ostatním automatům S7-200 pomocí instrukcí "Čtení ze sítě" a "Zápis do sítě" automatu S7-200. Pokud posíláte zprávy pomocí těchto instrukcí, můžete pro výpočet přibližné doby oběhu tokenu použít následující vzorec při dodržení těchto předpokladů: každá stanice vysílá jeden požadavek na jedno držení tokenu, požadavek představuje čtení nebo zápis za sebou jdoucích oblastí dat, použití jednoho komunikačního zásobníku v S7-200 nezpůsobuje konflikty a žádný automat S7-200 nemá delší programový cyklus než cca 10 ms.

Doba držení tokenu (T_{hold}) = (128 režie + *n* datových znaků) x 11 bitů/znak x 1/přenosová rychlost v baudech
Doba oběhu tokenu (T_{rot}) = T_{hold} master 1 + T_{hold} master 2 + . . . + T_{hold} master *m*
kde *n* je počet datových znaků (bytů) *m* je počet zařízení typu master

Následující rovnice počítají doby oběhu (jeden "bit time" se rovná trvání jedné signální periody) pro příklad ukázaný na obrázku 7-31:

T (doba držení tokenu) = (128 + 4 znaky) x 11 bitů/znak x 1/9600 bit time/s
= 151,25 ms/master
T (doba oběhu tokenu) = 151,25 ms/master ≦ 6 master
= 907,5 ms



Tip
Software SIMATIC NET COM PROFIBUS zahrnuje analyzátor pro vyhodnocení výkonu sítě.

Porovnání dob oběhu tokenu

Tabulka 7-11 přináší srovnání doby oběhu tokenu při různých počtech stanic, velikosti přenášených dat a přenosových rychlostech. Časy jsou vypočítané pro případ použití instrukcí ”Čtení ze sítě” a ”Zápis do sítě” CPU S7-200 nebo jiným masterem.

Tabulka 7-11 Doba oběhu tokenu (v sekundách)

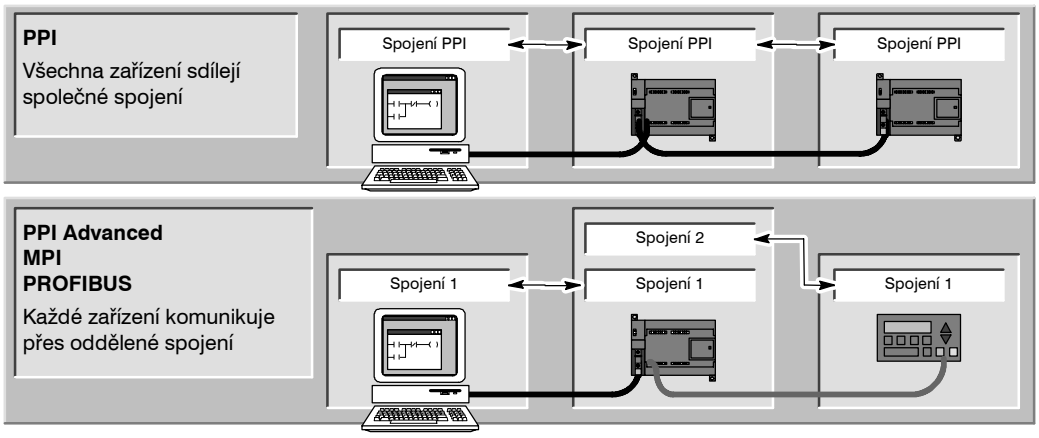
Přenosová rychlost	Počet přenášených bytů	Počet masterů								
		2	3	4	5	6	7	8	9	10
9,6 kbaud	1	0.30	0.44	0.59	0.74	0.89	1.03	1.18	1.33	1.48
	16	0.33	0.50	0.66	0.83	0.99	1.16	1.32	1.49	1.65
19,2 kbaud	1	0.15	0.22	0.30	0.37	0.44	0.52	0.59	0.67	0.74
	16	0.17	0.25	0.33	0.41	0.50	0.58	0.66	0.74	0.83
187,5 kbaud	1	0.009	0.013	0.017	0.022	0.026	0.030	0.035	0.039	0.043
	16	0.011	0.016	0.021	0.026	0.031	0.037	0.042	0.047	0.052

Vysvětlení datových spojů mezi zařízeními v síti

Zařízení v síti komunikují přes jednotlivá spojení, která tvoří spoje mezi zařízeními typu master a slave. Jak je vidět na obrázku 7-32, komunikační protokoly se liší v tom, jak se spojeními zacházejí:

- Protokol PPI využívá jedno spojení sdílené všemi zařízeními v síti.
- Protokoly PPI Advanced, MPI a PROFIBUS využívají oddělená spojení mezi libovolnými dvěma zařízeními, která spolu komunikují.

Při použití protokolu PPI Advanced, MPI nebo PROFIBUS nemůže druhý master zasahovat do spojení, které bylo navázáno mezi zařízeními typu master a slave. CPU S7-200 a EM 277 mají vždy vyhrazeno jedno spojení pro STEP 7–Micro/WIN a jedno spojení pro zařízení HMI. Jiná zařízení typu master tato vyhrazená spojení nemohou používat. Tím je zajištěno, že je vždy možné spojit nejméně jednu programovací stanici a nejméně jedno zařízení HMI s CPU S7-200 nebo EM 277, jestliže master používá protokol, který spojení podporuje (jako je například PPI Advanced).



Obr. 7-32 Řízení komunikačních spojení

Jak je vidět z tabulky 7-12, CPU S7-200 a EM 277 mají určitý počet spojení. Každý port (Port 0 a Port 1) CPU S7-200 podporuje až čtyři oddělená spojení. (To umožňuje maximálně osm spojení na CPU S7-200.) Tato spojení jsou navíc ke sdílenému spojení PPI. EM 277 podporuje šest spojení.

Tabulka 7-12 Možnosti CPU S7-200 a modulů EM 277

Modul		Přenosová rychlost	Počet spojení	Volby profilu protokolu STEP 7 – Micro/WIN
CPU S7-200,	Port 0	9,6 kbaud, 19,2 kbaud nebo 187,5 kbaud	4	PPI, PPI Advanced, MPI a PROFIBUS[1]
	Port 1	9,6 kbaud, 19,2 kbaud nebo 187,5 kbaud	4	PPI, PPI Advanced, MPI a PROFIBUS[1]
Modul EM 277		9,6 kbaud až 12 Mbaud	6 na modul	PPI Advanced, MPI a PROFIBUS

[1] Jestliže je pro spojení STEP 7 – Micro/WIN s CPU S7-200 přes port 0 nebo port 1 použita CP karta, můžete profily MPI nebo DP PROFIBUS vybrat pouze tehdy, je – li automat S7-200 nakonfigurován jako slave.

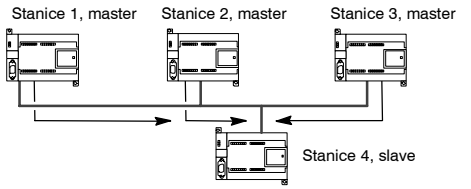
Práce se složitými sítěmi

Pokud jde o S7-200, mají složité sítě více masterů S7-200, které pro komunikaci s jinými zařízeními v síti PPI užívají instrukce "Čtení ze sítě" (NETR) a "Zápis do sítě" (NETW). Tyto sítě mívají speciální problémy, které mohou zablokovat komunikaci mezi zařízením typu master a slave.

Pokud síť pracuje s nižší přenosovou rychlostí (jako například 9,6 kbaud nebo 19,2 kbaud), dokončí každý master transakci (čtení nebo zápis) dříve, než předá token. Při rychlosti 187,5 kbaud však vyšle master požadavek na zařízení typu slave, a pak předá token, čímž na slave ponechává nevyřízený požadavek.

Obrázek 7-33 znázorňuje síť s potenciálními komunikačními konflikty. V této síti jsou stanice 1, 2 a 3 zařízení typu master, která používají instrukce "Čtení ze sítě" nebo "Zápis do sítě" pro komunikaci se stanicí 4. Instrukce "Čtení ze sítě" a "Zápis do sítě" používají protokol PPI, takže všechny automaty S7-200 sdílejí jediné spojení se stanicí 4.

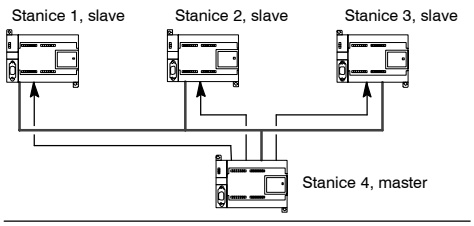
V tomto příkladu stanice 1 generuje požadavek na stanici 4. Při přenosových rychlostech vyšších než 19,2 kbaud pak stanice 1 předá token stanici 2. Pokud se stanice 2 pokusí vyslat požadavek na stanici 4, je požadavek stanice 2 zamítnut, protože je stále přítomen požadavek stanice 1. Všechny požadavky na stanici 4 budou zamítány, dokud stanice 4 nedokončí odpověď pro stanici 1. Až po dokončení této odpovědi, může jiná master stanice vyslat požadavek na stanici 4.



Obr. 7-33 Komunikační konflikt

Abyste zabránili konfliktu na komunikačním portu stanice 4, zvažte možnost konfigurace stanice 4 jako jediného masteru v síti, jak ukazuje obrázek 7-34. Potom stanice 4 generuje požadavky na čtení nebo zápis ostatním automatům S7-200.

Nejen že tato konfigurace zabraňuje komunikačnímu konfliktu, ale také snižuje režii, způsobenou více mastery a umožňuje efektivnější práci sítě.



Obr. 7-34 Zabránění konfliktu

U některých aplikací však není možné snížit počet masterů v síti. Jestliže je v síti několik masterů, musíte řídit dobu oběhu tokenu a zajistit, aby síť nepřekročila cílovou dobu oběhu tokenu. (Doba oběhu tokenu je časový interval, který uplyne od doby, kdy master předalo token, do doby, kdy ho opět obdrží.)

Tabulka 7-13 HSA a celková doba oběhu tokenu

HSA	9,6 kbaud	19,2 kbaud	187,5 kbaud
HSA = 15	0,613 s	0,307 s	31 ms
HSA = 31	1,040 s	0,520 s	53 ms
HSA = 63	1,890 s	0,950 s	97 ms
HSA = 126	3,570 s	1,790 s	183 ms

Pokud je doba potřebná pro návrat tokenu na master větší než celková doba oběhu tokenu, nemůže master generovat požadavek. Master může generovat požadavek pouze tehdy, je – li skutečná doba oběhu tokenu kratší než cílová doba oběhu tokenu.

Adresa nejvyšší stanice (HSA) a nastavená přenosová rychlost S7-200 určují cílovou dobu oběhu tokenu. Tabulka 7-13 uvádí cílové doby oběhu.

U nižších přenosových rychlostí, jako například 9,6 kbaud a 19,2 kbaud, vyčká master na odpověď na svůj požadavek, než předá token. Protože zpracování požadavku/odpovědi může v porovnání s programovým cyklem trvat relativně dlouho, je vysoce pravděpodobné, že každý master bude v síti připraven vyslat požadavek pokaždé, když obdrží token. Pak poroste skutečná doba oběhu tokenu a některé mastery by nemusely být schopny zpracovat požadavky. V některých situacích by master mohl zpracovávat požadavky opravdu jen zřídka.

Například: Představte si síť 10 masterů, které vysílají 1 byte rychlostí 9,6 kbaud, přičemž je tato síť konfigurována s HSA 15. V tomto příkladu má každý master vždy připravenou zprávu k vyslání. Jak je patrné z tabulky 7-13, celková doba oběhu tokenu je pro tuto síť 0,613 s. Ale na základě údajů o výkonu, uvedených v tabulce 7-11, je skutečná doba oběhu tokenu potřebná pro danou síť 1,48 s. Protože je skutečná doba oběhu tokenu delší než cílová doba oběhu tokenu, budou některé mastery moci vyslat zprávu teprve při nějakém pozdějším oběhu tokenu.

Pro zlepšení situace, kdy je skutečná doba oběhu tokenu delší než cílová doba oběhu tokenu, máte dvě základní volby:

- Můžete zkrátit skutečnu dobu oběhu tokenu snížením počtu masterů v síti; záleží na vaší aplikaci. Toto řešení ale nemusí být proveditelné.
- Můžete prodloužit celkovou dobu oběhu tokenu zvýšením HSA pro všechny mastery v síti.

Zvýšení HSA může v síti způsobit jiný problém tím, že ovlivní délku času, který automat S7-200 potřebuje, aby se přepnul do režimu master a vstoupil do sítě. Pokud používáte časovač, abyste si zajistili, že bude instrukce ”Čtení ze sítě” nebo ”Zápis do sítě” dokončena v požadované době, může zpoždění aktivace režimu master a přidání automatu S7-200 do sítě jako master způsobit, že čas pro instrukci vyprší. Zpoždění můžete při připojování masteru minimalizovat snížením faktoru obnovení mezery (GUF) pro všechny mastery v síti.

Vzhledem ke způsobu, jakým jsou požadavky do slave zapisovány a jakým jsou v něm při rychlosti 187,5 kbaud uchovávány, měli byste při volbě celkové doby oběhu tokenu počítat s nějakým časem navíc. Při rychlosti 187,5 kbaud by se skutečná doba oběhu tokenu měla rovnat přibližně polovině celkové doby oběhu tokenu.

Pro zjištění doby oběhu tokenu použijte údaje o výkonu z tabulky 7-11.Tím určíte dobu potřebnou k dokončení operací ”Čtení ze sítě” a ”Zápis do sítě”. Pro výpočet doby potřebné pro zařízení HMI (jako například TD 200) použijte údaje o výkonu pro přenos 16 bytů. Dobu oběhu tokenu vypočítejte součtem časů pro každé zařízení v síti. Nejhorší případ nastane v případě součtu časů tehdy, kdy chtějí všechna zařízení zpracovat požadavek během jednoho oběhu tokenu. Tím je definována maximální doba oběhu tokenu potřebná pro danou síť.

Například: Představte si síť s přenosovou rychlostí 9,6 kbaud a se čtyřmi displeji TD 200 a čtyřmi automaty S7-200, kde každý S7-200 zapisuje sekundu 10 bytů dat do jiného S7-200. Použijte tabulku 7-11 a vypočítejte příslušné doby přenosu pro síť:

4 displeje TD 200 přenášejí 16 bytů dat =	0,66 s
4 automaty S7-200 přenášejí 10 bytů dat =	0,63 s
Celková doba oběhu tokenu =	1,29 s

Abyste síti poskytli dostatek času na zpracování všech požadavků během jednoho oběhu tokenu, nastavte HSA na 63. (Viz tabulku 7-13.) Volba cílové doby oběhu tokenu (1,89 s), která je delší než maximální doba oběhu tokenu (1,29 s), zajišťuje, že každé zařízení může přenášet data při každém oběhu tokenu.

Pro zlepšení spolehlivosti sítě s více mastery byste měli zvážit také následující vylepšení:

- Změna rychlosti aktualizace zařízení HMI, která ponechá delší čas mezi jednotlivým aktualizacemi. Změňte například aktualizace displeje TD 200 z "Co možná nejrychleji" na "Jedenkrát za sekundu".
- Snížení počtu požadavků (a režie sítě na zpracování požadavků) zkombinováním operací "Čtení ze sítě"a "Zápis do sítě". Například místo dvou operací použijte"Čtení ze sítě", z nichž každá čte 4 byty, jednu operaci "Čtení ze sítě", která čte 8 bytů. Doba zpracování dvou požadavků po 4 bytech je daleko delší než doba zpracování jednoho požadavku o 8 bytech.
- Změňte rychlost aktualizace automatů S7-200 master, aby se nepokoušely o aktualizaci rychleji, než je doba oběhu tokenu.

Konfigurace RS-232/PPI Multi-Master kabelu pro dálkové ovládání

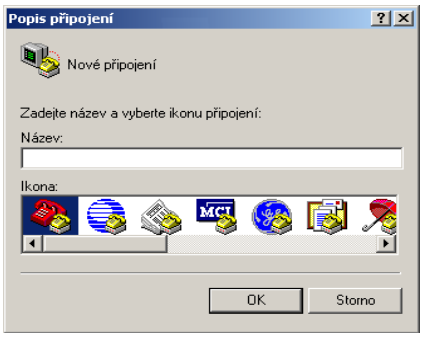
HyperTerminal jako konfigurační nástroj

Pokud nemáte pro konfiguraci RS-232/PPI Multi-Master kabelu pro dálkový provoz k dispozici STEP 7–Micro/WIN, můžete použít HyperTerminal nebo jakýkoliv jiný balík neprogramovatelného terminálu. RS-232/PPI Multi-Master kabel má zabudovaná menu, která vás provedou konfigurací kabelu pro dálkový provoz.

Při konfigurování RS-232/PPI Multi-Master kabelu pomocí HyperTerminal musíte zapojit konektor RS-485 do CPU S7-200. Je to zdroj napájení 24 V, potřebného pro práci kabelu. Zkontrolujte, že je napájena CPU S7-200.

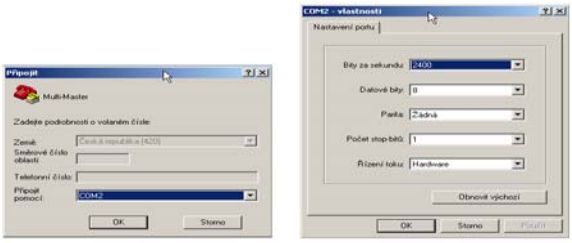
Chcete–li vyvolat HyperTerminal na PC, klikněte na **Start > Programy > Příslušenství > Komunikace > HyperTerminal**.

Spustí se aplikace HyperTerminal a vyzve k zadání popisu spojení. Musíte zadat název spojení (např. Multi-Master). Klikněte na OK. Můžete si vybrat ikonu nebo akceptovat implicitní ikonu, přiřazenou novému spojení. Viz obrázek 7-35.



Obr. 7-35 Popis připojení HyperTerminal

Zobrazí se obrazovka "Připojit k". Vyberte komunikační port, který budete používat, a klikněte na OK. Jako další se objeví obrazovka "Vlastnosti COMx". Přijměte implicitní nastavení a klikněte na OK. Viz obrázek 7-36.

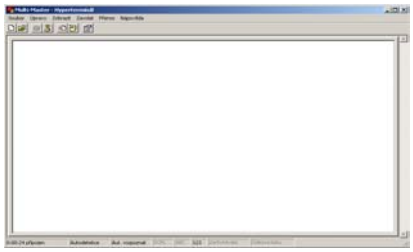


Obr. 7-36 Obrazovky HyperTerminal "Připojit k" a "Vlastnosti COMx"

Po kliknutí na OK se kurzor přemístí do editačního okna obrazovky HyperTerminal, jak je to ukázáno na obrázku 7-37. Všimněte si, že stavový řádek dole v okně HyperTerminal ukazuje, že jste připojeni a že běží časovač, který ukazuje délku připojení.

Z menu vyberte **Volání > Odpojit**. Stavový řádek teď ukazuje, že jste odpojeni.

Vyberte **Zobrazit > Font**. Vyberte Courier New a klikněte na OK.

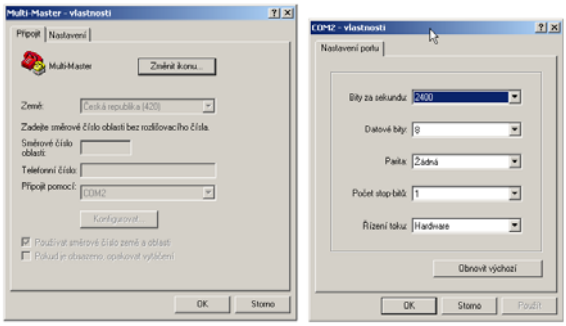


Obr. 7-37 Editační okno pro Multi-Master HyperTerminal

Vyberte **Soubor > Vlastnosti**. Na záložce ”Připojit k” klikněte na tlačítko **Konfigurovat ...**, aby se zobrazily vlastnosti komunikačního portu. Viz obrázek 7-38.

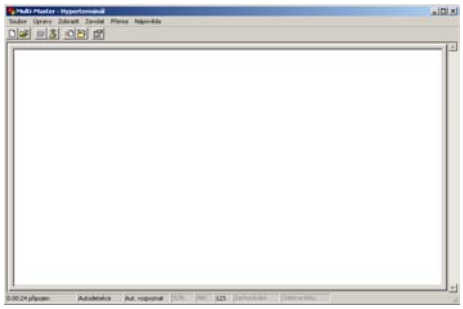
V dialogovém okně ”Vlastnosti COMx” vyberte přenosovou rychlost z roletového menu pro bity za sekundu. Musíte vybrat přenosovou rychlost od 9600 do 115200 bitů za sekundu (normálně 9600). Pomocí příslušných roletových menu vyberte 8 datových bitů, žádnou paritu, jeden stop bit a bez řízení toku.

Klikněte na OK, abyste se vrátili na záložku ”Připojit k”.



Obr. 7-38 Vlastnosti Multi-Master a vlastnosti COMx

Vyberte záložku ”Nastavení”. V roletovém menu ”Emulace” vyberte ANSI a klikněte na OK. Tím se vrátíte do editačního okna obrazovky HyperTerminal. Stavový řádek dole na obrazovce by měl ukazovat: ”Odpojeno ANSI 9600 8–N–1”, jak ukazuje obr. 7-39.



Obr. 7-39 Editace HyperTerminal – Odpojení ANSI

Chcete–li zahájit komunikaci s RS-232/PPI Multi-Master kabelem, napište ”hhh”. Když napíšete ”hhh”, musí LED Rx na kabelu blikat po dobu asi jedné sekundy. LED TX se krátce rozsvítí, když kabel odpoví volbou jazyků.

Vložte číslo, které odpovídá vaší volbě jazyka (pro zrušení implicitního výběru použijte klávesu zpětného kroku), a stiskněte klávesu ENTER. Obrázek 7-40 ukazuje displej volby jazyka a displej voleb ”Nastavení RS232/PPI kabelu pro dálkový provoz”.

Tento displej také ukazuje verzi firmwaru kabelu.



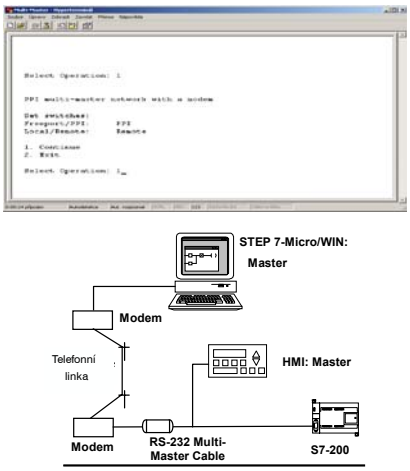
Obr. 7-40 Volba jazyka HyperTerminal a nastavení RS-232/PPI kabelu

Displej ”Nastavení RS232/PPI kabelu pro dálkový provoz” vás provede kroky potřebnými pro konfiguraci kabelu na typ požadovaného dálkového provozu.

- ❏ Pokud máte starší verzi STEP 7–Micro/WIN, vyberte volbu 2 ”Síť PPI s jedním zařízením typu master a s modemem”.
- ❏ Pokud s modemem komunikujete v režimu Freeport, vyberte volbu 3.

Vyberte například volbu 1 pro síť PPI s více zařízeními typu master a s modemem, který užívá STEP 7–Micro/WIN 3.2 Service Pack 4 nebo novější.

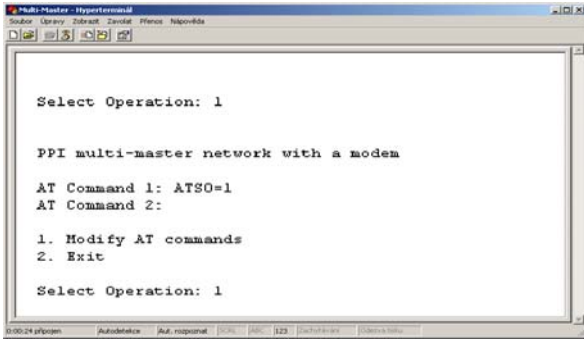
Displej HyperTerminal na obrázku 7-41 ukazuje nastavení přepínačů, které je třeba na kabelu nastavit. Nastavení přepínačů umožňují STEP 7–Micro/WIN účast v dálkové síti prostřednictvím modemů s jedním nebo více mastery a jedním nebo více programovatelnými automaty S7-200. Taková síť je ukázána na obrázku 7-41.



Obr. 7-41 HyperTerminal – Nastavení RS-232/PPI kabelu

Když jste nastavili přepínače tak, jak je ukázáno, vyberte pokračování. Výsledný displej HyperTerminal ukazuje obrázek 7-42.

Vzdálený modem (ten, který je připojen k RS-232/PPI Multi-Master kabelu) musí být nastaven na výchozí nastavení od výrobce. Když je vzdálený modem nastaven na výchozí nastavení od výrobce, vložte AT řetězce potřebné pro naprogramování modemu na provoz s RS-232/PPI Multi-Master kabelem. Běžně je třeba odeslat jediný řetězec, a to ATS0=1, který konfiguruje modem na automatické odpovídání na příchozí hovory po prvním zazvonění.



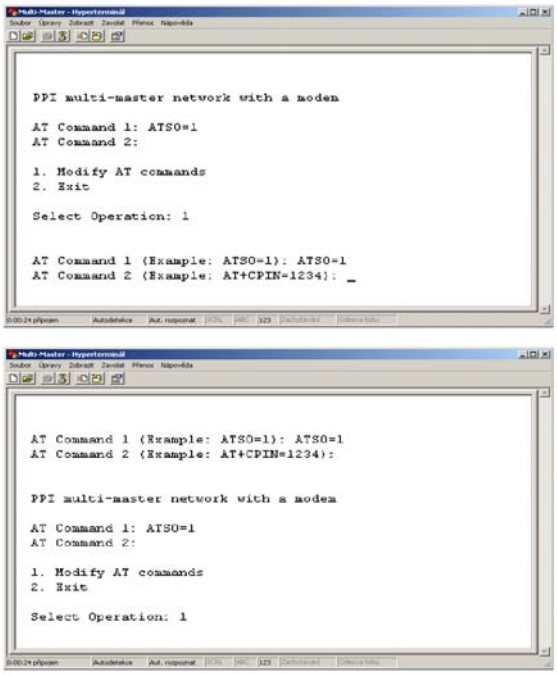
Obr. 7-42 HyperTerminal – Vzdálený modem

Jestliže používáte GSM modem, který vyžaduje PIN, použijte druhý AT příkaz k zadání PIN (AT příkazy podporované vaším modemem najdete v manuálu pro modem). Pokud potřebujete AT příkazy modifikovat, proveďte výběr a vložte na vyzvání požadované příkazy. Výzvy obsahují příklady řetězců AT příkazů, aby vám s jejich formátováním pomohly.

RS-232/PPI Multi-Master kabel odešle tyto AT řetězce modemu při každém zapnutí kabelu. Ujistěte se, že se modem zapne před zapnutím kabelu nebo velmi těsně kolem této doby. Určitě také vypněte a obnovte napájení kabelu, když vypnete a obnovíte napájení modemu. To kabelu umožní správně konfigurovat modem a pracovat s nejvyšší dostupnou přenosovou rychlostí.

Displeje HyperTerminal na obrázku 7-43 ukazují, jak vkládat AT příkazy. Pokud na výzvu nepotřebujete vkládat druhý AT příkaz, stiskněte klávesu ENTER. Tím se vrátíte do výběru pro modifikaci AT příkazů nebo ukončení práce. Jestliže jste s vkládáním AT příkazů skončili, vyberte ”Konec”.

Když opustíte konfiguraci HyperTerminal RS-232/PPI Multi-Master kabelu, odpojte kabel od PC a připojte ho k modemu. Vypněte a obnovte napájení modemu i kabelu. Teď jste připraveni používat kabel pro dálkový provoz v síti PPI s více zařízeními typu master.



Obr. 7-43 HyperTerminal – AT příkazy

Provoz v režimu Freeport s HyperTerminal

Konfigurování RS-232/PPI Multi-Master kabelu pro provoz v režimu Freeport s použitím HyperTerminal je velmi podobné výše popsanému příkladu konfigurace. Pracujte podle výzev a konfigurujte kabel podle svých potřeb.

Návod pro odstraňování poruch hardwaru a ladění softwaru

8

STEP 7–Micro/WIN má softwarové nástroje, které vám pomohou při ladění a odzkoušení vašeho programu. Tyto prvky zahrnují prohlížení stavu programu při jeho vykonávání, zadání běhu S7-200 po určený počet programových cyklů a vnucování hodnot.

Jako návod pro stanovení příčiny a možného řešení při odstraňování hardwarových problémů S7-200 použijte tabulku 8-1.

Přehled kapitoly

Funkce pro adaptaci uživatelského programu 244
Zobrazení stavu programu 246
Použití stavového diagramu pro monitorování a modifikaci dat 247
Force (vnucení) specifických hodnot 248
Zadání běhu programu po zadaný počet programových cyklů 248
Návod pro odstraňování poruch hardwaru 249

Funkce pro adaptaci uživatelského programu

STEP 7 – Micro/WIN má několik funkcí, které vám pomohou při ladění vašeho programu: záložky, tabulku křížových referencí a editaci v režimu RUN.

Použití záložek pro snadný přístup k programu

V programu je možné nastavit záložky, které usnadňují přesuny dopředu a zpátky mezi označenými (záložkou opatřenými) řádky dlouhého programu. Je možné přejít na následující nebo předcházející řádek programu opatřený záložkou.

Použití tabulky křížových odkazů pro kontrolu odkazů v programu


Křížové odkazy

Tabulka křížových odkazů zobrazuje informace o křížových odkazech a použití prvků v programu.

Tabulka křížových odkazů uvádí všechny operandy použité v programu a identifikuje umístění v programovém bloku, networku nebo řádku a kontext instrukce pro operand při každém jeho použití.

Chcete – li změnit zobrazení všech operandů, můžete přepínat mezi symbolickým a absolutním náhledem.

Křížové odkazy

	Element	Blok	Umístění	Kontext
1	Disp_Current	MAIN (OB1)	Network 3, Řádek 2	MOVW
2	Disp_Preset	MAIN (OB1)	Network 2, Řádek 2	MOVW
3	Disp_Preset	MAIN (OB1)	Network 3, Řádek 3	MOVW
4	Saved_Preset	MAIN (OB1)	Network 2, Řádek 2	MOVW
5	Saved_Preset	MAIN (OB1)	Network 3, Řádek 3	MOVW
6	Saved_Preset	MAIN (OB1)	Network 6, Řádek 2	TON
7	Disp_Msg_Two	MAIN (OB1)	Network 1, Řádek 2	=
8	Disp_Msg_One	MAIN (OB1)	Network 4, Řádek 2	=
9	Msg_Ack	MAIN (OB1)	Network 5, Řádek 1	LD
10	Msg_Ack	MAIN (OB1)	Network 5, Řádek 2	R
11	Edit_Bit	MAIN (OB1)	Network 2, Řádek 1	LD

Obr. 8-1 Tabulka křížových odkazů



Tip

Pokud kliknete dvakrát na položku v tabulce křížových odkazů, přeskočíte do dané části programu nebo bloku.

Editace uživatelského programu v režimu RUN

Modely CPU S7-200 verze 2.0 (a vyšší) podporují editaci v režimu RUN. Schopnost editování v režimu RUN vám má umožňovat provádění malých změn programu s minimálním narušením procesu programem řízeného. Implementace této možnosti ale také umožňuje zásadní změny programu, které mohou program narušit nebo dokonce být nebezpečné.



Varování

Jestliže nahrajete změny do S7-200 v režimu RUN, ovlivní tyto změny okamžitě průběh procesu. Změna programu v režimu RUN může mít za následek neočekávané chování systému, které může způsobit smrt nebo vážné poranění obsluhy nebo škodu na zařízení.

Editaci v režimu RUN by měli provádět pouze oprávnění pracovníci, kteří znají účinek úprav v režimu RUN na provoz systému.

Chcete – li provést editaci programu v režimu RUN, musí připojená CPU S7-200 podporovat editaci v režimu RUN a musí být v režimu RUN.

1. Vyberte příkaz menu **Odladit > Editace programu v RUN**.
2. Pokud se projekt liší od programu v automatu S7-200, jste vyzváni k jeho uložení. Editaci v režimu RUN je možné provádět pouze v programu uloženém v S7-200.
3. STEP 7 – Micro/WIN vás upozorní, editujete – li program v režimu RUN, a vyzve k pokračování nebo ke zrušení operace. Pokud kliknete na ”Pokračovat”, STEP 7 – Micro/WIN provede upload programu z S7-200. Teď můžete program editovat v režimu RUN. Pro editaci neplatí žádná omezení.



Tip

Instrukce nástupné (EU) a sestupné hrany (ED) jsou zobrazeny s operandem. Chcete–li si prohlédnout informace o instrukcích na vyhodnocení hran, vyberte v ”Zobrazit” ikonu ”Křížové odkazy”. Pod záložkou ”Využití hran” je seznam čísel pro instrukce na vyhodnocení hran v uživatelském programu. Při editaci programu je třeba dávat pozor, abyste nepřiřadili stejné číslo dvěma hranám.

Download programu v režimu RUN

Editace v režimu RUN umožňuje provést download programového bloku, pouze pokud je S7-200 v režimu RUN. Dříve než provedete download programového bloku v režimu RUN, zvažte účinek úpravy v režimu RUN na práci S7-200 v následujících situacích:

- Pokud jste zrušili řídicí logiku nějakého výstupu, S7-200 uchová poslední stav toho výstupu do příštího vypnutí a obnovení napájení nebo přechodu do režimu STOP.
- Pokud jste zrušili funkce vysokorychlostního čítače nebo impulzního výstupu, které byly v chodu, pokračují tyto funkce v činnosti až do příštího vypnutí a obnovení napájení nebo přechodu do režimu STOP.
- Pokud jste zrušili instrukci ”Připojit přerušení”, ale nezrušili jste podprogram přerušení, bude S7-200 nadále pokračovat v provádění podprogramu přerušování až do příštího vypnutí a obnovení napájení nebo přechodu do režimu STOP. Obdobně, pokud jste zrušili instrukci ”Odpojit přerušení”, nejsou přerušení zastavena až do příštího vypnutí a obnovení napájení nebo přechodu do režimu STOP.
- Pokud jste přidali instrukci ”Připojit přerušení”, která je podmíněna bitem prvního programového cyklu, není tato událost aktivována až do příštího vypnutí a obnovení napájení nebo přechodu režimu STOP do režimu RUN.
- Pokud jste zrušili instrukci ”Povolit přerušení”, budou přerušení pokračovat ve své činnosti až do příštího vypnutí a obnovení napájení nebo přechodu režimu RUN režimu STOP.
- Pokud jste modifikovali adresu příjmové schránky v tabulce a příjmová schránka je aktivní v době, kdy S7-200 přepne ze starého na modifikovaný program, pokračuje S7-200 v zapisování přijímaných dat na starou adresu v tabulce. Instrukce ”Čtení ze sítě” a ”Zápis do sítě” fungují stejným způsobem.
- Jakákoliv logika, která je podmíněna stavem bitu prvního programového cyklu, nebude provedena až do příštího vypnutí a obnovení napájení nebo přechodu z režimu STOP do režimu RUN. Bit prvního programového cyklu je nastaven pouze přechodem do režimu RUN a není ovlivněn editací v režimu RUN.



Tip

Abyste mohli provést download programu v režimu RUN, S7-200 musí podporovat editaci v režimu RUN, program musí být přeložený bez chyb a nesmí docházet k chybám v komunikaci mezi STEP 7–Micro/WIN a S7-200.

Je možné nahrát pouze programový blok.

Chcete–li provést download programu v režimu RUN, klikněte na tlačítko ”Download” nebo vyberte příkaz menu **Soubor > Download**. Pokud se program úspěšně přeloží, nahraje STEP 7–Micro/WIN programový blok do S7-200.

Ukončení editace v režimu RUN

Chcete–li ukončit editaci v režimu RUN, vyberte příkaz menu **Odladit > Editace programu v RUN** a zrušte zaškrtnutí. Pokud jste provedli změny, které nebyly uloženy, STEP 7–Micro/WIN zobrazí výzvu, abyste pokračovali v editaci, nebo provedli download změn a ukončili editaci v režimu RUN, anebo ji ukončili bez provedení downloadu.

Zobrazení stavu programu

STEP 7 – Micro/WIN umožňuje monitorovat stav uživatelského programu během jeho provádění. Pokud sledujete stav programu, zobrazuje programový editor stav hodnot operandů instrukcí.

Chcete – li stav zobrazit, klikněte na tlačítko ”Stav programu” nebo vyberte příkaz menu **Odladit > Stav programu**.

Zobrazení stavu programu v LAD a FBD

STEP 7 – Micro/WIN nabízí pro zobrazení stavu programů LAD a FBD dvě možnosti:

- Stav na konci programového cyklu: STEP 7 – Micro/WIN získává hodnoty pro zobrazení stavu během více programových cyklů; pak aktualizuje zobrazení na obrazovce. Zobrazení stavu neodráží aktuální stav každého prvku v době provádění. Stav na konci programového cyklu neukazuje stav paměti L nebo akumulátorů.

  Při vyhodnocování stavu na konci programového cyklu jsou stavové hodnoty aktualizovány ve všech pracovních módech CPU.

- Stav při běhu programu: STEP 7 – Micro/WIN zobrazuje hodnoty networků tak, jak jsou jednotlivé prvky prováděny S7-200. Pro zobrazení stavu při běhu programu vyberte příkaz menu **Odladit > Při běhu programu**.

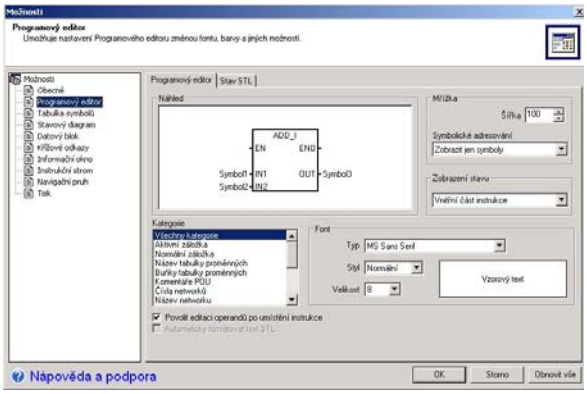
  Stavové hodnoty pro stav provádění jsou aktualizovány pouze tehdy, je – li CPU v režimu RUN.

Tip
STEP 7 – Micro/WIN obsahuje jednoduchý způsob, jak změnit stav proměnné. Vyberte proměnnou a klikněte pravým tlačítkem myši, aby se zobrazilo menu voleb.

Nastavení způsobu zobrazení stavu v programech LAD a FBD

STEP 7 – Micro/WIN má různé možnosti, jak zobrazit stav programu.

Jestliže chcete změnit možnosti zobrazení pro stavovou obrazovku, vyberte příkaz menu **Nástroje > Možnosti**, zvolte “Programový editor” a klikněte na záložku “Programový editor”, jak to ukazuje obrázek 8-2.



Obr. 8-2 Možnosti zobrazení stavu

Zobrazení stavu programu v STL

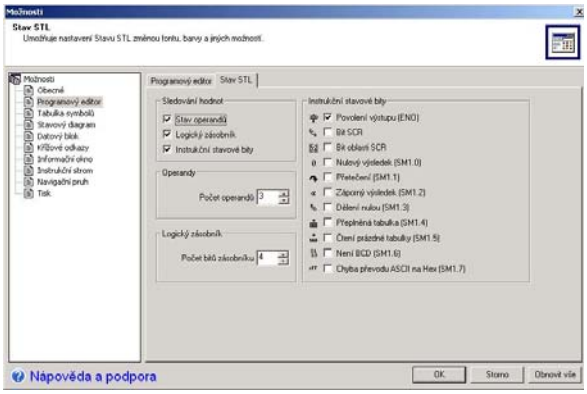
Stav provádění programu STL můžete sledovat při postupném prováděním jedné instrukce za druhou. V programu STL zobrazuje STEP 7–Micro/WIN ukazuje stav instrukcí, které jsou zobrazeny na obrazovce.

STEP 7–Micro/WIN shromažďuje stavové informace z automatu S7-200, počínaje prvním příkazem STL nahoře v okně editoru. Při přetáčení okna editoru směrem dolů jsou z automatu S7-200 vyčteny nové informace.

STEP 7–Micro/WIN trvale aktualizuje hodnoty na obrazovce. Chcete–li pozastavit aktualizace obrazovky, vyberte tlačítko ”Spuštěná pauza”. Současná data zůstanou na obrazovce, dokud nezrušíte volbu tlačítka ”Spuštěná pauza”.

Konfigurace zobrazovaných parametrů v programu STL

STEP 7–Micro/WIN umožňuje zobrazení stavu mnoha parametrů instrukcí STL. Vyberte příkaz menu **Nástroje > Možnosti**, zvolte “Programový editor” a klikněte na záložku “Stav STL”. Viz obrázek 8-3.



Obr. 8-3 Možnosti zobrazení stavu STL

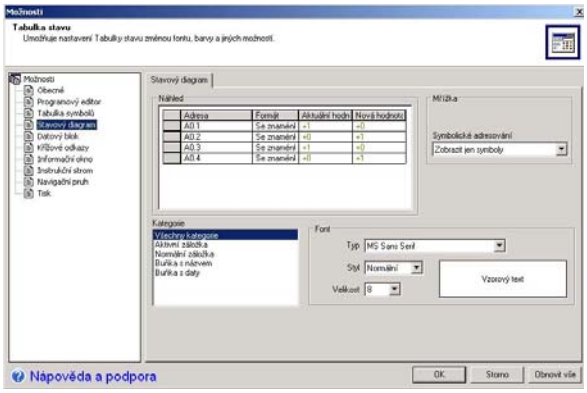
Použití stavového diagramu pro monitorování a modifikaci dat

Stavový diagram umožňuje čtení, zápis, vnucení (force) a monitorování proměnných v době, kdy S7-200 provádí uživatelský program. Chcete–li vytvořit stavovou tabulku, vyberte příkaz menu **Zobrazit > Komponenty > Stavový diagram**. Obrázek 8–4 8-4 ukazuje příklad stavového diagramu.

Můžete vytvořit několik stavových diagramů.

STEP 7–Micro/WIN obsahuje v nástrojové liště ikony pro manipulace se stavovým diagramem: ”Setřídit vzestupně”, ”Setřídit sestupně”, ”Jednorázové čtení”, ”Zápis všeho”, ”Force”, ”Zrušení force”, ”Zrušení všech force” a ”Čtení všech force”.

Chcete–li vybrat formát buňky, vyberte ji a klikněte na pravé tlačítko myši, aby se zobrazilo kontextové menu.



Obr. 8-4 Stavový diagram

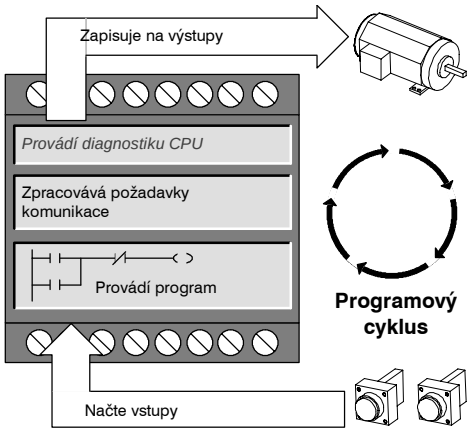
Force (vnucení) specifických hodnot

Programovatelný automat S7-200 umožňuje vnutit hodnoty všech, nebo jen některých vstupů a výstupů (bity I a Q). Dále můžete vnutit hodnoty do až 16 paměťových míst (V nebo M) nebo vnutit hodnoty analogových vstupů a výstupů (AI nebo AQ). Paměti V nebo M můžete vnutit hodnoty typu byte, word nebo double word. Analogové hodnoty mohou mít vnucenu pouze hodnotu typu word na sudých bytech, jako například AIW6 nebo AQW14. Všechny vnucené hodnoty jsou uloženy v permanentní paměti automatu S7-200.

Protože mohou být vnucená data během programového cyklu změněna (programem, cyklem aktualizace vstupů a výstupů, nebo komunikací), aplikuje S7-200 opakovaně vnucené hodnoty v různých dobách programového cyklu.

- *Čtení vstupů:* S7-200 aplikuje vnucené hodnoty na vstupy při jejich načítání.
- *Provádění řídicí logiky programu:* Automat S7-200 aplikuje vnucené hodnoty vždy, když přistupujete ke vstupům a výstupům. 16 vnucených hodnot je aplikováno po provedení programu.
- *Zpracování požadavků komunikace:* Automat S7-200 aplikuje vnucené hodnoty na všechny požadavky čtení a zápisu komunikace.
- *Zápis na výstupy:* S7-200 aplikuje vnucené hodnoty na výstupy při jejich zápisu.

Pro vnucení hodnot můžete použít stavový diagram. Chcete–li vnutit novou hodnotu, vložte ji do sloupce "Nová hodnota" ve stavové tabulce a klikněte na nástrojové liště na tlačítko "Force". Chcete–li vnutit existující hodnotu, vyberte ji ve sloupci "Současná hodnota"a klikněte na tlačítko "Force".



Obr. 8-5 Programový cyklus automatu S7-200

Tip

Funkce "Force" vyřadí instrukce "Okamžité čtení" nebo "Okamžitý zápis". Funkce "Force" také vyřadí tabulku výstupu, která byla konfigurována pro přechod do režimu STOP. Pokud přejde automat S7-200 do režimu STOP, odráží výstup vnucenou hodnotu, nikoliv hodnotu, která byla konfigurována v tabulce výstupů.

Zadání běhu programu po zadaný počet programových cyklů

Jako pomoc při ladění programu STEP 7–Micro/WIN spustit program na zadaný počet programových cyklů.

Můžete nechat S7-200 provést pouze první programový cyklus. To umožní monitorovat data v automatu S7-200 prvním programovém cyklu. Pro spuštění prvního programového cyklu vyberte příkaz menu **Odladit > První krok**.

Můžete také zadat, aby S7-200 prováděl program po několik programových cyklů (od 1 programového cyklu do 65.535 cyklů). To umožňuje sledovat program při změně proměnných. Chcete–li specifikovat počet programových cyklů, které mají být provedeny, vyberte příkaz menu **Odladit > Několik kroků**.

Návod pro odstraňování poruch hardwaru

Tabulka 8-1 Návod pro odstraňování poruch hardwaru automatu S7-200

Symptom	Možné příčiny	Možné řešení
Výstupy přestanou pracovat	• Elektrický impulz od zařízení poškodil výstup • Chyba uživatelského programu • Uvolněná nebo nesprávná kabeláž • Přetížení • Výstup je vnucený (force)	• Při připojení indukční zátěže (jako například motoru nebo relé) použijte vhodný obvod pro potlačení elektrických impulzů. Viz kapitola 3. • Opravte uživatelský program • Zkontrolujte a opravte kabeláž • Zkontrolujte zátěž podle jmenovitých hodnot výstupů • Zkontrolujte, zda není aktivní vnucení hodnot vstupů a výstupů
Rozsvítí se signálka SF (System Fault – systémová chyba) na S7-200 (červená)	Následující seznam popisuje nejběžnější chybové kódy a příčiny: • Chyba programování uživatelského programu – 0003 Chyba obvodu sledování funkce (watchdog) – 0011 Nepřímé adresování – 0012 Neplatná hodnota s plovoucí řádkovou čárkou – 0014 Chyba rozsahu • Elektrické rušení (0001 až 0009) • Poškozený modul (0001 až 0010)	Přečtěte si číslo kritické chyby; Informace o typu chyby najdete v příloze C: • Pokud jde o chybu programování, zkontrolujte použití instrukcí FOR, NEXT, JMP, LBL a "Porovnání". • Pokud jde o elektrické rušení: – Nahlédněte do návodu pro kabeláž v kapitole 3. Je velmi důležité, aby byl řídicí panel dobře uzemněn a aby nebyly vodiče vysokého napětí vedeny paralelně s vodiči nízkého napětí. – Připojte svorku M napájení snímačů 24 VDC na zem.
Nesvítí žádná LED dioda	• Přepálená pojistka • Obrácené napájecí kabely 24 V • Nesprávné napětí	Připojte k systému analyzátor vedení a zkontrolujte velikost a trvání přepěťových špiček. Na základě naměřených hodnot přidejte do systému příslušný typ přepěťové ochrany. Informace o instalaci kabeláže najdete v návodu v kapitole 3.
Přerušovaný provoz ve spojení s vysokoenergetickým zařízením	• Nesprávné uzemnění • Vedení kabeláže uvnitř rozvaděče • Příliš krátké čas vstupních filtrů	Nahlédněte do návodu pro kabeláž v kapitole 3. Je velmi důležité, aby řídicí systém dobře uzemněn a aby vodiče vysokého napětí vedeny paralelně s vodiči nízkého napětí. Připojte svorku M napájení snímačů 24 VDC na zem. Zvětšete zpoždění vstupních filtrů v systémovém datovém bloku.
Komunikační síť se poškodí při připojení na externí zařízení Je poškozen port počítače, port S7-200, nebo kabel PC/PPI	Komunikační kabel může vést pro nežádoucí proudy u zařízení bez galvanického oddělení, jako jsou programovatelné automaty, počítače nebo jiná zařízení, která jsou zapojena do sítě bez společného referenčního bodu. Tyto nežádoucí proudy mohou způsobit chyby v komunikaci nebo poškodit obvody.	• Návod pro kabeláž naleznete v kapitole 3 a návod pro síť v kapitole 7. • Použijte galvanicky oddělený PC/PPI kabel. • Pro propojení zařízení bez společného referenčního potenciálu použijte galvanicky oddělené repeatery RS-485. Informace o objednacích číslech pro vybavení S7-200 najdete v příloze NO TAG.
Jiné problémy s komunikací (STEP 7 – Micro/WIN)	Informace o síťové komunikaci najdete v kapitole 7.	
Ošetření chyb	Informace o chybových kódech najdete v příloze C.	

Řízení pohybu v otevřené smyčce u automatu S7-200

9

Automat S7-200 umožňuje tři metody řízení pohybu v otevřené smyčcey:

- Pulzní šířková modulace (PWM) – integrována do S7-200 pro řízení rychlosti, polohy nebo střídy
- Výstup série impulzů (PTO) – integrován do S7-200 pro řízení rychlosti a polohy
- Polohovací modul EM 253 – přídavný modul pro řízení rychlosti a polohy

Aby se zjednodušilo použití řízení polohy v aplikaci, obsahuje STEP7–Micro/WIN ”Průvodce polohováním”, který umožňuje během několika minut kompletně nakonfigurovat PWM, PTO nebo polohovací modul. Průvodce generuje polohovací instrukce, které můžete použít pro dynamické řízení rychlosti a polohy ve vaší aplikaci. Pro polohovací modul má STEP7–Micro/WIN také ovládací panel, který umožňuje kontrolovat, monitorovat a zkoušet pohybové operace.

Přehled kapitoly

Přehled 252
Použití výstupu PWM (pulzní šířková modulace) 253
Základní informace pro řízení polohy v otevřené smyčce pomocí krokových motorů a servomotorů 255
Instrukce vytvořené pomocí ”Průvodce polohováním” 260
Chybové kódy pro instrukce PTO 264
Funkce polohovacího modulu 265
Konfigurace polohovacího modulu 267
Polohovací instrukce vytvořené pomocí ”Průvodce polohováním” 273
Vzorové programy pro polohovací modul 285
Monitorování polohovacího modulu "Control panelem EM 253" 290
Chybové kódy pro polohovací modul a polohovací instrukce 292
Témata pro pokročilé 294
Vysvětlení režimů vyhledávání RP, podporovaných polohovacím modulem 303

Přehled

Automat S7-200 umožňuje tři metody řízení pohybu otevřené smyčky:

- Pulzní šířková modulace (PWM) – integrována do S7-200 pro řízení rychlosti, polohy nebo střídy
- Výstup série impulzů (PTO) – integrován do S7-200 pro řízení rychlosti a polohy
- Polohovací modul EM 253 – přídavný modul pro řízení rychlosti a polohy



Řízení polohy

S7-200 má dva digitální výstupy (Q0.0 a Q0.1), které lze konfigurovat pomocí ”Průvodce polohováním” pro použití buď jako výstup PWM, nebo výstup PTO. ”Průvodce polohováním” lze použít také pro konfiguraci polohovacího modulu EM 253.

Když je výstup nakonfigurován pro funkci PWM, doba cyklu výstupu je pevná a šířka pulzu nebo střída pulzu je řízena vaším programem. Změny šířky pulzu lze použít pro řízení rychlosti nebo polohy ve vaší aplikaci.

Když je výstup nakonfigurován pro funkci PTO, je generována série pulzů s 50% střídou pro řízení rychlosti a polohy v otevřené smyčce buď pro krokové motory, nebo pro servomotory. Integrovaná funkce PTO má pouze výstup pro sérii pulzů. Ovladače směru a omezení musí dodat váš aplikační program pomocí vstupů/výstupů zabudovaných do PLC, nebo je musí umožnit rozšiřovací moduly.

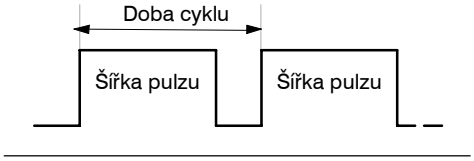
Polohovací modul EM 253 má jeden výstup pro sérii pulzů s integrovaným řízením směru, blokovací výstup a nulovací výstup. Obsahuje také vyhrazené vstupy, které umožňují konfiguraci modulu pro několik režimů provozu včetně automatického vyhledávání referenčního bodu. Modul poskytuje jednotné řešení pro řízení rychlosti a polohy v otevřené smyčce buď pro krokové motory, nebo pro servomotory.

Aby se zjednodušilo použití řízení polohy v aplikaci, obsahuje STEP7–Micro/WIN ”Průvodce polohováním” který umožňuje kompletně nakonfigurovat PWM, PTO nebo polohovací modul během několika minut. Průvodce generuje polohovací instrukce, které můžete použít pro dynamické řízení rychlosti a polohy ve vaší aplikaci. Pro polohovací modul má STEP7–Micro/WIN také ovládací panel, který umožňuje řídit, monitorovat a odzkoušet pohybové operace.

Použití výstupu PWM (pulzní šířková modulace)

Funkce PWM zajišťuje výstupní impulzy s pevnou periodou a proměnnou střídou. Po spuštění na specifikované frekvenci (doba cyklu) běží výstup PWM nepřetržitě. Šířka pulzu se mění podle potřeby požadovaného řízení. Střídu lze vyjádřit jako procento doby cyklu nebo jako časovou hodnotu odpovídající šířce impulzu. Šířka pulzu se může měnit od 0 % (žádný pulz, vždy vypnuto) do 100 % (žádný pulz, vždy zapnuto). Viz obrázek 9-1

Protože se výstup PWM může měnit od 0 do 100 %, je k dispozici digitální výstup, který je v mnoha směrech obdobný jako analogový výstup. Například výstup PWM lze použít pro řízení rychlosti motoru od zastavení po plnou rychlost, nebo ho lze použít pro řízení polohy ventilu od zavřeného po plně otevřený.

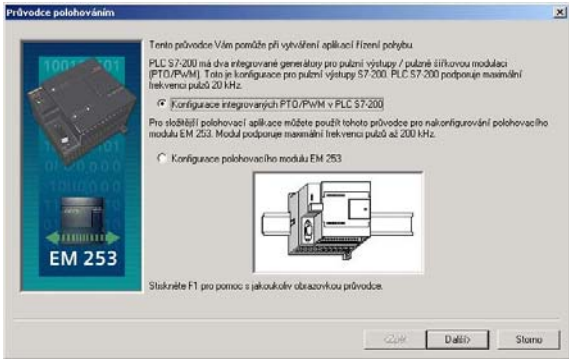


Obr. 9-1 Pulzní šířková modulace (PWM)

Konfigurace výstupu PWM

Pro konfiguraci některého ze zabudovaných výstupů pro řízení PWM použijte ”Průvodce polohováním” v softwaru STEP 7–Micro/WIN. Chcete–li spustit ”Průvodce polohováním”, klikněte nejprve na ikonu ”Nástroje” v navigační liště, a pak dvakrát na ikonu ”Průvodce polohováním”, nebo vyberte příkaz menu **Nástroje > Průvodce polohováním**. Viz obrázek 9-2.

1. Vyberte možnost konfigurovat vestavěnou funkci PTO/PWM pro PLC S7-200.
2. Vyberte výstup Q0.0 nebo Q0.1, který chcete konfigurovat jako výstup PWM.
3. Dále zvolte pulzní šířkovou modulaci (PWM) z rozbalovacího dialogového okna, zvolte čas v mikrosekundách nebo milisekundách a specifikujte dobu cyklu.
4. Průvodce ukončíte kliknutím na “Dokončit”.



Obr. 9-2 Konfigurace výstupu PWM

Průvodce vygeneruje jednu instrukci, kterou použijete k řízení střídy výstupu PWM.

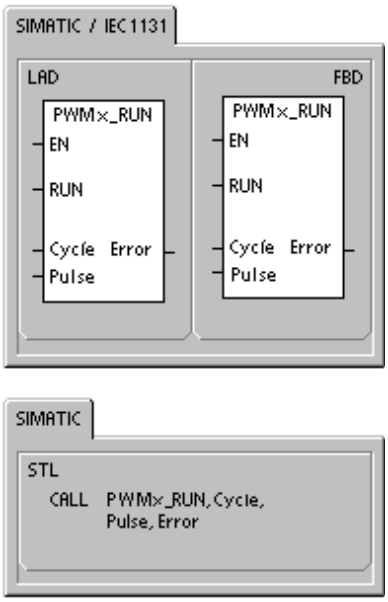
Instrukce PWMx_RUN

Instrukce PWMx_RUN umožňuje řídit střídu výstupu změnou šířky pulzu od 0 po šířku pulzu doby cyklu.

Vstup Cycle (cyklus) je hodnota word, která definuje dobu cyklu pro výstup PWM. Povolený rozsah je od 2 do 65535 jednotek času (mikrosekundy nebo milisekundy), který byl specifikován v průvodci.

Vstup Duty_Cycle (střída) je hodnota word, která definuje šířku impulzu pro výstup PWM. Povolený rozsah hodnot je od 0,0 do 65535 jednotek času (mikrosekundy nebo milisekundy), který byl specifikován v průvodci.

Error (Chyba) je hodnota bytu vrácená instrukcí PWMx_RUN, která označuje výsledek provedení. Možné chybové kódy jsou popsány v tabulce.



Tabulka 9-1 Parametry instrukce PWMx_RUN

Vstupy/Výstupy	Typy dat	Operandy
Cycle, Duty_Cycle	Word	IW, QW, VW, MW, SMW, SW, T, C, LW, AC, AIW, *VD, *AC, *LD, konstanta
Error	byte	IB, QB, VB, MBV, SMB, LB, AC, *VD, *AC, *LD, konstanta

Tabulka 9-2 Chybové kódy instrukce PWMx_RUN

Chybové kódy	Popis
0	Bez chyby, normální ukončení
1	Během pohybu vydán příkaz okamžitý STOP. Příkaz STOP úspěšně vykonán.

Základní informace pro řízení polohy v otevřené smyčce pomocí krokových motorů a servomotorů

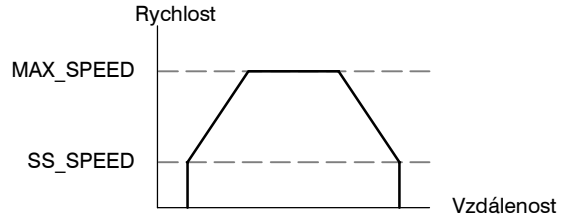
PTO zabudovaný do PLC S7-200 i polohovací modul EM 253 používají výstup série pulzů pro řízení rychlosti i polohy krokového motoru nebo servomotoru.

Použití PTO nebo modulu pro řízení polohy v otevřené smyčce vyžaduje odbornost v oblasti řízení pohybu. Tato kapitola nemá za úkol vzdělávat začátečníky v této oblasti. Podává však základní informace, které vám pomohou při použití "Průvodce polohováním" v softwaru STEP 7–Micro/WIN ke konfiguraci PTO nebo modulu pro vaši aplikaci.

Maximální rychlosti a spouštěcí/zastavovací rychlosti

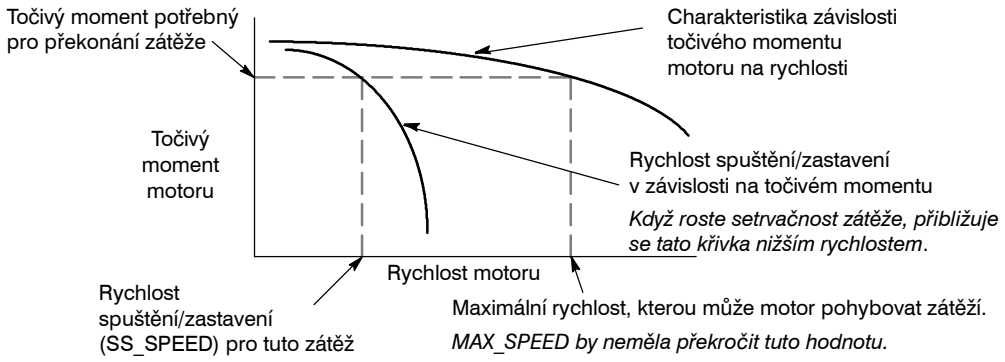
Průvodce vás vyzve, abyste specifikovali maximální rychlost (MAX_SPEED) a rychlost pro spuštění/zastavení (SS_SPEED): Viz obrázek 9-3.

- MAX_SPEED: Vložte hodnotu optimální provozní rychlosti pro vaši aplikaci v rámci možností točivého momentu motoru. Točivý moment potřebný pro pohyb zatížení je určen třením, setrvačností a dobou zrychlení/zpomalení.
- "Průvodce polohováním" vypočítá a zobrazí minimální rychlost, která může být řízena polohovacím modulem na základě specifikované MAX_SPEED.
- Pro výstup PTO musíte specifikovat požadovanou spouštěcí/zastavovací rychlost. Protože při každém provedení pohybu je generován alespoň jeden cyklus při spouštěcí/zastavovací rychlosti, použijte spouštěcí/zastavovací rychlost, jejíž perioda je menší než doba zrychlení/zpomalení.
- SS_SPEED: Vložte hodnotu v rámci možností motoru, která dokáže pohybovat s danou zátěží při nízké rychlosti. Pokud je hodnota SS_SPEED příliš nízká, mohou se motor i zátěž chvět nebo se pohybovat krátkými skoky na začátku a na konci posunu. Pokud je hodnota SS_SPEED příliš vysoká, motor může při spuštění ztrácet pulzy a zátěž může při pokusu o zastavení motor přetížit.



Obr. 9-3 Maximální rychlost a rychlost spuštění/zastavení

Technické specifikace motoru uvádějí rychlost spuštění/zastavení pro motor a dané zatížení různě. Běžně se používá hodnota SS_SPEED 5 % až 15 % hodnoty MAX_SPEED. Při volbě správné rychlosti pro vaši aplikaci pomůže technická specifikace motoru. Obrázek 9-4 ukazuje typickou momentovou charakteristiku motoru.

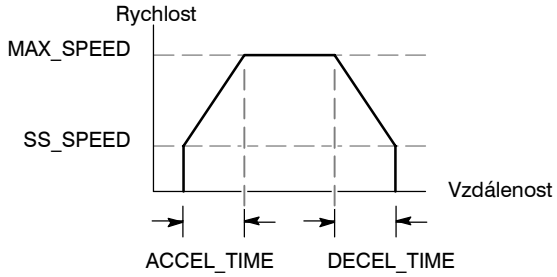


Obr. 9-4 Typická momentová charakteristika motoru

Vložení doby zrychlení a zpomalení

Jako součást konfigurace nastavte dobu zrychlení a zpomalení. Implicitní nastavení pro dobu zrychlení i pro dobu zpomalení je 1 sekunda. Motory mohou běžně pracovat s dobou kratší než 1 sekunda. Viz obrázek 9-5. Specifikujte následující doby v milisekundách:

- ACCEL_TIME: Doba potřebná k tomu, aby motor zrychlil z SS_SPEED na MAX_SPEED. Implicitní nastavení = 1000 ms
- DECEL_TIME: Doba potřebná k tomu, aby motor zpomalil z MAX_SPEED na SS_SPEED. Implicitní nastavení = 1000 ms



Obr. 9-5 Doba zrychlení a zpomalení



Tip

Doby zrychlení a zpomalení motoru jsou stanoveny pomocí pokusu a omylu. Měli byste začít zadáním velké hodnoty. Optimalizujte tato nastavení pro aplikaci postupným zkracováním těchto dob, dokud se motor nezačne zastavovat.

Konfigurace profilů pohybu

Profil je předdefinovaný popis pohybu, skládající se z jedné nebo více hodnot rychlosti pohybu, které způsobují změnu polohy od výchozího do koncového bodu. Abyste mohli používat PTO nebo modul, nemusíte definovat žádný profil. ”Průvodce polohováním” dodává instrukce, abyste mohli použít řízení pohybů bez spuštění profilu.

Profil se programuje v krocích sestávajících se zrychlení/zpomalení na cílovou rychlost s následným pevným počtem pulzů při cílové rychlosti. V případě jednokrokových pohybů nebo posledního kroku pohybu existuje také zpomalení z cílové rychlosti (poslední cílová rychlost) do zastavení.

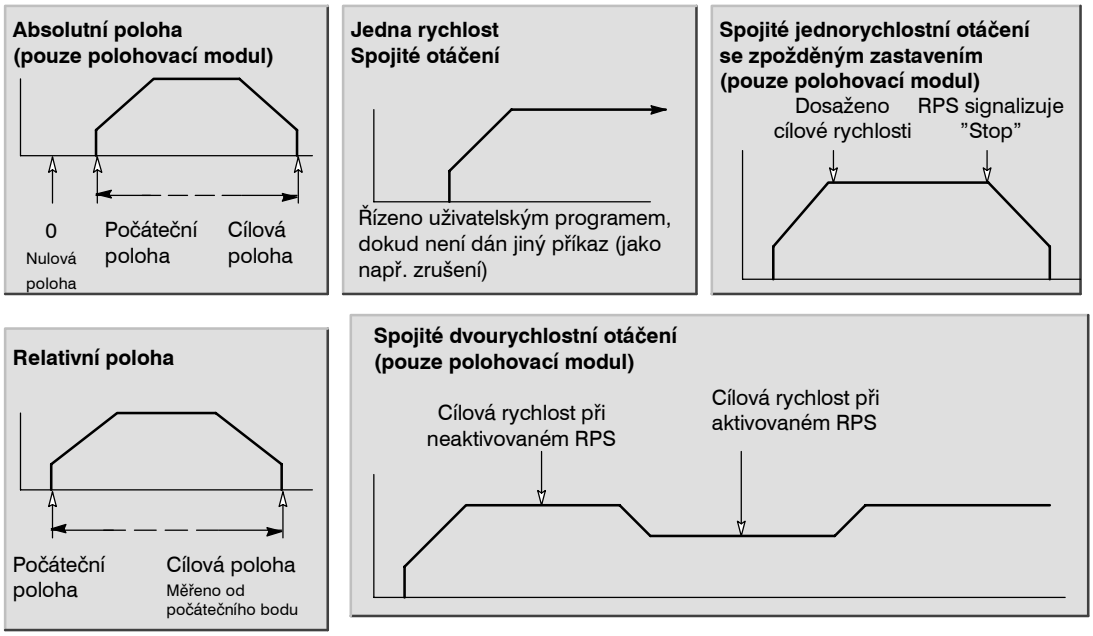
PTO podporuje maximálně 100 profilů a modul maximálně 25 profilů.

Definice profilů pohybu

”Průvodce polohováním” vás provede definicí profilu pohybu, kde definujete každý profil pohybu pro vaši aplikaci. Pro každý profil vyberte provozní režim a definujte specifické vlastnosti každého jednotlivého kroku profilu. ”Průvodce polohováním” také umožňje definovat pro každý profil symbolický název tím, že ho jednoduše vložíte při definování profilu.

Výběr pracovního režimu profilu

Nakonfigurujte profil podle požadovaného pracovního režimu. PTO podporuje relativní polohu a jednorychlostní spojité otáčení. Polohovací modul podporuje absolutní polohu, relativní polohu, jednorychlostní- spojité otáčení a dvourychlostní- spojité otáčení. Na obrázku 9-6 jsou ukázány různé pracovní režimy.



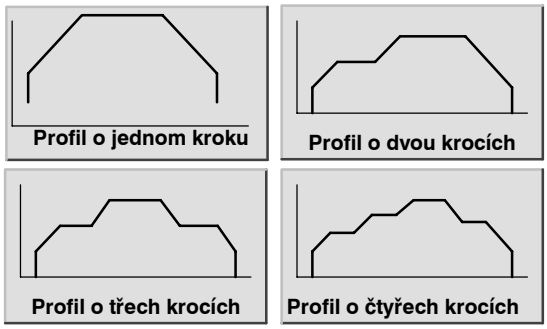
Obr. 9-6 Výběr režimu polohovacího modulu

Tvorba kroků profilu

Krok je vzdálenost, o kterou se posune nástroj, včetně vzdálenosti pokryté během doby zrychlení a zpomalení. V případě PTO je v každém profilu povoleno maximálně 29 kroků. Modul podporuje v každém profilu maximálně 4 kroky.

Pro každý krok specifikujte cílovou rychlost a koncovou polohu nebo počet impulzů. Další kroky se vkládají jeden po druhém. Obrázek 9-7 zobrazuje profil s jedním krokem, dvěma kroky, třemi kroky a čtyřmi kroky.

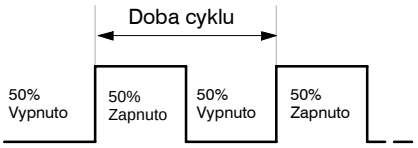
Všimněte si, že profil s jedním krokem má jeden segment konstantní rychlosti, profil se dvěma kroky má dva segmenty konstantní rychlosti atd. Počet kroků v profilu odpovídá počtu segmentů konstantní rychlosti profilu.



Obr. 9-7 Příklad profilů pohybu

Použití výstupu PTO

PTO generuje výstup obdélníkového signálu (50% střída) pro specifikovaný počet pulzů. Frekvence doby cyklu každého pulzu se mění lineárně s frekvencí během zrychlení a zpomalení a zůstává stálá během konstantní frekvence pohybu. Jakmile byl generován specifikovaný počet pulzů, výstup PTO se vypne a další pulzy se negenerují, dokud nebude načtena nová aplikace. Viz obrázek 9-8.



Obr. 9-8 Pulzní výstup (PTO)

Konfigurace výstupu PTO

Pro konfiguraci některého ze zabudovaných výstupů funkce PTO použijte ”Průvodce polohováním” v softwaru STEP 7–Micro/WIN. Chcete–li spustit ”Průvodce polohováním”, klikněte nejprve na ikonu ”Nástroje” v navigační liště a pak dvakrát na ikonu ”Průvodce polohováním”, nebo vyberte příkaz menu **Nástroje > Průvodce polohováním**.

1. Vyberte možnost konfigurovat vestavěnou funkci PTO/PWM pro PLC S7-200.
2. Vyberte výstup Q0.0 nebo Q0.1, který chcete konfigurovat jako výstup PTO.
3. Vyberte výstup lineární série pulzů (PTO) z rozbalovacího dialogového okna.
4. Jestliže si přejete monitorovat počet pulzů generovaných pomocí PTO, klikněte na zaškrtávací políčko vysokorychlostního čítače.
5. Zadejte MAX_SPEED a SS_SPEED do příslušných editačních polí.
6. Zadejte dobu zrychlení a zpomalení do příslušných editačních polí.
7. Definování nového pohybového profilu umožníte kliknutím na tlačítko nového profilu na obrazovce definice profilu. Vyberte požadovaný pracovní režim.

   Pro profil relativní polohy:

   Zadejte cílovou rychlost a počet pulzů. Pak můžete zobrazit grafické znázornění pohybu kliknutím na tlačítko ”Nakreslit krok”.

   Je–li potřeba více než jeden krok, klikněte na tlačítko nového profilu a zadejte požadované informace o kroku.

   Pro jednu rychlost, spojité otáčení:

   Zadejte hodnotu jedné rychlosti do editačního pole.

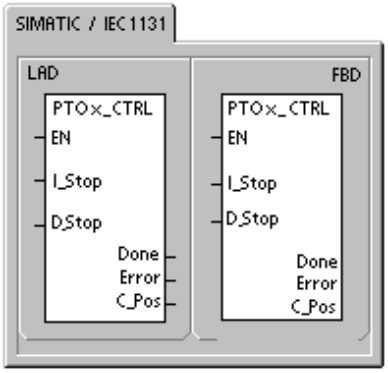
   Jestliže si přejete ukončit jednorychlostní, spojitý otáčivý pohyb, klikněte na zaškrtávací políčko ”Programovat podprogram” a zadejte počet pulzů pohybu po zastavení.
8. Definujte tolik profilů a kroků, kolik potřebujete k provedení požadovaného pohybu.
9. Průvodce pak ukončíte kliknutím na “Dokončit”.

Instrukce vytvořené pomocí ”Průvodce polohováním”

”Průvodce polohováním” usnadňuje řízení zabudovaného PTO vytvořením čtyř unikátních podprogramů instrukcí. Každá polohovací instrukce má prefix ”PTO*x*_”, kde *x* znamená umístění modulu.

Podprogram PTOx_CTRL

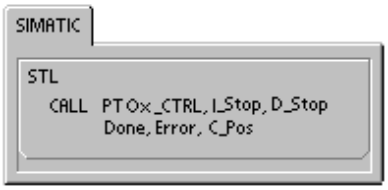
Podprogram PTOx_CTRL (Control) zapíná a inicializuje výstup PTO pro použití s krokovým motorem nebo servomotorem. Tento podprogram použijte ve svém programu pouze jednou a zajistěte jeho provedení při každém programovém cyklu. Jako vstup pro EN vstup vždy použijte SM0.0.



Vstup I_STOP (okamžitý STOP) je booleovský vstup. Když je tento vstup ve stavu 0, PTO funguje normálně. Když je tento vstup ve stavu 1, PTO ihned ukončí vysílání pulzů.

Vstup D_STOP (zpomalený STOP) je booleovský vstup. Když je tento vstup ve stavu 0, PTO funguje normálně. Když je tento vstup ve stavu 1, PTO generuje sérii pulzů, která zpomalí motor do zastavení.

Výstup Done je booleovský výstup. Když je bit Done nastaven na 1, znamená to, že CPU provedl podprogram.



Když je bit Done ve stavu 1, chybový byte hlásí normální ukončení bez chyby nebo s chybovým kódem. Definice chybových kódů jsou popsány v tabulce 9-7.

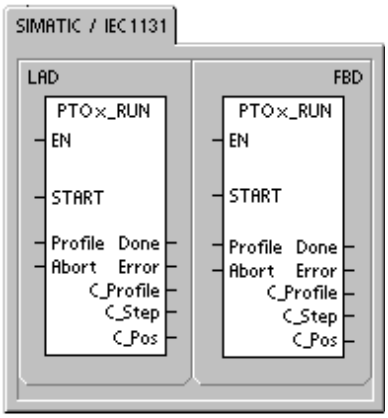
Parametr C_Pos obsahuje aktuální polohu modulu jako počet impulzů, jestliže v průvodci byl zapnut HSC. Jinak je aktuální poloha vždy 0.

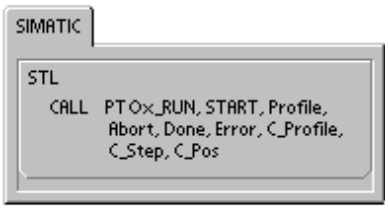
Tabulka 9-3 Parametry instrukce PTOx_CTRL

Vstupy/Výstupy	Typy dat	Operandy
I_STOP	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
D_STOP	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD
C_Pos	DWORD	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD

Podprogram PTOx_RUN

Podprogram PTOx_RUN (Spuštění profilu) přikazuje PLC, aby provedl pohyb podle konkrétního profilu uloženého v tabulce konfigurace profilu.





Zapnutí bitu EN povoluje podprogram. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit Done signalizovat, že je provádění podprogramu ukončeno.

Zapnutí parametru START inicializuje provedení profilu. Pro každý programový cyklus, když je zapnutý parametr START a PTO není právě aktivní, instrukce aktivuje PTO. Aby bylo jisté, že je vyslán pouze jediný příkaz, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

Parametr ”Profile” (Profil) obsahuje číslo symbolického názvu profilu pohybu.

Zapnutí parametru ”Abort” (Zrušení) dá polohovacímu modulu příkaz, aby zastavil aktuální profil a zpomaloval motor až do zastavení.

Parametr ”Done” (Dokončeno) se zapne ve chvíli, kdy modul dokončí tuto instrukci.

Parametr ”Error” (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-7.

Parametr C_Profile obsahuje profil, který v současné době polohovací modul provádí.

Parametr C_Step obsahuje krok profilu, který se právě provádí.

Parametr C_Pos obsahuje aktuální polohu modulu jako počet pulzů, jestliže v průvodci byl zapnut HSC. Jinak je aktuální poloha vždy 0.

Tabulka 9-4 Parametry instrukce PTOx_RUN

Vstupy/Výstupy	Typy dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Profile	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD, konstanta
Abort, Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error, C_Profile, C_Step	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD
C_Pos	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD

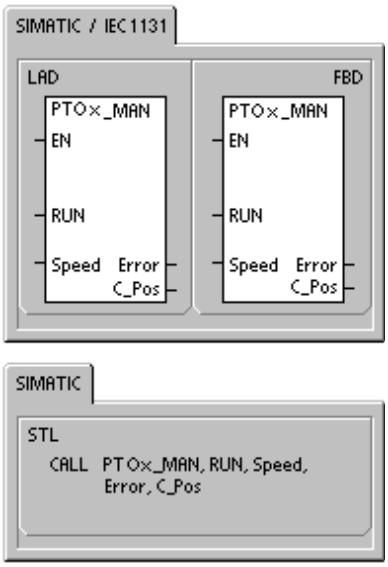
Podprogram PTOx_MAN

Podprogram PTOx_MAN (Manuální režim) uvede PTO do manuálního režimu. Ten umožní spuštění motoru, zastavení motoru a chod motoru při různých rychlostech. Když je podprogram PTOx_MAN povolen, provedení žádného dalšího podprogramu PTO není povoleno.

Povolení parametru RUN (chod/zastavení) přikazuje PTO zrychlit na specifikovanou rychlost (parametr rychlosti). Hodnotu parametru rychlosti můžete měnit v době chodu motoru. Blokování parametru RUN dá PTO příkaz zpomalit až do zastavení motoru.

Parametr "Speed" (Rychlost) určuje rychlost, když je povolen RUN. Rychlost je hodnota DINT pro pulzy/sekundy. Tento parametr můžete měnit v době chodu motoru.

Parametr "Error" (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-7.



Parametr C_Pos obsahuje aktuální polohu modulu jako počet impulzů, jestliže v průvodci byl zapnut HSC. Jinak je aktuální poloha vždy 0.

Tabulka 9-5 Parametry instrukce PTOx_MAN

Vstupy/Výstupy	Typy dat	Operandy
RUN	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
SPEED	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD, konstanta
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD
C_Pos	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD

Tip
PTO nemusí zareagovat na malé změny parametru Speed, obzvláště je–li nakonfigurovaná doba zrychlení nebo zpomalení krátká a rozdíl mezi nakonfigurovanou maximální rychlostí a rychlostí pro spuštění/zastavení je velký.

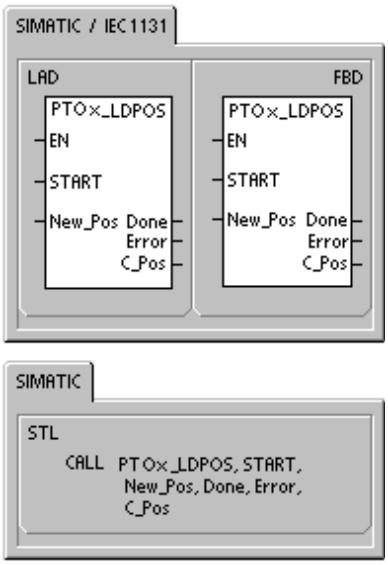
Instrukce PTOx_LDPOS

Instrukce PTOx_LDPOS (Nahrát polohu) mění hodnotu aktuální polohy v čítači pulzů PTO na novou. Tuto instrukci můžete také použít pro stanovení nové nulové polohy pro kterýkoliv příkaz pohybu.

Zapnutí vstupu EN instrukci povoluje. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit "Done" signalizovat, že je provádění instrukce ukončeno.

Zapnutí parametru START načte novou polohu do čítače pulzů PTO. Při každém programovém cyklu, kdy je parametr START zapnutý a PTO právě není v činnosti, instrukce načte novou polohu do čítače pulzů PTO. Aby bylo jisté, že je vyslán pouze jediný příkaz, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

Parametr New_Pos uvádí novou hodnotu, která nahradí hlášenou aktuální hodnotu polohy. Hodnota polohy je vyjádřena jako počet pulzů.



Parametr "Done" (Dokončeno) se zapne ve chvíli, kdy modul dokončí tuto instrukci.

Parametr "Error" (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-7.

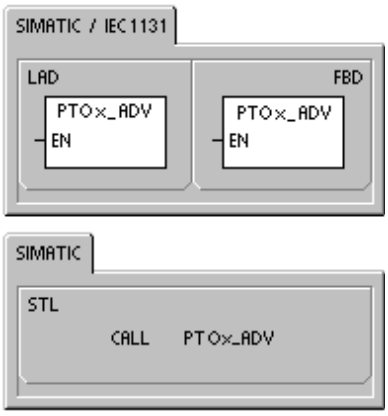
Parametr C_Pos obsahuje aktuální polohu modulu jako počet impulzů, jestliže v průvodci byl zapnut HSC. Jinak je aktuální poloha vždy 0.

Tabulka 9-6 Parametry instrukce PTOx_LDPOS

Vstupy/Výstupy	Typy dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
New_Pos, C_Pos	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD
Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD

Podprogram PTOx_ADV

Podprogram PTOx_ADV zastavuje aktuální profil spojitého pohybu a vyšle počet pulzů, specifikovaný v definici profilu v průvodci. Tento podprogram se vytvoří, jestliže jste v "Průvodci polohováním" specifikovali alespoň jeden jednorychlostní spojitý otáčivý pohyb se zapnutou funkcí PTOx_ADV.



Chybové kódy pro instrukce PTO

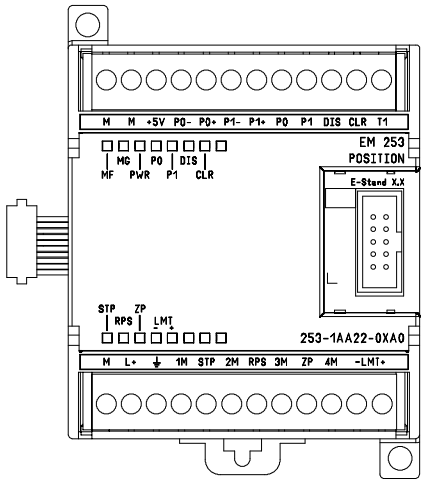
Tabulka 9-7 Chybové kódy instrukcí PTO

Chybové kódy	Popis
0	Bez chyby, normální ukončení.
1	Během pohybu vydán příkaz okamžitý STOP. Příkaz STOP úspěšně vykonán.
2	Během pohybu vydán příkaz zpomalený STOP. Příkaz STOP úspěšně vykonán.
3	Zjištěna chyba provedení v generátoru pulzů nebo ve formátu tabulky PTO.
127	Instrukce HSC, PLS nebo PTO způsobila chybu ENO během provádění.
128	Tento požadavek nelze zpracovat. CPU zpracovává jiný požadavek, nebo pro tento požadavek nebyl pulz START.
129	Příkazy okamžitý STOP i zpomalený STOP byly vyslány současně.
130	Instrukci PTO je právě předáván příkaz STOP.
132	Pro zadaný profil není nakonfigurován profilový blok.

Funkce polohovacího modulu

Polohovací modul přidává do řídícího systému funkce a výkon, které potřebujete pro řízení polohy v otevřené smyčce:

- ❏ Umožňuje vysokorychlostní řízení s rozsahem od 20 pulzů za sekundu až do 200 000 pulzů za sekundu.
- ❏ Podporuje hladké (S křivka) nebo lineární zrychlení a zpomalení.
- ❏ Obsahuje konfigurovatelný měřicí systém, který umožňuje vkládat data v inženýrských jednotkách (např. milimetry nebo centimetry) nebo jako počet pulzů.
- ❏ Má konfigurovatelnou kompenzaci vůle.
- ❏ Podporuje absolutní, relativní a manuální způsob řízení polohy.
- ❏ Nabízí kontinuální provoz.
- ❏ Nabízí až 25 pohybových profilů, každý s až 4 změnami rychlosti.
- ❏ Nabízí čtyři různé režimy vyhledávání referenčního bodu s volbou počátečního směru vyhledávání a směru konečného přístupu pro každou sekvenci.
- ❏ Má odnímatelné svorkovnice kabeláže pro snadnou instalaci a demontáž.



Obr. 9-9 Polohovací modul EM 253

Pro vytvoření celé konfigurace a všech profilových informací použitých polohovacím modulem se používá STEP 7–Micro/WIN. Tyto informace jsou ukládány do automatu S7-200 spolu s programovými bloky. Protože jsou všechny potřebné informace pro řízení polohy uloženy v S7-200, můžete polohovací modul vyměnit, aniž byste museli modul znovu programovat nebo konfigurovat.

Automat S7-200 rezervuje osm bitů registru obrazu výstupů (paměť Q) pro rozhraní s polohovacím modulem. Uživatelský program v S7-200 tyto bity používá pro řízení práce polohovacího modulu. Těchto osm výstupních bitů není spojeno s žádným fyzickým výstupem polohovacího modulu.

Polohovací modul má pět digitálních vstupů a čtyři digitální výstupy, které představují rozhraní pro řízení pohybu. Viz tabulku 9-8. Tyto vstupy a výstupy jsou fyzické vstupy a výstupy polohovacího modulu. V příloze A je uvedena podrobná specifikace polohovacího modulu; v ní jsou i schémata zapojení pro připojení polohovacího modulu k nejběžnějším zesilovačům motoru.

Tabulka 9-8 Vstupy a výstupy polohovacího modulu

Signál	Popis
STP	Vstup STP způsobí, že modul zastaví pohyb, který právě probíhá. Požadovanou práci STP můžete zvolit v "Průvodci polohováním".
RPS	Vstup RPS (Reference Point Switch – snímač referenčního bodu) stanovuje referenční bod neboli výchozí polohu pro operace s absolutním pohybem.
ZP	Vstup ZP (Zero Pulse – nulový pulz) pomáhá stanovit referenční bod nebo výchozí polohu. Běžně zesilovač motoru vysílá ZP jednou za otáčku motoru.
LMT+ LMT–	Vstupy LMT+ a LMT– stanovují maximální meze pohybu při posunu. "Průvodce polohováním" umožňuje práci vstupu LMT+ a LMT– konfigurovat.
P0 P1 P0+, P0– P1+, P1–	P0 a P1 jsou tranzistorové pulzní výstupy s otevřeným kolektorem, které řídí pohyb a směr motoru. P0+, P0– a P1+, P1– jsou diferenční pulzní výstupy, které mají stejné funkce jako P0 a P1, ale dávají špičkovou kvalitu signálu. Výstupy s otevřeným kolektorem a diferenční výstupy jsou všechny aktivní ve stejnou dobu. V závislosti na požadavcích rozhraní zesilovače motoru si vyberete, kterou sadu pulzních výstupů budete používat.
DIS	DIS (Disable – Zablokování) je tranzistorový výstup s otevřeným kolektorem, používaný pro zablokování nebo povolení zesilovače motoru.
CLR	CLR (Clear – Smazání) je tranzistorový výstup s otevřeným kolektorem, používaný pro mazání registru počtu pulzů servopohonu.

Programování polohovacího modulu

STEP 7 – Micro/WIN umožňuje snadno konfigurovat a programovat polohovací modul. Jednoduše proveďte tyto kroky:

1. Nakonfigurujte polohovací modul. STEP 7 – Micro/WIN obsahuje ”Průvodce polohováním”, pomocí kterého vypracujete tabulku konfigurace, profilů a polohovacích instrukcí. Informace o konfigurování polohovacího modulu naleznete na str. 270 – Konfigurace polohovacího modulu.
2. Přezkoušejte činnost polohovacího modulu. STEP 7 – Micro/WIN má řídicí panel EM 253, pomocí kterého lze přezkoušet zapojení vstupů a výstupů, konfiguraci polohovacího modulu a činnost profilů pohybu. Informace o řídícím panelu EM 253 najdete na straně 290;.
3. Vytvořte program, který má S7-200 vykonávat. ”Průvodce polohováním” automaticky vytvoří polohovací instrukce, které do programu vložíte. Informace o instrukcích najdete na straně 273. Do programu vložte následující instrukce:
   - Chcete – li povolit polohovací modul, vložte instrukci POSx_CTRL. Abyste zajistili, že se tato instrukce provede v každém programovém cyklu, použijte SM0.0 (vždy zapnuto).
   - Chcete – li motor posunout do konkrétní polohy, použijte instrukce POSx_GOTO nebo POSx_RUN. Instrukce POSx_GOTO provede posun na místo specifikované vstupy uživatelského programu. Instrukce POSx_RUN provede profily pohybu, které jste nakonfigurovali pomocí ”Průvodce polohováním.
   - Abyste pro daný pohyb mohli použít absolutní souřadnice, musíte stanovit nulovou polohu vaší aplikace. Pro stanovení nulové polohy použijte instrukce POSx_RSEEK nebo POSx_LDPOS.
   - Ostatní instrukce vytvořené ”Průvodcem polohováním” poskytují funkce pro typické aplikace a pro konkrétní aplikaci jsou jen volitelné.
4. Zkompilujte uživatelský program a proveďte download systémového, datového a programového bloku do S7-200.



Tip

Informace o připojení polohovacího modulu k několika běžným zesilovačům krokových motorů najdete v příloze A.



Tip

Pro odsouhlasení implicitních nastavení v ”Průvodci polohováním” nastavte DIP přepínače na regulátoru krokového motoru na 10.000 pulzů na otáčku.

Konfigurace polohovacího modulu



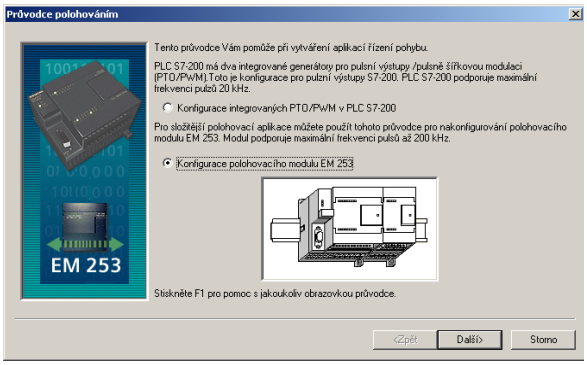
Řízení polohy

Aby polohovací modul řídil aplikaci, musíte pro něj vytvořit tabulku konfigurace a profilů. Díky ”Průvodci polohováním” je postup konfigurace rychlý a snadný – průvodce vás tímto procesem provede krok za krokem. Detailní informace o tabulce konfigurace a profilů najdete v Tématech pro pokročilé na straně 294.

”Průvodce polohováním” umožní vytvořit tabulku konfigurace a profilu offline. Konfiguraci můžete vytvořit, aniž byste byli připojeni na CPU S7-200 s instalovaným polohovacím modulem.

Abyste mohli spustit ”Průvodce polohováním”, musí být váš projekt zkompilován a nastaven do režimu symbolického adresování.

Chcete–li spustit ”Průvodce polohováním”, klikněte nejprve na ikonu ”Nástroje” v navigační liště a pak dvakrát na ikonu ”Průvodce polohováním” nebo vyberte příkaz menu **Nástroje > Průvodce polohováním**.



Obr. 9-10 ”Průvodce polohováním”

Pro konfigurování polohovacího modulu použijte ”Průvodce polohováním” STEP 7–Micro/WIN. Zvolte možnost pro konfigurování polohovacího modulu EM 253.

Zadání umístění modulu

Specifikujte polohu slotu modulu (modul 0 až modul 6). Jestliže je STEP 7–Micro/WIN připojen k PLC, stačí jen kliknout na tlačítko ”Přečíst moduly”. U CPU S7-200 s firmwarem před verzí 1.2 musí být modul nainstalován vedle CPU.

Volba typu měření

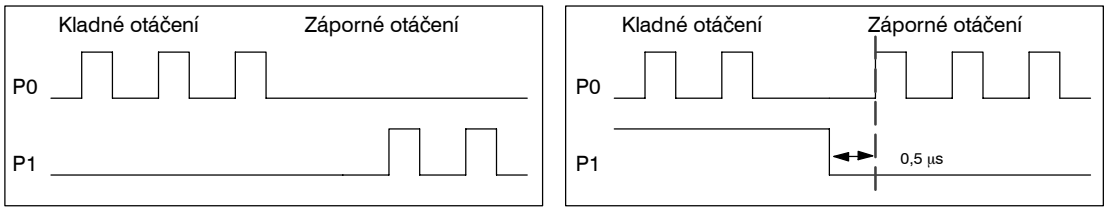
Zvolte systém měření. Můžete si zvolit inženýrské jednotky nebo pulzy. Jestliže zvolíte pulzy, nepožadují se žádné další informace. Jestliže zvolíte inženýrské jednotky, požaduje se počet pulzů na jednu otáčku motoru (údaje najdete ve specifikaci motoru nebo pohonu), základní jednotka měření (např. palec, stopa, milimetr nebo centimetr) a vzdálenost daná jednou otáčkou motoru.

- STEP7–Micro/WIN podporuje ovládací panel EM253, který umožňuje měnit počet jednotek na otáčku poté, co byl polohovací modul nakonfigurován.
- Pokud byste někdy v pozdější době měnili systém měření, musíte v ”Průvodci polohováním” smazat celou konfiguraci včetně všech generovaných instrukcí. Následně musíte vložit volby, které jsou v souladu s novým systémem měření.

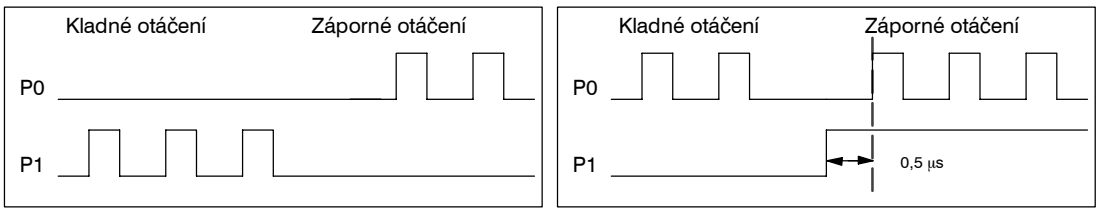
Editace implicitní konfigurace vstupů a výstupů

Volbou tlačítka "Další možnosti" změníte nebo zobrazíte implicitní konfiguraci integrovaných vstupů/výstupů.

- ❏ Volbu aktivní úrovně (vysoká nebo nízká) provedete na záložce "Aktivní úrovně". Jestliže je úroveň signálu nastavena na "Vysoký", je logická 1 čtena tehdy, protéká–li vstupem proud. Jestliže je úroveň nastavena na "Nízký", je logická 1 čtena tehdy, když vstupem proud neprotéká. Úroveň logická 1 se vždy interpretuje tak, že znamená aktivní podmínku. LED diody svítí, jestliže vstupem protéká proud, bez ohledu na úroveň aktivace. (Implicitní nastavení = aktivní vysoká úroveň)
- ❏ Záložku "Časy vstupního filtru" použijte k volbě časové konstanty filtru (0,20 ms až 12,80 ms) pro vstupy STP, RPS, LMT+ a LMT–. Zvýšení časové konstanty filtru eliminuje více šumu, ale také zpomaluje dobu odezvy na změnu stavu signálu. (Implicitně nastavená hodnota = 6,4 ms)
- ❏ Záložku "Výstupy pro impulzy a směr" použijte pro volbu polarity výstupů a volbu metody řízení směru. Účinky polarity a metody řízení směru vidíte na obrázku 9-11 a 9-12.



Obr. 9-11 Možnosti volby směru pohybu u kladné polarity



Obr. 9-12 Možnosti volby směru pohybu u záporné polarity



Varování

Řídicí zařízení mohou selhat a dostat se do nebezpečného stavu, který může mít za následek nepředvídatelné chování řízeného zařízení. Toto nepředvídatelné chování by mohlo mít za následek smrt nebo vážné zranění osob nebo poruchu zařízení.

Funkce omezení a zastavení polohovacího modulu jsou implementovány logikou a neposkytují úroveň ochrany danou elektromechanickými ovládacími prvky. Použijte funkci nouzového zastavení, elektromechanické spínače nebo jiné doplňující ochranné obvody nezávislé na CPU S7-200.

Konfigurace odezvy modulu na fyzické vstupy

Dále zvolte odezvu modulu na vstupy LMT+, LMT– a STP. Pomocí rozbalovacího dialogového okna vyberte: nereaguje (ignoruje vstupní podmínku), zastaví se zpomalením (implicitní) nebo okamžitě.

Zadání maximální rychlosti pro spuštění a zastavení

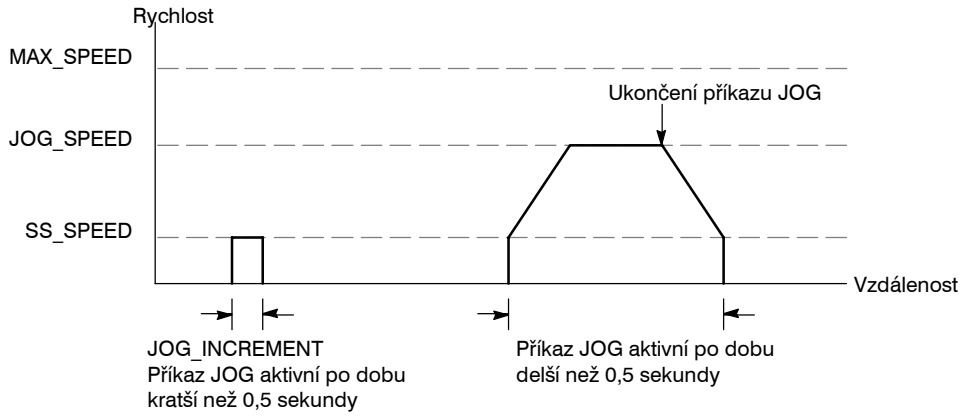
Zadejte maximální rychlost (MAX_SPEED) a rychlost pro spuštění/zastavení (SS_SPEED).

Zadání parametrů pro skokový posuv

Dále zadejte hodnoty JOG_SPEED a JOG_INCREMENT.

- ❏ JOG_SPEED: JOG_SPEED (rychlost motoru pro skokový posuv) je maximální rychlost, které je možné docílit, dokud zůstává aktivní příkaz JOG.
- ❏ JOG_INCREMENT: Vzdálenost, o kterou je nástroj posunut krátkodobým příkazem JOG.

Obrázek 9-13 ukazuje funkci příkazu ”Jog”. Když polohovací modul dostane příkaz ”Jog”, spustí časovač. Jestliže příkaz ”Jog” skončí dříve, než uplyne 0,5 sekundy, posune polohovací modul nástroj o vzdálenost specifikovanou v JOG_INCREMENT rychlostí definovanou v SS_SPEED. Jesliže je příkaz ”Jog” stále aktivní i po uplynutí 0,5 sekundy, zrychlí polohovací modul až na JOG_SPEED. Pohyb pokračuje, dokud není ukončen příkaz ”Jog”. Polohovací modul pak zastavení se zpomalením. Příkaz ”Jog” můžete aktivovat buď z ”Control panelu EM 253”, nebo polohovací instrukcí.



Obr. 9-13 Znázornění funkce příkazu JOG

Zadání doby zrychlení

Zadejte dobu zrychlení a zpomalení do editačních polí.

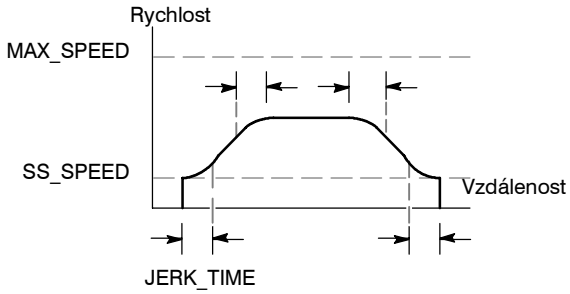
Zadání doby zaoblení

Pro pohyby o jednom kroku zadejte kompenzaci doby rázu. Kompenzace rázu dává plynulejší řízení polohy zmenšením rázu (rychlosti změny) ve zrychlovací a zpomalovací části pohybového profilu. Viz obrázek 9-14.

Kompenzace rázu je také známá jako ”profilování křivky S”. Tato kompenzace se aplikuje stejně na počáteční i koncovou část křivky zrychlení i křivky zpomalení. Kompenzace rázu se neaplikuje na počáteční a konečný krok mezi nulovou rychlostí a SS_SPEED.

Kompenzaci rázu specifikujete vložením hodnoty času (JERK_TIME). Je to doba nutná pro změnu zrychlení z nuly na maximální rychlost zrychlení. Delší doba rázu způsobí plynulejší chod s menším prodloužením celkové doby cyklu, než k jakému by se dospělo snížením ACCEL_TIME a DECEL_TIME. Nulová hodnota znamená, že kompenzace nemá být aplikována.

(Implicitně nastavená hodnota = 0 ms)



Obr. 9-14 Kompenzace rázu

Tip
Vhodná implicitní hodnota pro JERK_TIME je 40 % ACCEL_TIME.

Konfigurace parametrů referenčního bodu a vyhledávání

Zvolte používání nebo nepoužívání referenčního bodu pro vaši aplikaci.

- Jestliže aplikace vyžaduje, aby pohyby začaly od absolutní polohy nebo aby k ní byly vztahovány, musíte stanovit referenční bod (RP) nebo nulovou polohu, která váže měření polohy na známý bod ve fyzickém systému.
- Jestliže se použije referenční bod, budete potřebovat definovat způsob automatického přemístění referenčního bodu. Proces automatického umístění referenčního bodu se nazývá "Vyhledání referenčního bodu". Definování procesu vyhledání referenčního bodu vyžaduje dva kroky v průvodci.

  Zadejte rychlosti vyhledávání referenčního bodu (rychlost rychlého vyhledávání a pomalého vyhledávání). Definujte počáteční směr vyhledávání a konečný směr přístupu k referenčnímu bodu. Pomocí tlačítka "Možnosti RP" zadejte hodnoty offsetu referenčního bodu a kompenzace vůle.

  RP_FASTje výchozí rychlost, kterou modul použije při provádění příkazu vyhledávání RP. Běžně se hodnota RP_FAST rovná přibližně 2/3 hodnoty MAX_SPEED.

  RP_SLOW je rychlost konečného přiblížení k RP. Abyste ho neminuli, používá se pro přibližování k RP pomalejší rychlost. Běžně se hodnota RP_SLOW rovná hodnotě SS_SPEED.

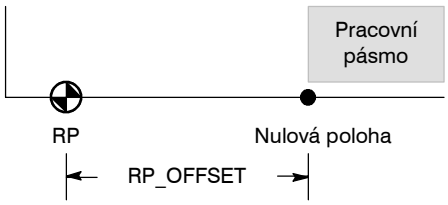
  RP_SEEK_DIR je počáteční směr operace vyhledávání RP. Běžně je to směr z pracovního pásma do blízkosti RP. Mezní spínače hrají při definování oblasti, která je prohledávána na RP, důležitou roli. Při provádění operace vyhledávání RP může mít kontakt s mezním spínačem za následek obrácení směru, který umožňuje pokračovat v hledání. (Implicitní nastavení = záporný)

  RP_APPR_DIRje směr konečného přiblížení k RP. Aby se snížila vůle a zajistila větší přesnost, přiblížení k referenčnímu bodu by mělo být ve stejném směru jako posun od RP k pracovnímu pásmu. (Implicitní nastavení = kladný)

❑ ”Průvodce polohováním” má pro referenční bod volbu pro pokročilé, která umožňuje specifikovat offset RP (RP_OFFSET), což je vzdálenost mezi RP a nulovou polohou. Viz obrázek 9-15.

RP_OFFSET: Vzdálenost RP od nulové polohy fyzického odměřovacího systému. Implicitně nastavená hodnota = 0

Kompenzace vůle: Vzdálenost, o kterou se musí motor pohnout, aby eliminoval vůli (např. v přechodech) v systému při změně směru. Kompenzace vůle je vždy kladná hodnota. Implicitně nastavená hodnota = 0



Obr. 9-15 Vztah mezi RP a nulovou polohou

Vyberte sekvenci vyhledávání referenčního bodu.

❑ Polohovací modul má vstup pro spínač referenčního bodu (RPS), který se používá pro hledání RP. RP se určí metodou lokalizace přesné polohy vzhledem k RPS. RP může být umístěn ve středu aktivní zóny RPS, může být umístěn na hraně aktivní zóny RPS nebo může ležet o specifikovaný počet přechodů vstupu nulových pulzů (ZP) dále od hrany aktivní zóny RPS.

Můžete nakonfigurovat sekvenci, kterou polohovací modul používá pro hledání referenčního bodu. Obrázek 9-16 ukazuje zjednodušený diagram implicitní sekvence vyhledávání RP. Pro sekvenci hledání RP si můžete vybrat následující volby:

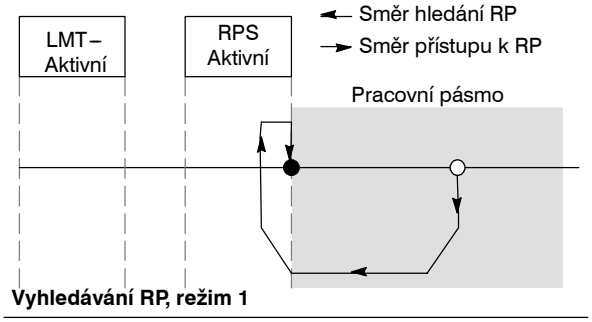
Vyhledávání RP, režim 0: Neprovádí sekvenci ”RP Seek”

Vyhledávání RP, režim 1: RP je tam, kde vstup RPS přejde do aktivního stavu při přiblížení ze strany pracovního pásma. (implicitně)

Vyhledávání RP, režim 2: RP leží ve středu aktivní oblasti vstupu RPS.

Vyhledávání RP, režim 3: RP je umístěn mimo aktivní oblast vstupu RPS. RP_Z_CNT specifikuje, kolik má být obdrženo impulzů na vstupu ZP po tom, co RPS přestane být aktivní.

Vyhledávání RP, režim 4: RP leží obvykle uvnitř aktivní oblasti vstupu RPS. RP_Z_CNT specifikuje, kolik má být obdrženo impulzů na vstupu ZP po tom, co RPS přejde do aktivního stavu.



Obr. 9-16 Implicitní sekvence hledání RP (zjednodušená)

Tip

Aktivní oblast RPS (což je vzdálenost, po kterou vstup RPS zůstává aktivní) musí být větší než vzdálenost potřebná ke zpomalení z rychlosti RP_FAST na rychlost RP_SLOW. Pokud je tato vzdálenost příliš krátká, generuje polohovací modul chybu.

Příkazový byte

Dále zadejte adresu bytu Q pro příkazový byte. Příkazový byte je adresa 8 digitálních výstupů rezervovaných v registru obrazu vstupů a výstupů pro rozhraní polohovacího modulu. Popis číslování vstupů a výstupů najdete na obrázku 4-10 v kapitole 4.

Definice profilů pohybu

Definování nového profilu umožníte kliknutím na tlačítko nového profilu na obrazovce definice profilu pohybu. Vyberte požadovaný pracovní režim.

- Pro profil absolutní polohy:

  Zadejte cílovou rychlost a koncovou polohu. Pak můžete zobrazit grafické znázornění pohybu kliknutím na tlačítko "Nakreslit krok".

  Je–li potřeba více než jeden krok, klikněte na tlačítko nového profilu a zadejte požadované informace o kroku.

- Pro profil relativní polohy:

  Zadejte cílovou rychlost a koncovou polohu. Pak můžete zobrazit grafické znázornění pohybu kliknutím na tlačítko "Nakreslit krok".

  Je–li potřeba více než jeden krok, klikněte na tlačítko nového profilu a zadejte požadované informace o kroku.

- Pro jednu rychlost, spojité otáčení:

  Zadejte hodnotu jedné rychlosti do editačního pole.

  Zvolte směr otáčení.

  Jestliže si přejete ukončit jednorychlostní spojitý otáčivý pohyb pomocí vstupu RPS, zatrhněte zaškrtávací políčko.

- Pro dvě rychlosti, spojité otáčení:

  Zadejte hodnotu cílové rychlosti, když je RPS ve stavu 1, do editačního pole.

  Zadejte hodnotu cílové rychlosti, když je RPS ve stavu 0, do editačního pole.

  Zvolte směr otáčení.

Definujte tolik profilů a kroků, kolik potřebujete k provedení požadovaného pohybu.

Ukončení konfigurace

Když jste nakonfigurovali operace polohovacího modulu, jednoduše klikněte na "Dokončit" a "Průvodce polohováním" provede následující úlohy:

- Vloží konfiguraci modulu a tabulku profilů do datového bloku programu S7-200.
- Vytvoří tabulku globálních symbolů pro parametry pohybu.
- Do programového bloku projektu přidá podprogramy pro pohybové instrukce pro použití ve vaší aplikaci.

Chcete–li modifikovat jakýkoliv prvek konfigurace nebo hodnotu profilu, můžete znovu spustit "Průvodce polohováním" .



Tip

Protože "Průvodce polohováním" provádí změny v programovém, datovém i systémovém bloku, zkontrolujte si, je–li proveden download všech tří bloků do CPU S7-200. Jinak by polohovací modul nemusel mít všechny komponenty programu, které potřebuje pro správný provoz.

Polohovací instrukce vytvořené pomocí ”Průvodce polohováním”

”Průvodce polohováním” usnadňuje řízení polohovacího modulu tím, že vytváří jednoznačné podprogramy na základě polohy modulu a voleb konfigurace, které jste si vybrali. Každá polohovací instrukce má prefix ”POSx_”, kde *x* znamená umístění modulu. Protože je každá polohovací instrukce podprogramem, používá 11 polohovacích instrukcí 11 podprogramů.



Tip

Díky polohovacím instrukcím se zvětšuje část paměti potřebná pro uživatelský program až o 1700 bytů. Abyste zmenšili spotřebovanou část paměti, můžete vymazat nepoužívané polohovací instrukce. Chcete–li obnovit vymazanou polohovací instrukci, jednoduše znovu spusťte ”Průvodce polohováním”.

Návod pro užívání polohovacích instrukcí

Musíte zajistit, aby byla v danou dobu aktivní pouze jedna polohovací instrukce.

Instrukce POSx_RUN a POSx_GOTO můžete provést z podprogramu přerušení. Je ale velmi důležité, abyste se nepokoušeli spustit instrukci v podprogramu přerušení, pokud modul právě zpracovává jiný příkaz. Když spustíte některou instrukci v podprogramu přerušení, můžete použít výstupy instrukce POSx_CTRL pro sledování toho, kdy polohovací modul pohyb dokončí.

”Průvodce polohováním” automaticky nakonfiguruje hodnoty parametru rychlosti (Speed a C_Speed) a parametru polohy (Pos nebo C_Pos) v souladu se systémem odměňování, který jste vybrali. V případě pulzů mají tyto parametry hodnotu DINT. U inženýrských jednotek jsou parametry hodnoty REAL. Například: výběr centimetrů (cm) uloží parametry polohy jako hodnoty REAL v centimetrech a parametry rychlostí jako hodnoty REAL v centimetrech za sekundu (cm/s).

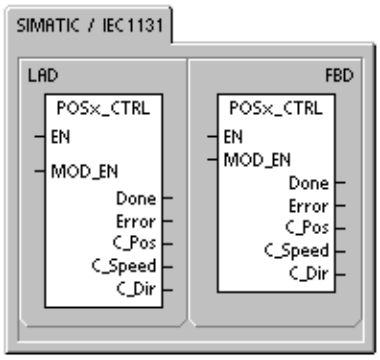
Pro konkrétní úkoly řízení polohy jsou zapotřebí následující polohovací instrukce:

- ❏ Vložte do svého programu instrukci POSx_CTRL a použijte kontakt SM0.0 pro její provedení v každém programovém cyklu.
- ❏ Chcete–li specifikovat pohyb k absolutní poloze, musíte pro stanovení nulové polohy nejprve použít instrukci POSx_RSEEK nebo instrukci POSx_LDPOS.
- ❏ Pro pohyb na konkrétní místo na základě vstupu z uživatelského programu použijte instrukci POSx_GOTO.
- ❏ Pro spuštění profilů pohybu, které jste nakonfigurovali pomocí ”Průvodce polohováním”, použijte instrukci POSx_RUN.

Ostatní polohovací instrukce jsou volitelné.

Instrukce POSx_CTRL

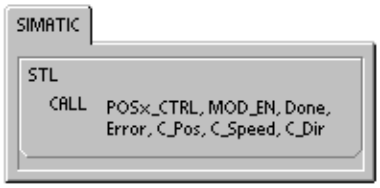
Instrukce POSx_CTRL (Řízení) povoluje a inicializuje polohovací modul tím, že mu automaticky přikazuje, aby nahrál tabulku konfigurace profilů pokaždé, když automat S7-200 přejde do režimu RUN.



Tuto instrukci použijte ve svém projektu pouze jednou a zajistěte, aby ji váš program volal při každém programovém cyklu. Jako vstup pro parametr EN použijte SM0.0 (Vždy zapnut).

Parametr MOD_EN musí být zapnutý, aby povolil ostatním polohovacím instrukcím posílat příkazy polohovacímu modulu. Pokud se parametr MOD_EN vypne, polohovací modul přeruší jakýkoliv probíhající příkaz.

Výstupní parametry instrukce POSx_CTRL ukazují aktuální stav polohovacího modulu.



Parametr ”Done” (Dokončeno) přejde do stavu log. 1, když polohovací modul dokončí jakoukoliv instrukci.

Parametr ”Error” (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

Parametr C_Pos je aktuální poloha modulu. V závislosti na jednotkách měření je tato hodnota buď počet pulzů (DINT), nebo počet inženýrských jednotek (REAL).

Parametr C_Speed uvádí aktuální rychlost modulu. Pokud jste systém odměřování polohovacího modulu nakonfigurovali na pulzy, C_Speed je hodnota DINT, která obsahuje počet pulzů/sekundu. Pokud jste systém odměřování nakonfigurovali na inženýrské jednotky, C_Speed je hodnota REAL, která obsahuje vybrané inženýrské jednotky/sekundu (REAL).

Parametr C_Dir označuje aktuální směr motoru.

Tabulka 9-9 Parametry instrukce POSx_CTRL

Vstupy/Výstupy	Typ dat	Operandy
MOD_EN	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Done, C_Dir	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD
C_Pos, C_Speed	DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD

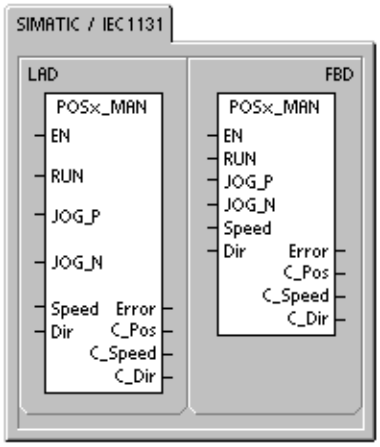
Tip

Polohovací modul načítá tabulku konfigurace profilů pouze při studeném startu nebo když dostane příkaz nahrát konfiguraci.

- Pokud použijete pro modifikaci konfigurace ”Průvodce polohováním”, dá instrukce POSx_CTRL automaticky příkaz polohovacímu modulu, aby nahrál tabulku konfigurace profilů pokaždé, když CPU S7-200 přejde do režimu RUN.
- Pokud pro modifikaci konfigurace použijete ”Control panel pro EM 253”, kliknutí na tlačítko ”Aktualizace konfigurace” přikáže polohovacímu modulu, aby nahrál novou tabulku konfigurace profilu.
- Jestliže pro modifikaci konfigurace použijete jiný způsob, musíte dát polohovacímu modulu také příkaz ”Znovu nahrát konfiguraci”, aby nahrál tabulku konfigurace profilů. V opačném případě polohovací modul nadále používá starou tabulku konfigurace profilů.

Instrukce POSx_MAN

Instrukce POSx_MAN (Manuální režim) uvede polohovací modul do manuálního režimu, což umožňuje použít různou rychlost motoru nebo jej uvést do pomalého chodu kladným nebo záporným směrem. V době, kdy je povolena instrukce POSx_MAN, jsou přípustné pouze instrukce POSx_CTRL a POSx_DIS.



Naráz můžete povolit pouze jeden ze vstupů RUN, JOG_P nebo JOG_N.

Povolení parametru RUN (Chod/Zastavení) dá polohovacímu modulu příkaz zrychlit na specifikovanou rychlost (parametr "Speed") a směr (parametr "Dir"). V době chodu motoru můžete měnit hodnotu parametru "Speed", ale parametr "Dir" musí zůstat konstantní. Blokování parametru RUN dá polohovacímu modulu příkaz zpomalit až do zastavení motoru.



Povolení parametru JOG_P (Manuální chod kladným směrem) nebo parametru JOG_N (Manuální chod záporným směrem) dá polohovacímu modulu příkaz manuálního chodu v kladném nebo záporném směru. Pokud parametr JOG_P nebo JOG_N zůstane povolen na dobu kratší než 0,5 sekundy, generuje polohovací modul pulzy pro posun o vzdálenost specifikovanou v JOG_INCREMENT. Pokud parametr JOG_P nebo JOG_N zůstane povolen po dobu 0,5 sekundy nebo déle, začne polohovací modul zrychlovat na specifikované JOG_SPEED.

Parametr "Speed" (Rychlost) určuje rychlost, když je povolen RUN. Pokud jste nastavili systém odměřování polohovacího modulu na pulzy, má rychlost pro pulzy/sekundu hodnotu DINT. Pokud jste nastavili systém odměřování polohovacího modulu na inženýrské jednotky, má rychlost pro jednotky/sekundu hodnotu REAL. Tento parametr můžete měnit v době chodu motoru.



Tip

Polohovací modul nemusí zareagovat na malé změny parametru Speed, obzvláště je–li nakonfigurovaná doba zrychlení nebo zpomalení krátká a rozdíl mezi nakonfigurovanou maximální rychlostí a rychlostí pro spuštění/zastavení je velký.

Parametr "Dir" (Směr) určuje směr pohybu, když je povolen RUN. Tuto hodnotu nemůžete měnit, pokud je povolen parametr RUN.

Parametr "Error" (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

Parametr C_Pos obsahuje aktuální polohu modulu. V závislosti na vybraných jednotkách odměřování je tato hodnota buď počet pulzů (DINT), nebo počet inženýrských jednotek (REAL).

Parametr C_Speed obsahuje aktuální rychlost modulu. V závislosti na vybraných jednotkách odměřování je tato hodnota buď počet pulzů/sekundu (DINT), nebo počet inženýrských jednotek/sekundu (REAL).

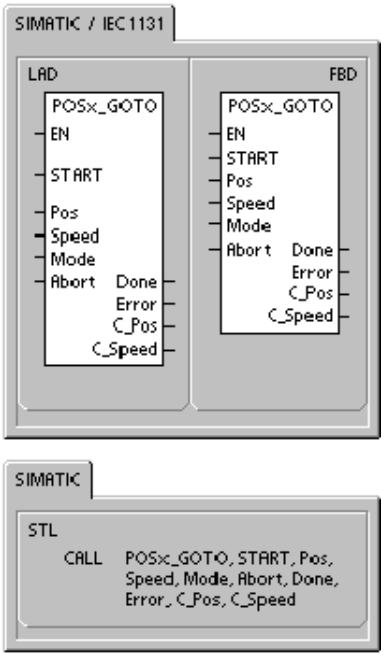
Parametr C_Dir označuje aktuální směr motoru.

Tabulka 9-10 Parametry instrukce POSx_MAN

Vstupy/Výstupy	Typ dat	Operandy
RUN, JOG_P, JOG_N	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Speed	DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD, konstanta
Dir, C_Dir	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD
C_Pos, C_Speed	DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD

Instrukce POSx_GOTO

Instrukce POSx_GOTO přikazuje polohovacímu modulu přejít do požadované polohy.



Zapnutí vstupu EN instrukci povoluje. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit DONE signalizovat, že je instrukce ukončena.

Zapnutí parametru START vyšle příkaz GOTO polohovacímu modulu. Při každém programovém cyklu, kdy je parametr START zapnutý a polohovací modul právě není v činnosti, vyšle instrukce polohovacímu modulu příkaz GOTO. Aby bylo jisté, že je vyslán pouze jediný příkaz GOTO, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

Parametr "Pos" obsahuje hodnotu, která představuje buď cíl pohybu (pro absolutní pohyb), nebo délku pohybu (pro relativní pohyb). V závislosti na vybraných jednotkách měření je tato hodnota buď počet pulzů (DINT), nebo inženýrských jednotek (REAL).

Parametr "Speed" určuje maximální rychlost tohoto pohybu. V závislosti na jednotkách odměřování je tato hodnota buď počet pulzů/sekundu (DINT), nebo inženýrských jednotek/sekundu (REAL).

Parametr "Mode" (Režim) volí typ pohybu:
- 0 – Absolutní poloha
- 1 – Relativní poloha
- 2 – Spojitý kladný pohyb s jednou rychlostí
- 3 – Spojitý záporný pohyb s jednou rychlostí

Parametr "Done" (Dokončeno) se zapne ve chvíli, kdy polohovací modul dokončí tuto instrukci.

Zapnutí parametru "Abort" (Zrušení) dá polohovacímu modulu příkaz, aby zastavil aktuální profil a zpomaloval motor až do zastavení.

Parametr "Error" (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

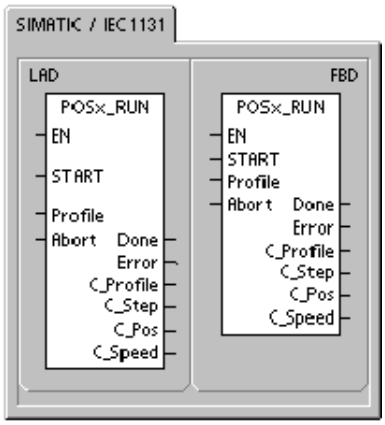
Parametr C_Pos obsahuje aktuální polohu modulu. V závislosti na jednotkách měření je tato hodnota buď počet pulzů (DINT), nebo počet inženýrských jednotek (REAL).

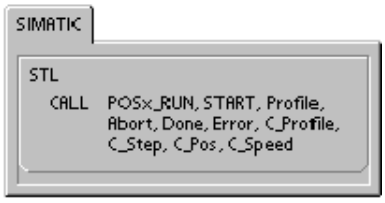
Parametr C_Speed aktuální rychlost modulu. V závislosti na jednotkách odměřování je tato hodnota buď počet pulzů/sekundu (DINT), nebo inženýrských jednotek/sekundu (REAL).

Tabulka 9-11 Parametry instrukce POSx_GOTO

Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Pos, Speed	DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD, konstanta
Mode	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD, konstanta
Abort, Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD
C_Pos, C_Speed	DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD

Instrukce POSx_RUN





Instrukce POSx_RUN (Spuštění profilu) přikazuje polohovacímu modulu, aby provedl pohybu podle konkrétního profilu uloženého v tabulce konfigurace profilu.

Zapnutí vstupu EN instrukci povoluje. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit ”Done” signalizovat, že je provádění instrukce ukončeno.

Zapnutí parametru START vyšle příkaz RUN polohovacímu modulu. Při každém programovém cyklu, kdy je parametr START zapnutý a polohovací modul právě není v činnosti, vyšle instrukce polohovacímu modulu příkaz RUN. Aby bylo jisté, že je vyslán pouze jediný příkaz, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

Parametr ”Profile” (Profil) obsahuje číslo symbolického názvu profilu pohybu. Můžete také vybrat pohybové příkazy pro pokročilé (118 až 127). Informace o pohybových příkazech najdete v tabulce 9-26.

Zapnutí parametru ”Abort” (Zrušení) dá polohovacímu modulu příkaz, aby zastavil aktuální profil a zpomaloval motor až do zastavení.

Parametr ”Done” (Dokončeno) se zapne ve chvíli, kdy modul dokončí tuto instrukci.

Parametr ”Error” (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

Parametr C_Profile obsahuje profil, který v současné době polohovací modul provádí.

Parametr C_Step obsahuje krok profilu, který se právě provádí.

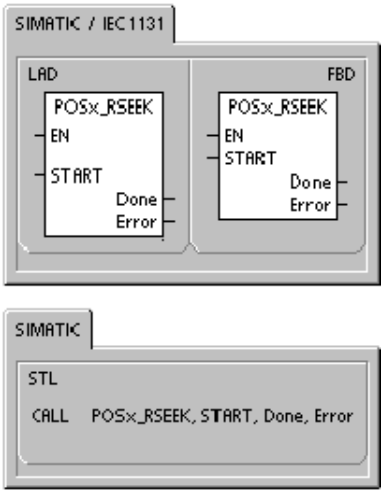
Parametr C_Pos obsahuje aktuální polohu modulu. V závislosti na jednotkách měření je tato hodnota buď počet pulzů (DINT), nebo počet inženýrských jednotek (REAL).

Parametr C_Speed aktuální rychlost modulu. V závislosti na jednotkách odměřování je tato hodnota buď počet pulzů/sekundu (DINT), nebo inženýrských jednotek/sekundu (REAL).

Tabulka 9-12 Parametry instrukce POSx_RUN

Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Profile	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD, konstanta
Abort, Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error, C_Profile, C_Step	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD
C_Pos, C_Speed	DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD

Instrukce POSx_RSEEK



Instrukce POSx_RSEEK (Vyhledání referenčního bodu) iniciuje operaci hledání referenčního bodu s použitím způsobu hledání z tabulky konfigurace profilu. Když polohovací modul lokalizuje referenční bod a pohyb se zastaví, nahraje polohovací modul hodnotu parametru RP_OFFSET na aktuální polohu a generuje 50milisekundový pulz na výstupu CLR.

Implicitní hodnota RP_OFFSET je 0. Můžete použít ”Průvodce polohováním”, ”Control panel EM253” nebo instrukci POSx_LDOFF (Nahrát offset), abyste změnili hodnotu RP_OFFSET.

Zapnutí vstupu EN instrukci povoluje. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit ”Done” signalizovat, že je provádění instrukce ukončeno.

Zapnutí parametru START vyšle příkaz RSEEK polohovacímu modulu. Při každém programovém cyklu, kdy je parametr START zapnutý a polohovací modul právě není v činnosti, vyšle instrukce polohovacímu modulu příkaz RSEEK. Aby bylo jisté, že je vyslán pouze jediný příkaz, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

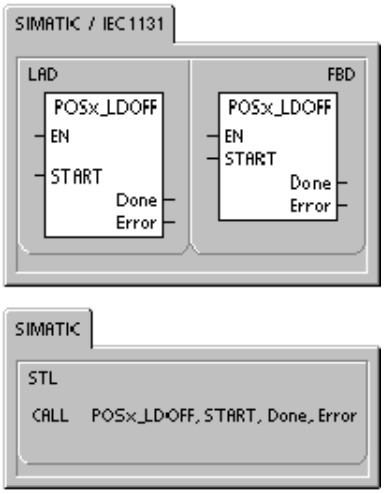
Parametr ”Done” (Dokončeno) se zapne ve chvíli, kdy modul dokončí tuto instrukci.

Parametr ”Error” (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

Tabulka 9-13 Parametry instrukce POSx_RSEEK

Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD

Instrukce POSx_LDOFF



Instrukce POSx_LDOFF (Nahrát offset referenčního bodu) stanovuje novou nulovou polohu, která leží na jiném místě, než je poloha referenčního bodu.

Před provedením této instrukce musíte nejprve určit polohu referenčního bodu. Musíte také stroj uvést do počáteční polohy. Když instrukce vyšle příkaz LDOFF, vypočítá polohovací modul offset mezi počáteční polohou (aktuální poloha) a polohou referenčního bodu. Pak polohovací modul uloží vypočítaný offset do parametru RP_OFFSET a nastaví aktuální polohu na 0. Tím je počáteční poloha určena jako nulová poloha.

V případě, že motor ztratí přehled o své poloze (například při ztrátě napájení nebo když je poloha motoru změněna ručně), můžete použít pro automatické opětovné stanovení nulové polohy instrukci POSx_RSEEK.

Zapnutí vstupu EN instrukci povoluje. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit "Done" signalizovat, že je provádění instrukce ukončeno.

Zapnutí parametru START vyšle příkaz LDOFF polohovacímu modulu. Při každém programovém cyklu, kdy je parametr START zapnutý a polohovací modul právě není v činnosti, vyšle instrukce polohovacímu modulu příkaz LDOFF. Aby bylo jisté, že je vyslán pouze jediný příkaz, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

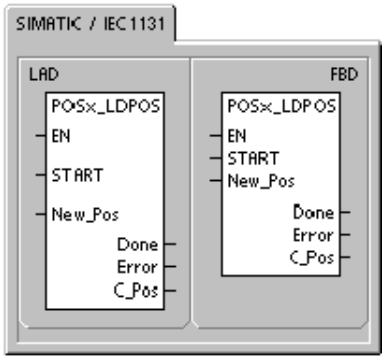
Parametr "Done" (Dokončeno) se zapne ve chvíli, kdy modul dokončí tuto instrukci.

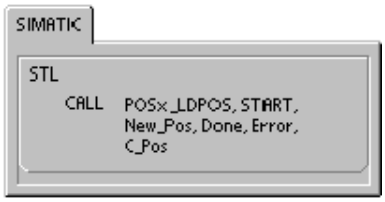
Parametr "Error" (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

Tabulka 9-14 Parametry instrukce POSx_LDOFF

Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD

Instrukce POSx_LDPOS





Instrukce POSx_LDPOS (Nahrát polohu) mění hodnotu aktuální polohy v polohovacím modulu na novou. Tuto instrukci můžete také použít pro stanovení nové nulové polohy pro kterýkoliv příkaz absolutního pohybu.

Zapnutí vstupu EN instrukci povoluje. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit ”Done” signalizovat, že je provádění instrukce ukončeno.

Zapnutí parametru START vyšle příkaz LDPOS polohovacímu modulu. V každém programovém cyklu, kdy je parametr START zapnutý a polohovací modul právě není v činnosti, vyšle instrukce polohovacímu modulu příkaz LDPOS. Aby bylo jisté, že je vyslán pouze jediný příkaz, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

Parametr New_Pos dává novou hodnotu, která nahrazuje hodnotu aktuální polohy a kterou polohovací modul hlásí a používá pro absolutní pohyby. V závislosti na jednotkách odměřování je tato hodnota buď počet pulzů (DINT), nebo inženýrských jednotek (REAL).

Parametr ”Done” (Dokončeno) se zapne ve chvíli, kdy modul dokončí tuto instrukci.

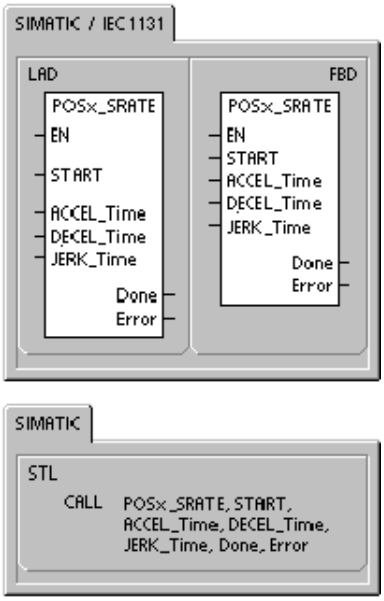
Parametr ”Error” (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

Parametr C_Pos obsahuje aktuální polohu modulu. V závislosti na jednotkách měření je tato hodnota buď počet pulzů (DINT), nebo počet inženýrských jednotek (REAL).

Tabulka 9-15 Parametry instrukce POSx_LDPOS

Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
New_Pos, C_Pos	DINT, REAL	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD
Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD

Instrukce POSx_SRATE



Instrukce POSx_SRATE (Nastavit rychlost) dává polohovacímu modulu příkaz změnit dobu zrychlení, zpomalení a zaoblení.

Zapnutí vstupu EN instrukci povoluje. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit "Done" signalizovat, že je provádění instrukce ukončeno.

Zapnutí parametru START zkopíruje nové časové hodnoty do tabulky konfigurace profilu a vyšle příkaz SRATE polohovacímu modulu. V každém programovém cyklu, kdy je parametr START zapnutý a polohovací modul právě není v činnosti, vyšle instrukce polohovacímu modulu příkaz SRATE. Aby bylo jisté, že je vyslán pouze jediný příkaz, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

Parametry ACCEL_Time, DECEL_Time a JERK_Time určují novou dobu zrychlení, dobu zpomalení a dobu zaoblení v milisekundách (ms).

Parametr "Done" (Dokončeno) se zapne ve chvíli, kdy modul dokončí tuto instrukci.

Parametr "Error" (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

Tabulka 9-16 Parametry instrukce POSx_SRATE

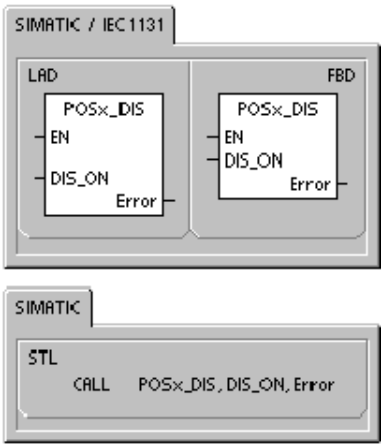
Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L
ACCEL_Time, DECEL_Time, JERK_Time	DINT	ID, QD, VD, MD, SMD, SD, LD, AC, *VD, *AC, *LD, konstanta
Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD

Instrukce POSx_DIS

Instrukce POSx_DIS zapíná nebo vypíná výstup DIS polohovacího modulu. To umožňuje použít výstup DIS pro zablokování nebo povolení řídicí jednotky motoru. Pokud použijete výstup DIS na polohovacím modulu, je možné volat tuto instrukci v každém programovém cyklu nebo pouze tehdy, je–li třeba změnit hodnotu výstupu DIS.

Když se zapne bit EN, aby byla instrukce povolena, řídí parametr DIS_ON výstup DIS polohovacího modulu. Více informací o výstupu DIS najdete v tabulce 9-8, nebo nahlédněte do specifikací polohovacího modulu v příloze A.

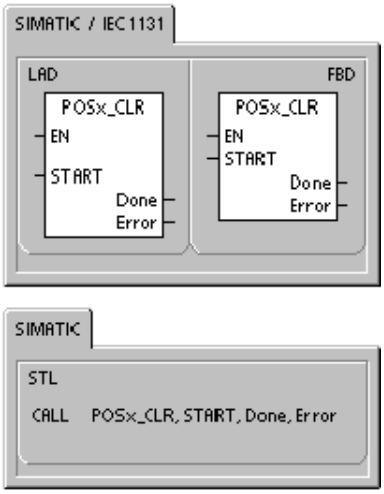
Parametr ”Error” (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.



Tabulka 9-17 Parametry instrukce POSx_DIS

Vstupy/Výstupy	Typ dat	Operandy
DIS_ON	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD, konstanta
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD

Instrukce POSx_CLR



Instrukce POSx_CLR (Pulz na výstupu CLR) dá polohovacímu modulu příkaz, aby na výstupu CLR generoval 50ms pulz.

Zapnutí vstupu EN instrukci povoluje. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit ”Done” signalizovat, že je provádění instrukce ukončeno.

Zapnutí parametru START vyšle příkaz CLR polohovacímu modulu. V každém programovém cyklu, kdy je parametr START zapnutý a polohovací modul právě není v činnosti, vyšle instrukce polohovacímu modulu příkaz CLR. Aby bylo jisté, že je vyslán pouze jediný příkaz, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

Parametr ”Done” (Dokončeno) se zapne ve chvíli, kdy modul dokončí tuto instrukci.

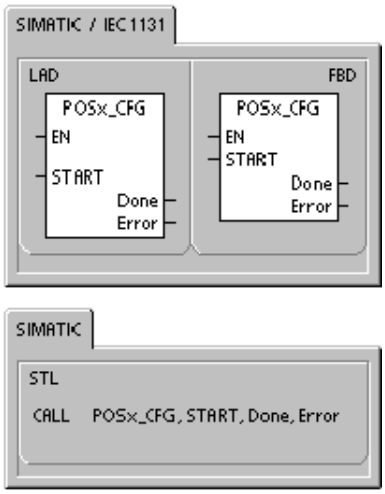
Parametr ”Error” (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

Tabulka 9-18 Parametry instrukce POSx_CLR

Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD

Instrukce POSx_CFG

Instrukce POSx_CFG (Znovu nahrát konfiguraci) dá polohovacímu modulu příkaz, aby načetl konfigurační blok z místa specifikovaného ukazatelem tabulky konfigurace profilu. Pak polohovací modul porovná novou konfiguraci se stávající a provede všechny potřebné změny nastavení nebo nové výpočty.



Zapnutí vstupu EN instrukci povoluje. Zajistěte, aby bit EN zůstal zapnutý, dokud nebude bit "Done" signalizovat, že je provádění instrukce ukončeno.

Zapnutí parametru START vyšle příkaz CFG polohovacímu modulu. V každém programovém cyklu, kdy je parametr START zapnutý a polohovací modul právě není v činnosti, vyšle instrukce polohovacímu modulu příkaz CFG. Aby bylo jisté, že je vyslán pouze jediný příkaz, použijte pro vyslání pulzu k zapnutí parametru START detekci hrany.

Parametr "Done" (Dokončeno) se zapne ve chvíli, kdy modul dokončí tuto instrukci.

Parametr "Error" (Chyba) obsahuje výsledek této instrukce. Definice chybových kódů jsou popsány v tabulce 9-20.

Tabulka 9-19 Parametry instrukce POSx_CFG

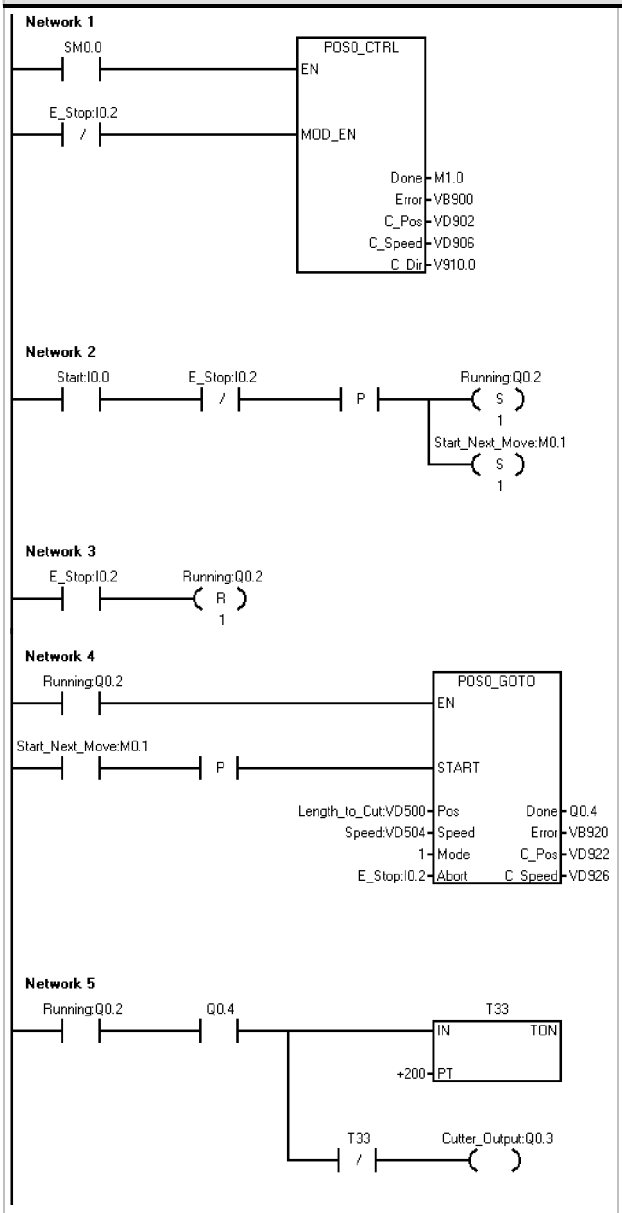
Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, V, M, SM, S, T, C, L, signálový tok
Done	BOOL	I, Q, V, M, SM, S, T, C, L
Error	BYTE	IB, QB, VB, MB, SMB, SB, LB, AC, *VD, *AC, *LD

Vzorové programy pro polohovací modul

První vzorový program ukazuje jednoduchý relativní pohyb, který používá instrukce POSx_CTRL a POSx_GOTO pro provedení operace řezání materiálu o konstantní délce. Tento program nepotřebuje režim hledání RP nebo profil pohybu a délku lze měřit buď na pulzy, nebo na inženýrské jednotky. Vložte délku (VD500) a cílovou rychlost (VD504). Když se zapne I0.0 (Start), začne stroj pracovat. Když se zapne I0.1 (Stop), stroj dokončí operaci a zastaví se. Když se zapne I0.2 (E_Stop), zruší stroj jakýkoliv pohyb a okamžitě se zastaví.

Ve druhém vzorovém programu jsou použity instrukce POSx_CTRL, POSx_RUN, POSx_RSEEK a POSx_MAN. Musíte nakonfigurovat režim hledání RP a profil pohybu.

Vzorový program 1: Jednoduchý relativní pohyb (použití pro řezání na délku)




```
Network 1       //Řídicí instrukce (modul v místě 0).
LD      SM0.0
=       L60.0
LDN     I0.2
=       L63.7
LD      L60.0
CALL    POS0_CTRL, L63.7, M1.0, VB900,
        VD902, VD906, V910.0

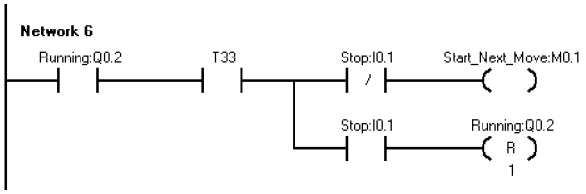
Network 2       //Spuštění uvede stroj do
                //automatického režimu.
LD      I0.0
AN      I0.2
EU
S       Q0.2, 1
S       M0.1, 1

Network 3       //E_Stop: provede okamžité
                //zastavení a vypne automatický
                //režim.
LD      I0.2
R       Q0.2, 1

Network 4       //Pohyb do určitého bodu:
                //Vložte délku řezu.
                //Vložte cílovou rychlost do "Speed".
                //Nastavte režim na 1 (Relativní
                //režim).
LD      Q0.2
=       L60.0
LD      M0.1
EU
=       L63.7
LD      L60.0
CALL    POS0_GOTO, L63.7, VD500, VD504,
        1, I0.2, Q0.4, VB920, VD922, VD926

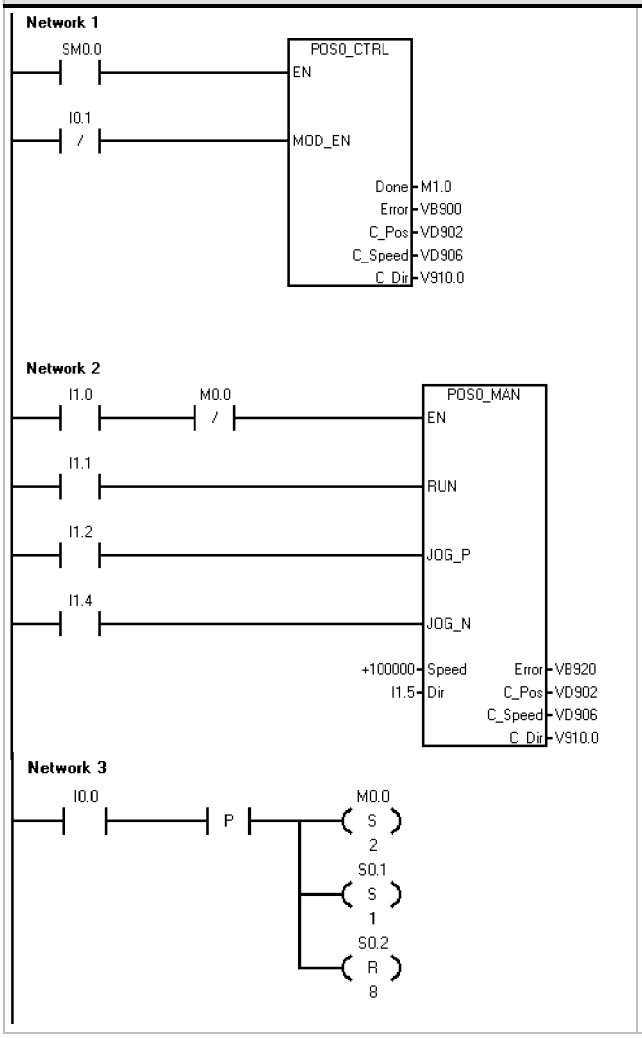
Network 5       //Dosáhne-li polohy, zapne řezací
                //stroj na 2 sekundy pro dokončení
                //řezu.
LD      Q0.2
A       Q0.4
TON     T33, +200
AN      T33
=       Q0.3
```

Vzorový program 1: Jednoduchý relativní pohyb (použití pro řezání na délku), pokračování




```
Network 6      //Když je řez dokončen, provede
               //restart, pokud není aktivní Stop.
LD     Q0.2
A      T33
LPS
AN     I0.1
=      M0.1
LPP
A      I0.1
R      Q0.2, 1
```

Vzorový program 2: Program s instrukcemi POSx_CTRL, POSx_RUN, POSx_SEEK a POSx_MAN

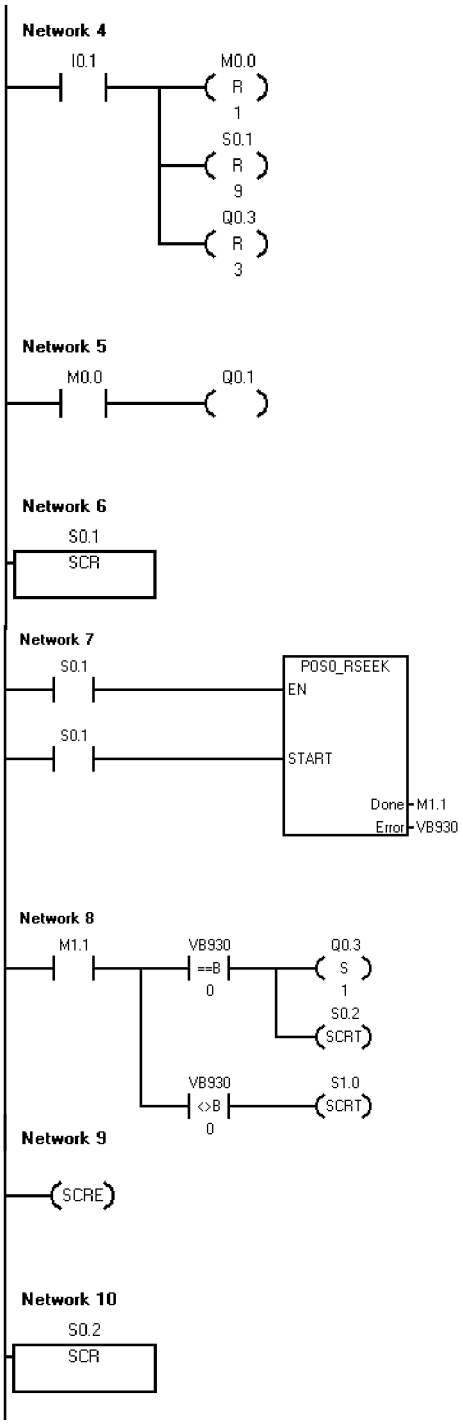



```
Network 1      //Povolí polohovací modul.
LD     SM0.0
=      L60.0
LDN    I0.1
=      L63.7
LD     L60.0
CALL   POS0_CTRL, L63.7, M1.0, VB900,
       VD902, VD906, V910.0

Network 2      //Manuální režim, pokud není
               //v automatickém režimu.
LD     I1.0
AN     M0.0
=      L60.0
LD     I1.1
=      L63.7
LD     I1.2
=      L63.6
LD     I1.4
=      L63.5
LD     L60.0
CALL   POS0_MAN, L63.7, L63.6,
       L63.5, +100000, 1.5, VB920,
       VD902, VD906, V910.0

Network 3      //Povolí automatický režim.
LD     I0.0
EU
S      M0.0, 2
S      S0.1, 1
R      S0.2, 8
```

Vzorový program 2: Program s instrukcemi POSx_CTRL, POSx_RUN, POSx_SEEK a POSx_MAN, pokračování




```
Network 4       //Nouzové zastavení
                //Zablokuje modul a automatický
                //režim.
LD      I0.1
R       M0.0, 1
R       S0.1, 9
R       Q0.3, 3

Network 5       //Když je v automatickém režimu:
                //Rozsvítí signálku pro signalizaci
                //chodu.
LD      M0.0
=       Q0.1

Network 6
LSCR    S0.1

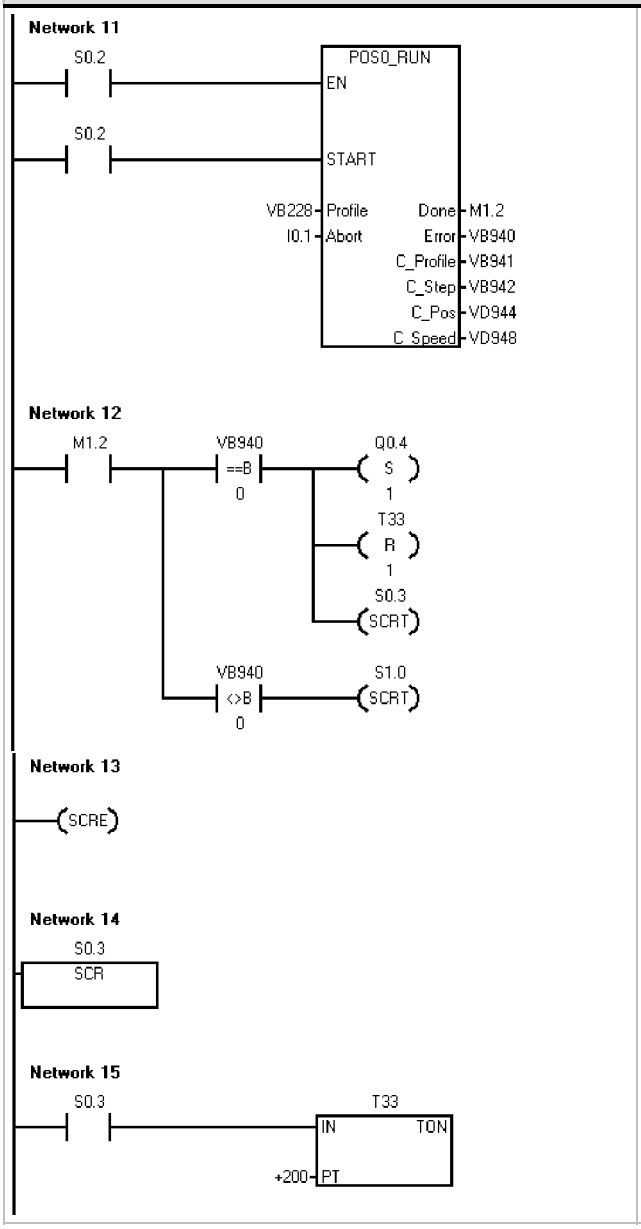
Network 7       //Vyhledá referenční bod (RP).
LD      S0.1
=       L60.0
LD      S0.1
=       L63.7
LD      L60.0
CALL    POS0_RSEEK, L63.7, M1.1, VB930

Network 8       //Když je dosaženo referenčního
                //bodu (RP):
                //Upne materiál a přejde na další
                //krok.
LD      M1.1
LPS
AB=     VB930, 0
S       Q0.3, 1
SCRT    S0.2
LPP
AB<>    VB930, 0
SCRT    S1.0

Network 9
SCRE

Network 10
LSCR    S0.2
```

Vzorový program 2: Program s instrukcemi POSx_CTRL, POSx_RUN, POSx_SEEK a POSx_MAN, pokračování



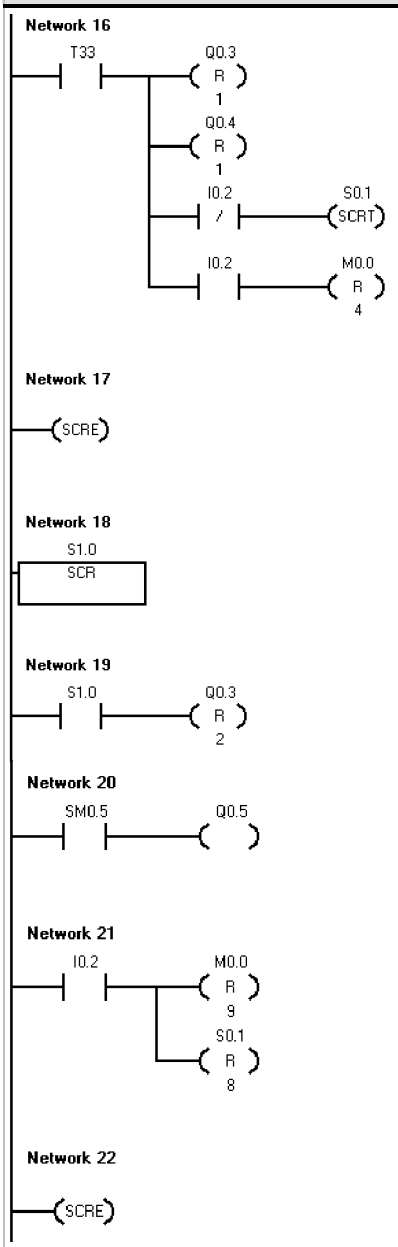

```
Network 11      //Použije profil 1 pro posun do
                //polohy.
LD      S0.2
=       L60.0
LD      S0.2
=       L63.7
LD      L60.0
CALL    POS0_RUN, L63.7, VB228, I0.1,
        M1.2, VB940, VB941, VB942,
        VD944, VD948
Network 12      //Když dosáhne polohy, zapne
                //řezací stroj a přejde na další krok.
LD      M1.2
LPS
AB=     VB940, 0
S       Q0.4, 1
R       T33, 1
SCRT    S0.3
LPP
AB<>    VB940, 0
SCRT    S1.0

Network 13
SCRE

Network 14      //Čeká na dokončení řezu.
LSCR    S0.3

Network 15
LD      S0.3
TON     T33, +200
```

Vzorový program 2: Program s instrukcemi POSx_CTRL, POSx_RUN, POSx_SEEK a POSx_MAN, pokračování




```
Network 16      //Pokud není zapnutý STOP,
                //provede po dokončení řezu
                //restart.
LD      T33
LPS
R       Q0.3, 1
R       Q0.4, 1
AN      I0.2
SCRT    S0.1
LPP
A       I0.2
R       M0.0, 4
Network 17
SCRE

Network 18
LSCR    S1.0

Network 19      //Resetuje výstupy.
LD      S1.0
R       Q0.3, 2

Network 20      //Bliká signálkou chyby.
LD      SM0.5
=       Q0.5

Network 21      //Ukončí chybový podprogram,
                //je–li zapnutý STOP.
LD      I0.2
R       M0.0, 9
R       S0.1, 8

Network 22
SCRE
```

Monitorování polohovacího modulu ”Control panelem EM 253”

Jako pomoc při vypracování vašeho řešení řízení polohy nabízí STEP 7 – Micro/WIN řídicí panel EM 253. Díky záložkám ”Ovládání”, ”Konfigurace” a ”Diagnostika” je pro vás snadné monitorovat a řídit práci polohovacího modulu během spuštění a zkušebních fází vývojového procesu.

”Control panel EM 253” použijte pro ověření toho, že je polohovací modul správně zapojen, pro úpravu konfiguračních dat a pro odzkoušení všech pohybových profilu.

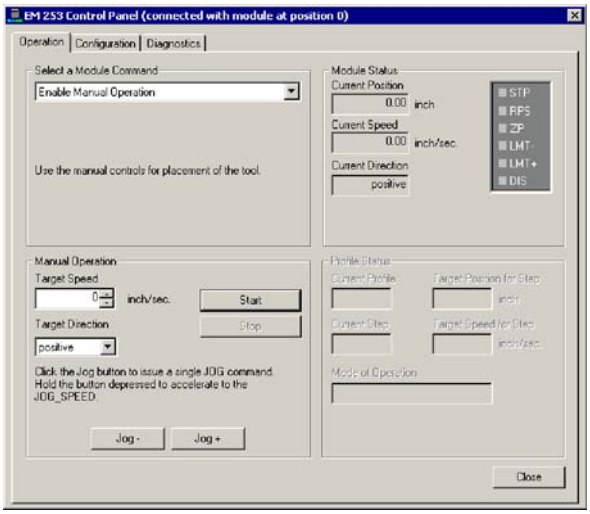
Zobrazení stavu a řízení polohovacího modulu

Záložka ”Ovládání” na ”Control panelu pro EM 253” umožňuje přímo ovládat polohovací modul. ”Control panel pro EM 253” ukazuje aktuální rychlost, polohu a směr polohovacího modulu. Můžete také vidět stav vstupních a výstupních LED diod (s výjimkou pulzních LED).

”Control panel pro EM 253” umožňuje interakci s polohovacím modulem tím, že měníte jeho rychlost a směr, zastavujete a spouštíte pohyb nebo manuálně posouváte nástroj (pokud je pohyb zastaven).

Můžete také generovat následující pohybové příkazy:

- ”Manuální ovládání”. Tento příkaz umožňuje použít pro polohování nástroje ruční ovládací prvky.
- ”Spustit profil pohybu”. Tento příkaz umožňuje vybrat profil, který bude prováděn. ”Control panel pro EM 253” zobrazuje stav profilu, který polohovací modul právě provádí.



Obr. 9-17 Záložka ”Ovládání” ”Control panelu pro EM 253”

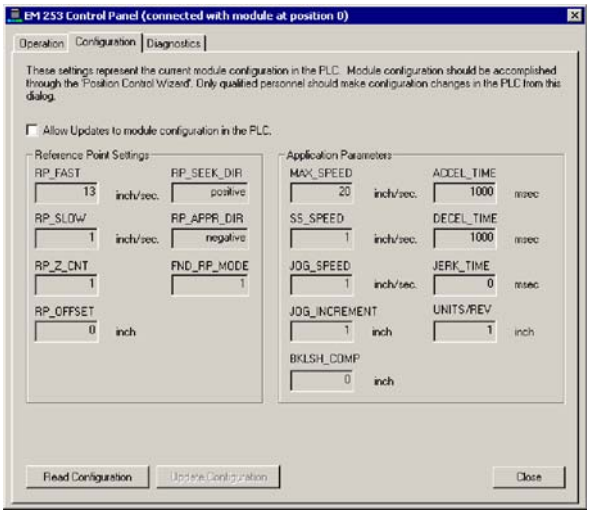
- ”Vyhledat referenční bod”. Tento příkaz najde referenční bod pomocí nakonfigurovaného způsobu hledání.
- ”Načíst offset referenčního bodu”. Poté, co jste použili ruční ovládací prvky pro offset nástroje do nové nulové polohy, nahrajte offset referenčního bodu.
- ”Znovu nahrát aktuální polohu”. Tento příkaz aktualizuje hodnotu a stanoví novou nulovou polohu.
- ”Aktivovat výstup DIS” a ”Deaktivovat výstup DIS”. Tyto příkazy zapínají nebo vypínají výstup DIS polohovacího modulu.
- ”Pulz na výstupu CLR”. Tento příkaz generuje 50ms pulz na výstupu CLR polohovacího modulu.
- ”Naučit profil pohybu”. Tento příkaz umožňuje uložit cílovou polohu a rychlost profilu pohybu a kroku, když ručně polohujete nástroj. ”Control panel pro EM 253” zobrazuje stav profilu, který polohovací modul právě provádí.
- ”Nahrát konfiguraci modulu”. Tento příkaz nahraje novou konfiguraci tak, že přikáže polohovacímu modulu načíst konfigurační blok z paměti V automatu S7-200.

- ”Posun na absolutní polohu”. Tento příkaz umožňuje posun na zadanou polohu při cílové rychlosti. Dříve než použijete tento příkaz, musíte mít stanovenou nulovou polohu.
- ”Posun o relativní vzdálenost”. Tento příkaz umožňuje posun o zadanou vzdálenost z aktuální polohy při cílové rychlosti. Můžete vložit kladnou i zápornou vzdálenost.
- ”Resetovat příkazy modulu”. Tento příkaz vymaže příkazový byte polohovacího modulu a nastaví bit ”Done” (Dokončeno). Použijte tento příkaz, pokud se zdá, že polohovací modul nereaguje na příkazy.

Zobrazení a úprava konfigurace polohovacího modulu

Záložka ”Konfigurace” na ”Control panelu pro EM 253” umožňuje prohlížet a modifikovat nastavení konfigurace polohovacího modulu, která jsou uložená v datovém bloku automatu S7-200.

Po změně nastavení konfigurace jednoduše klikněte na tlačítko, čímž aktualizujte jak nastavení v projektu STEP 7 – Micro/Win, tak i v datovém bloku S7-200.



Obr. 9-18 Záložka ”Konfigurace” ”Control panelu pro EM 253”

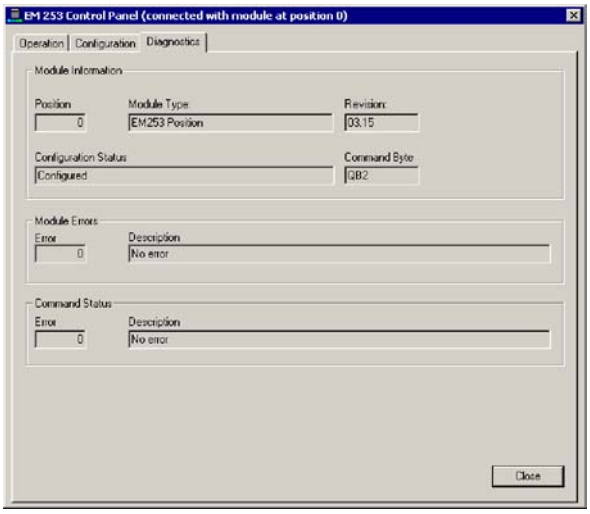
Zobrazení diagnostických informací polohovacího modulu

Záložka “Diagnostika” ”Control panelu pro EM 253” umožňuje prohlížet diagnostické informace, týkající se polohovacího modulu.

Je možné prohlížet konkrétní informace o polohovacím modulu, jako je umístění modulu, typ modulu a verze firmware i výstupní byte, používaný jako příkazový byte pro modul.

”Control panel pro EM 253” zobrazí jakýkoliv chybový stav, který vyplyne z přikázané operace. Chybové stavy pro jednotlivé instrukce najdete v tabulce 9-20.

Můžete také prohlížet jakýkoliv chybový stav hlášený polohovacím modulem. Chybové stavy modulu najdete v tabulce 9-21.



Obr. 9-19 Záložka ”Diagnostika” ”Control panelu pro EM 253”

Chybové kódy pro polohovací modul a polohovací instrukce

Tabulka 9-20 Chybové kódy instrukcí

Chybové kódy	Popis
0	Bez chyby
1	Zrušeno uživatelem
2	Chyba konfigurace Pro prohlížení chybových kódu použijte záložku ”Diagnostika” na ”Control panelu pro EM 253”
3	Neplatný příkaz
4	Zrušeno pro neexistenci platné konfigurace Pro prohlížení chybových kódu použijte záložku ”Diagnostika” na ”Control panelu pro EM 253”
5	Zrušeno vzhledem k nepřítomnosti napájení ze strany uživatele
6	Zrušeno pro neexistenci definovaného referenčního bodu
7	Zrušeno kvůli aktivaci vstupu STP
8	Zrušeno kvůli aktivaci vstupu LMT–
9	Zrušeno kvůli aktivaci vstupu LMT+
10	Zrušeno kvůli problému s prováděním pohybu
11	Pro zadaný profil není nakonfigurován profilový blok
12	Nepovolený pracovní režim
13	Pracovní režim není pro tento příkaz podporován
14	Nepovolený počet kroků v profilovém bloku
15	Nepovolená změna směru
16	Nepovolená vzdálenost
17	Ke spuštění RPS došlo před dosažením cílové rychlosti
18	Nedostatečná šířka aktivní oblasti RPS
19	Rychlost mimo rozsah
20	Nedostatečná vzdálenost pro provedení požadované změny rychlosti/otáček
21	Nepovolená poloha
22	Není známa nulová poloha
23 až 127	Rezervováno
128	Polohovací modul nemůže zpracovat tuto instrukci: polohovací modul zpracovává jinou instrukci, nebo pro tuto instrukci nebyl pulz ”Start”
129	Chyba polohovacího modulu: Nesprávné ID modulu nebo modul odhlášen. Ostatní chybové stavy viz SMB8 až SMB21 (ID vstupního/výstupního modulu a chybový registr).
130	Polohovací modul není povolen
131	Polohovací modul není dostupný kvůli chybě modulu nebo není povolen (Viz stav POSx_CTRL)
132	Adresa paměti Q, která byla nakonfigurována pomocí “Průvodce polohováním”, neodpovídá adrese paměti pro modul v tomto místě.

Tabulka 9-21 Chybové kódy modulu

Chybové kódy	Popis
0	Bez chyby
1	Není napájení ze strany uživatele
2	Není přítomen konfigurační blok
3	Chyba ukazatele konfiguračního bloku
4	Velikost konfiguračního bloku větší než dostupná paměť V
5	Nepovolený formát konfiguračního bloku
6	Zadáno příliš mnoho profilů
7	Nepovolená specifikace STP_RSP
8	Nepovolená specifikace LMT– _RPS
9	Nepovolená specifikace LMT+ _RPS
10	Nepovolená specifikace FILTER_TIME
11	Nepovolená specifikace MEAS_SYS
12	Nepovolená specifikace RP_CFG
13	Nepovolená hodnota PLS/REV
14	Nepovolená hodnota UNITS/REV
15	Nepovolená hodnota RP_ZP_CNT
16	Nepovolená hodnota JOG_INCREMENT
17	Nepovolená hodnota MAX_SPEED
18	Nepovolená hodnota SS_SPD
19	Nepovolená hodnota RP_FAST
20	Nepovolená hodnota RP_SLOW
21	Nepovolená hodnota JOG_SPEED
22	Nepovolená hodnota ACCEL_TIME
23	Nepovolená hodnota DECEL_TIME
24	Nepovolená hodnota JERK_TIME
25	Nepovolená hodnota BKLSH_COMP

Témata pro pokročilé

Vysvětlení tabulky konfigurace profilů

”Průvodce polohováním” byl vyvinut, aby usnadnil polohovací aplikace tím, že automaticky generuje konfigurační a profilové hodnoty na základě odpovědí, které o svém systému dáváte řízení polohy. Informace o tabulce konfigurace profilů jsou pro pokročilé uživatele, kteří chtějí vytvořit své vlastní podprogramy pro řízení polohy.

Tabulka konfigurace profilů je umístěna v oblasti paměti V automatu S7-200. Jak ukazuje tabulka 9-22, hodnoty jsou uloženy v následujících informacích:

- ❏ Konfigurační blok obsahuje informace použité pro nastavení modulu při přípravě na provádění pohybových příkazů.
- ❏ Interaktivní blok podporuje přímé nastavení pohybových parametrů uživatelským programem.
- ❏ Každý profilový blok popisuje předdefinovanou pohybovou operaci, kterou má provádět polohovací modul. Můžete nakonfigurovat až 25 profilových bloků.



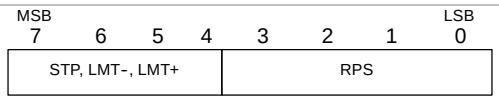
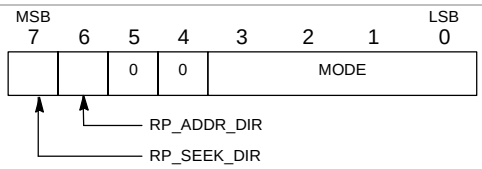
Tip

Chcete–li vytvořit více než 25 pohybových profilů, můžete vyměnit tabulky konfigurace profilů změnou hodnoty uložené v ukazateli tabulku konfigurace profilu.

Tabulka 9-22 Tabulka konfigurace profilu

Offset	Jméno	Popis funkce	Typ
Konfigurační blok			
0	MOD_ID	Identifikátor modulu	– –
5	CB_LEN	Délka konfiguračního bloku v bytech (1 byte)	– –
6	IB_LEN	Délka interaktivního bloku v bytech (1 byte)	– –
7	PF_LEN	Délka jednoho profilu v bytech (1 byte)	– –
8	STP_LEN	Délka jednoho kroku v bytech (1 byte)	– –
9	STEPS	Počet kroků na profil (1 byte)	– –
10	PROFILES	Počet profilů od 0 do 25 (1 byte)	– –
11	Rezervováno	Nastavit na 0x0000	– –
13	IN_OUT_CFG	Specifikuje použití vstupů a výstupů modulu (1 byte) MSB 7 6 5 4 3 2 1 0 LSB P/D | POL | 0 | 0 | STP | RPS | LMT- | LMT+ P/D Tento bit specifikuje použití P0 a P1. Kladná polarita (POL=0): 0 – Pulzy P0 pro kladný pohyb Pulzy P1 pro záporný pohyb 1 – Pulzy P0 pro pohyb P1 řídí směr pohybu (0 – kladné, 1 – záporné) Záporná polarita (POL=1): 0 – Pulzy P0 pro kladný pohyb Pulzy P1 pro záporný pohyb 1 – Pulzy P0 pro pohyb P1 řídí směr pohybu (0 – kladné, 1 – záporné) POL Tento bit volí polaritu pro P0 a P1. (0 – kladná polarita, 1 – záporná polarita) STP Tento bit řídí aktivní úroveň vstupu Stop. RPS Tento bit řídí aktivní úroveň vstupu RPS. LMT– Tento bit řídí aktivní úroveň vstupu meze záporného posunu. LMT+ Tento bit řídí aktivní úroveň vstupu meze kladného posunu. 0 – Aktivní vysoká úroveň 1 – Aktivní nízká úroveň	– –

Tabulka 9-22 Tabulka konfigurace profilu, pokračování

Offset	Jméno	Popis funkce	Typ
14	STP_RSP	Specifikuje odezvu pohonu na vstup STP (1 byte) 0 Nereagovat. Ignorovat stav na vstupu. 1 Zastavit se zpomalením a indikovat aktivní stav vstupu STP. 2 Ukončit pulzy a indikovat vstup STP 3 až 255 Rezervováno (zadání této hodnoty znamená chybu)	– –
15	LMT– _RSP	Specifikuje odezvu pohonu na vstup záporné meze (1 byte) 0 Nereagovat. Ignorovat stav na vstupu. 1 Zastavit se zpomalením a indikovat, že bylo dosaženo meze. 2 Ukončit impulzy a indikovat, že bylo dosaženo meze. 3 až 255 Rezervováno (zadání této hodnoty znamená chybu)	– –
16	LMT+ _RSP	Specifikuje odezvu pohonu na vstup kladné meze (1 byte) 0 Nereagovat. Ignorovat stav na vstupu. 1 Zastavit se zpomalením a indikovat, že bylo dosaženo meze. 2 Ukončit pulzy a indikovat, že bylo dosaženo meze. 3 až 255 Rezervováno (zadání této hodnoty znamená chybu)	– –
17	FILTER_TIME	Specifikuje čas filtrů pro vstupy STP, LMT–, LMT+ a RPS (1 byte)  ’0000’ 200µs ’0101’ 3200 µs ’0001’ 400µs ’0110’ 6400µs ’0010’ 800µs ’0111’ 12800µs ’0011’ 1600µs ’1000’ Bez filtru ’0100’ 1600µs ’1001 ’ až ’1111’ Rezervováno (zadání této hodnoty znamená chybu)	– –
18	MEAS_SYS	Specifikuje systém odměřování (1 byte) 0 Počet pulzů (rychlost je měřena v pulzech/sekundu, hodnoty polohy jsou měřeny v pulzech). Hodnoty jsou uchovávány jako DINT. 1 Inženýrské jednotky (rychlost je měřena v jednotkách/sekundu, hodnoty polohy jsou měřeny v jednotkách). Hodnoty jsou ukládány jako hodnota REAL s jednoduchou přesností. 2 až 255 Rezervováno (zadání této hodnoty znamená chybu)	– –
19	– –	Rezervováno (Nastaveno na 0)	– –
20	PLS/REV	Specifikuje počet pulzů na rychlost motoru (4 byty) Použitelné pouze tehdy, když je MEAS_SYS nastaven na 1.	DINT
24	UNITS/REV	Specifikuje počet inženýrských jednotek na rychlost motoru (4 byty) Použitelné pouze tehdy, když je MEAS_SYS nastaven na 1.	REAL
28	UNITS	Rezervováno pro STEP 7–Micro/WIN na uložení řetězce uživatelských jednotek (4 bytes)	– –
32	RP_CFG	Specifikuje konfiguraci vyhledávání referenčního bodu (1 byte)  RP_SEEK_DIR Tento bit specifikuje počáteční směr pro hledání referenčního bodu. (0 – kladný směr, 1 – záporný směr) RP_APPR_DIR Tento bit specifikuje směr přístupu pro ukončení hledání referenčního bodu. (0 – kladný směr, 1 – záporný směr) MODE Specifikuje způsob hledání referenčního bodu. ’0000’ Hledání referenčního bodu blokováno. ’0001’ Referenční bod je tam, kde vstup RPS přejde do aktivního stavu. ’0010’ Referenční bod leží ve středu aktivní oblasti vstupu RPS. ’0011’ Referenční bod leží mimo aktivní oblast vstupu RPS. ’0100’ Referenční bod leží v aktivní oblasti vstupu RPS. ’0101’ až ’1111’ Rezervováno (zadání této hodnoty znamená chybu)	– –
33	– –	Rezervováno (Nastaveno na 0)	– –
34	RP_Z_CNT	Počet pulzů vstupu ZP použitý pro definování referenčního bodu (4 byty)	DINT

Tabulka 9-22 Tabulka konfigurace profilu, pokračování

Offset	Jméno	Popis funkce	Typ
38	RP_FAST	Vysoká rychlost pro operaci hledání RP (RP Seek): MAX_SPD nebo méně, (4 byty)	DINT REAL
42	RP_SLOW	Nízká rychlost pro operaci hledání RP (RP Seek): maximální rychlost, ze které může motor okamžitě zastavit, nebo menší. (4 bytes)	DINT REAL
46	SS_SPEED	Rychlost spuštění/zastavení (4 byty) Rychlost spouštění je maximální rychlost, na kterou může motor okamžitě přejít ze zastavení, a maximální rychlost, ze které může motor okamžitě zastavit. Provoz je pod touto rychlostí povolený, neplatí ale doba zrychlení a zpomalení.	DINT REAL
50	MAX_SPEED	Maximální provozní rychlost motoru. (4 byty)	DINT REAL
54	JOG_SPEED	Rychlost manuálního posuvu. MAX_SPEED nebo méně. (4 byty)	
58	JOG_INCREMENT	Hodnota přírůstku manuálního posunu je vzdálenost (nebo počet pulzů), o kterou je proveden posun jako reakce na jeden pulz manuálního posunu. (4 byty)	DINT REAL
62	ACCEL_TIME	Čas potřebný pro zrychlení z minimální na maximální rychlost v milisekundách. (4 byty)	DINT
66	DECEL_TIME	Čas potřebný pro zpomalení z maximálních na minimální rychlost v milisekundách. (4 byty)	DINT
70	BKLSH_COMP	Kompenzace vůle: vzdálenost použitá na kompenzaci mrtvého chodu systému při změně směru. (4 byty)	DINT REAL
74	JERK_TIME	Doba, po kterou je kompenzace rázu aplikována na počáteční a konečnou část křivky zrychlení/zpomalení (křivka S). Zadání hodnoty 0 zakáže kompenzaci rázu. Doba rázu je uvedena v milisekundách. (4 byty)	DINT
Interaktivní blok			
78	MOVE_CMD	Vybírá pracovní režim (1 byte) 0 Absolutní poloha 1 Relativní poloha 2 Kontinuální kladný pohyb s jednou rychlostí 3 Kontinuální záporný pohyb s jednou rychlostí 4 Manuální řízení rychlosti, kladný pohyb 5 Manuální řízení rychlosti, záporný pohyb 6 Kontinuální kladný pohyb s jednou rychlostí a se spouštěným zastavením (vstup RPS signalizuje zastavení) 7 Kontinuální kladný pohyb s jednou rychlostí a se spouštěným zastavením (vstup RPS signalizuje zastavení) 8 až 255 – Rezervováno (zadání této hodnoty znamená chybu)	– –
79	– –	Rezervováno. Nastaveno na 0.	– –
80	TARGET_POS	Cílová poloha, která se má dosáhnout v tomto posunu. (4 byty)	DINT REAL
84	TARGET_SPEED	Cílová rychlost pro tento posun. (4 byty)	DINT REAL
88	RP_OFFSET	Absolutní poloha referenčního bodu. (4 byty)	DINT REAL
Profilový blok 0			
92 (+0)	STEPS	Počet kroku v této pohybové sekvenci (1 byte)	– –
93 (+1)	MODE	Volí pracovní režim pro tento profilový blok (1 byte) 0 Absolutní poloha 1 Relativní poloha 2 Kontinuální kladný pohyb s jednou rychlostí 3 Kontinuální záporný pohyb s jednou rychlostí 4 Rezervováno (zadání této hodnoty znamená chybu) 5 Rezervováno (zadání této hodnoty znamená chybu) 6 Kontinuální kladný pohyb s jednou rychlostí a se spouštěným zastavením (RPS volí rychlost) 7 Kontinuální záporný pohyb s jednou rychlostí se spouštěným zastavením (vstup RPS signalizuje stop) 8 Kontinuální kladný pohyb se dvěma rychlostmi (RPS volí rychlost) 9 Kontinuální záporný pohyb se dvěma rychlostmi (RPS volí rychlost) 10 až 255 – Rezervováno (zadání této hodnoty znamená chybu)	– –

Tabulka 9-22 Tabulka konfigurace profilu, pokračování

Offset	Jméno		Popis funkce	Typ
94 (+2)	0	POS	Poloha, které se má dosáhnout v kroku 0. (4 byty)	DINT REAL
98 (+6)		SPEED	Cílová rychlost pro krok 0. (4 byty)	DINT REAL
102 (+10)	1	POS	Poloha, které se má dosáhnout v kroku 1. (4 byty)	DINT REAL
106 (+14)		SPEED	Cílová rychlost pro krok 1. (4 byty)	DINT REAL
110 (+18)	2	POS	Poloha, které se má dosáhnout v kroku 2. (4 byty)	DINT REAL
114 (+22)		SPEED	Cílová rychlost pro krok 2. (4 byty)	DINT REAL
118 (+26)	3	POS	Poloha, které se má dosáhnout v kroku 3. (4 byty)	DINT REAL
122 (+30)		SPEED	Cílová rychlost pro krok 3. (4 byty)	DINT REAL
Profilový blok 1				
126 (+34)	STEPS		Počet kroku v této pohybové sekvenci. (1 byte)	– –
127 (+35)	MODE		Volí pracovní režim pro tento profilový blok. (1 byte)	– –
128 (+36)	0	POS	Poloha, které se má dosáhnout v kroku 0. (4 byty)	DINT REAL
132 (+40)		SPEED	Cílová rychlost pro krok 0. (4 byty)	DINT REAL
...	...	...	...	...

Byty speciální paměti

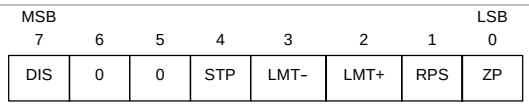
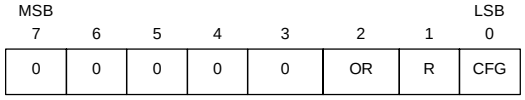
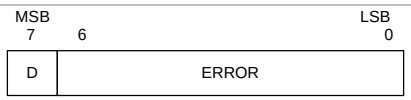
Automat S7-200 přiděluje 50 bytů speciální paměti (SM) každému inteligentnímu modulu na základě fyzické polohy modulu v systému vstupů a výstupů. Viz tabulku 9-23. Když modul detekuje chybu nebo změnu stavu dat, aktualizuje tato místa v paměti SM. Nejprve modul aktualizuje SMB200 až SMB249, které jsou nutné pro hlášení chybového stavu, za druhé modul aktualizuje SMB250 až SMB299 apod.

Tabulka 9-23 Byty speciální paměti SMB200 až SMB549

Byty SM pro inteligentní modul na:						
místě 0	**místě 1**	**místě 2**	**místě 3**	**místě 4**	**místě 5**	**místě 6**
SMB200 až SMB249	SMB250 až SMB299	SMB300 až SMB349	SMB350 až SMB399	SMB400 až SMB449	SMB450 až SMB499	SMB500 až SMB549

Tabulka 9-24 ukazuje strukturu datové oblasti SM pro inteligentní modul. Tabulka předpokládá umístění inteligentního modulu na pozici 0 v systému rozšiřovacích modulů.

Tabulka 9-24 Definice oblasti speciální paměti pro polohovací modul EM 253

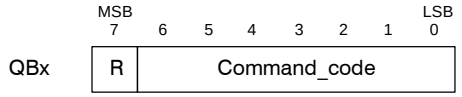
Adresa SM	**Popis**
SMB200 až SMB215	Název modulu (16 znaků ASCII). SMB200 začíná: "EM 253 Position"
SMB216 až SMB219	Číslo revize softwaru (4 ASCII znaky). SMB216 je první znak.
SMW220	Chybový kód modulu. Chybové kódy jsou popsány v tabulce 9-21.
SMB222	Stav vstupů a výstupů. Odráží stav vstupů a výstupů modulu.  DIS Blokování výstupů: 0 = Proud neprotéká; 1 = Proud protéká STP Vstup stop: 0 = Proud neprotéká; 1 = Proud protéká LMT– Vstup meze záporného posunu: 0 = Proud neprotéká; 1 = Proud protéká LMT+ Vstup meze kladného posunu: 0 = Proud neprotéká; 1 = Proud protéká RPS Vstup snímače referenčního bodu: 0 = Proud neprotéká; 1 = Proud protéká ZP Vstup nulového pulzu: 0 = Proud neprotéká; 1 = Proud protéká
SMB223	Okamžitý stav modulu. Odráží stav konfigurace modulu a stav směru otáčení.  OR Cílová rychlost mimo rozsah: 0 =V rozsahu; 1 = Mimo rozsah R Směr pohybu: 0 = Kladné otáčení; 1 = Záporné otáčení CFG Modul konfigurován: 0 = Nekonfigurován; 1 = Konfigurován
SMB224	CUR_PF je byte, který ukazuje na právě prováděný profil.
SMB225	CUR_STP je byte, který ukazuje na právě prováděný krok v profilu.
SMD226	CUR_POS je hodnota double word, která ukazuje na aktuální polohu modulu.
SMD230	CUR_SPD je hodnota double word, která ukazuje aktuální rychlost modulu.
SMB234	Výsledek instrukce. Viz tabulka 9-20 – Chybové kódyinstrukcí. Chybové stavy nad 127 jsou generovány instrukčními podprogramy vytvořenými průvodcem.  D Bit "Done" (Dokončeno) 0= Operace probíhá 1= Operace dokončena (nastaveno modulem během inicializace)
SMB235 až SMB244	Rezervováno
SMB245	Offset prvního bytu Q použitého jako příkazové rozhraní tohoto modulu. Offset dodává automat S7-200 automaticky pro pohodlí uživatele, modul ho nepotřebuje.
SMD246	Ukazatel na místo tabulky konfigurace profilu v paměti V. Ukazatel na jinou oblast než paměť V není platný. Polohovací modul toto místo monitoruje, dokud neobdrží jinou než nulovou hodnotu ukazatele.

Vysvětlení příkazového bytu polohovacího modulu

Polohovací modul má jeden byte digitálních výstupů, který se používá jako příkazový byte. Obrázek 9-20 a tabulka 9-20 ukazují definici příkazového bytu.

Zápis do příkazového bytu, kdy se bit R změní z 0 na 1, interpretuje modul jako nový příkaz.

Pokud modul detekuje přechod na nečinnost (bit R změní stav na 0) v době, kdy je příkaz aktivní, je prováděná operace zrušena. Pokud je prováděn pohyb, zastaví se zpomalením.



R 0 = Nečinný stav
1 = Provedení příkazu specifikovaného v Command_code (viz tabulku 9-25)

Obr. 9-20 Definice příkazového bytu

Po dokončení operace musí modul zaregistrovat přechod do nečinného stavu dříve, než přijme nový příkaz. Pokud je operace zrušena, musí modul dokončit jakékoliv zpomalení dříve, než přijme nový příkaz. Jakákoliv změna hodnoty Command_code v době, kdy je nějaký příkaz aktivní, je ignorována.

Odezva polohovacího modulu na změnu pracovního režimu automatu S7-200 nebo na chybu se řídí tím, jak S7-200 ovlivní digitální výstupy, podle stávající definice funkce programovatelného automatu S7-200:

- *Pokud S7-200 přejde z režimu STOP do režimu RUN:* Program v automatu S7-200 řídí práci polohovacího modulu.
- *Pokud S7-200 přejde z režimu RUN do režimu STOP:* Můžete vybrat stav, do kterého mají digitální výstupy přejít při přechodu do režimu STOP, nebo to, že si uchovají svůj poslední stav.
  - *Pokud se při přechodu do režimu STOP vypne bit R:* Polohovací modul zastaví jakýkoliv probíhající pohyb se zpomalením.
  - *Pokud se při přechodu do režimu STOP zapne bit R:* Polohovací modul dokončí jakýkoliv probíhající příkaz. Pokud žádný příkaz neprobíhá, provede polohovací modul příkaz specifikovaný bity Command_code.
  - *Pokud je bit R uchován ve svém posledním stavu:* Polohovací modul dokončí jakýkoliv probíhající příkaz.
- *Pokud automat S7-200 detekuje kritickou chybu a vypne všechny digitální výstupy:* Polohovací modul zastaví jakýkoliv probíhající pohyb se zpomalením.

Tabulka 9-25 Definice Command_code

Command_code	Příkaz	
000 0000 až 000 1111	0 až 24	Provést pohyb specifikovaný v profilových blocích 0 až 24
100 0000 až 111 0101	25 až 117	Rezervováno (zadání této hodnoty znamená chybu)
111 0110	118	Aktivovat výstup DIS
111 0111	119	Deaktivovat výstup DIS
111 1000	120	Pulz na výstupu CLR
111 1001	121	Znovu nahrát aktuální polohu
111 1010	122	Provést pohyb specifikovaný v interaktivním bloku
111 1011	123	Zachytit offset referenčního bodu
111 1100	124	Kladný manuální chod
111 1101	125	Záporný manuální chod
111 1110	126	Vyhledat polohu referenčního bodu
111 1111	127	Znovu nahrát konfiguraci

Při ztrátě komunikace s automatem S7-200 aktivuje polohovací modul obvod sledování funkce, který vypne výstupy. Pokud vyprší doba obvodu sledování funkce, polohovací modul zastaví jakýkoliv pohyb se zpomalením.

Při zjištění kritické chyby hardwaru nebo firware modulu uvede polohovací modul výstupy P0, P1, DIS a CLR do neaktivního stavu.

Tabulka 9-26 Pohybové příkazy

Příkaz	Popis
Příkazy 0 až 24: *Provede pohyb specifikovaný v profilových blocích 0 až 24*.	Při uskutečňování tohoto příkazu provádí polohovací modul pohybovou operaci specifikovanou v poli MODE profilového bloku, označeného částí příkazu Command_code. • V režimu 0 (absolutní poloha) definuje blok pohybového profilu jeden až čtyři kroky, z nichž každý obsahuje polohu (POS) i rychlost (SPEED), které popisují daný segment pohybu. Specifikace POS představuje absolutní místo, které se počítá podle místa označeného jako referenční bod. Směr pohybu je určen vztahem mezi aktuální polohou a polohou prvního kroku profilu. U pohybu o více krocích je obrácení směru posunu zakázáno a má za následek ohlášení chyby. • V režimu 1 (relativní poloha) definuje blok pohybového profilu jeden až čtyři kroky, z nichž každý obsahuje polohu (POS) i rychlost (SPEED), které popisují daný segment pohybu. Znaménko hodnoty polohy (POS) určuje směr pohybu. U pohybu o více krocích je obrácení směru posunu zakázáno a má za následek ohlášení chyby • V režimech 2 a 3 (režimy s jednou rychlostí a kontinuálním pohybem) je specifikace polohy (POS) ignorována a modul zrychluje na rychlost specifikované v poli SPEED prvního kroku. Režim 2 se používá pro kladný pohyb, režim 3 pro záporný. Pohyb se zastaví, když příkazový byte přejde do nečinného stavu. • V režimech 6 a 7 (režimy s jednou rychlostí a nepřetržitým provozem se spouštěným zastavením) modul zrychluje na rychlost specifikované v poli SPEED prvního kroku. Jestliže vstup RPS přejde do aktivního stavu, pak se pohyb zastaví po dokončení vzdálenosti specifikované v poli POS prvního kroku. (Vzdálenost specifikovaná v poli POS musí zahrnovat vzdálenost pro zpomalování.) Pokud má pole POS hodnotu nula v době, kdy vstup RPS přejde do aktivního stavu, polohovací modul zastaví se zpomalením. Režim 6 se používá pro kladný pohyb, režim 7 pro záporný. • V režimech 8 a 9 vybere binární hodnota vstupu RPS jednu ze dvou hodnot rychlostí specifikovaných prvními dvěma kroky profilového bloku. – Pokud není RPS aktivní: Krok 0 řídí rychlost pohonu. – Pokud je RPS aktivní: Krok 1 řídí rychlost pohonu. Režim 8 se používá pro kladný pohyb, režim 9 pro záporný. Hodnota SPEED řídí rychlost pohybu. V tomto režimu jsou hodnoty POS ignorovány.
Příkaz 118 *Aktivuje výstup DIS*.	Je–li prováděn tento příkaz, aktivuje polohovací modul výstup DIS.
Příkaz 119 *Deaktivuje výstup DIS*.	Je–li prováděn tento příkaz, deaktivuje polohovací modul výstup DIS.
Příkaz 120 *Pulz na výstupu CLR*.	Je–li prováděn tento příkaz, generuje polohovací modul 50milisekundový pulz na výstupu CLR.
Příkaz 121 *Znovu nahraje současnou polohu*.	Je–li prováděn tento příkaz, nastaví polohovací modul současnou polohu na hodnotu nalezenou v poli TARGET_POS interaktivního bloku.

Tabulka 9-26 Pohybové příkazy

Příkaz	Popis
Příkaz 122 *Provede pohyb specifikovaný v interaktivním bloku.*	Je–li prováděn tento příkaz, provede polohovací modul pohybovou operaci specifikovanou v poli MOVE_CMD interaktivního bloku. • V režimech 0 a 1 (režimy absolutního a relativního pohybu) je pohyb o jednom kroku prováděn na základě informací o cílové rychlosti a poloze uvedených v polích TARGET_SPEED a TARGET_POS interaktivního bloku. • V režimech 2 a 3 (režimy s jednou rychlostí a nepřetržitým provozem) je specifikace polohy ignorována a polohovací modul zrychluje na rychlost specifikovanou v poli TARGET_SPEED interaktivního bloku. Pohyb se zastaví, když příkazový byte přejde do nečinného stavu. • V režimech 4 a 5 (režimy manuálního řízení rychlosti) je specifikace polohy ignorována a uživatelský program nahraje hodnotu změn rychlosti do pole TARGET_SPEED interaktivního bloku. Polohovací modul toto místo trvale monitoruje a příslušně reaguje na změny hodnoty rychlosti.
Příkaz 123 *Zachycuje posunutí referenčního bodu*.	Je–li prováděn tento příkaz, stanovuje polohovací modul nulovou polohu, která je na jiném místě, než je poloha referenčního bodu. Před vydáním tohoto příkazu musíte mít stanovenou polohu referenčního bodu a stroj musí být manuálním posunem připraven v počáteční pracovní poloze. Po obdržení tohoto příkazu vypočítá polohovací modul offset mezi počáteční pracovní polohou (aktuální poloha) a polohou referenčního bodu a vypočítaný offset zapíše do pole RP_OFFSET interaktivního bloku. Pak je aktuální poloha nastavena na 0, čímž je počáteční pracovní poloha stanovena jako nulová poloha. V případě, že krokový motor ztratí přehled o své poloze (například při ztrátě napájení nebo když je poloha krokového motoru změněna ručně), je možné vydat příkaz ”Vyhledat polohu referenčního bodu” pro automatické opětné stanovení nulové polohy.
Příkaz 124 *Kladný manuální chod*	Tento příkaz umožňuje ručně vysílat pulzy pro posun krokového motoru v kladném směru. Pokud zůstane příkaz aktivní po dobu kratší než 0,5 sekundy, vysílá polohovací modul pulzy pro posun o vzdálenost specifikovanou v JOG_INCREMENT. Pokud příkaz zůstane aktivní po dobu 0,5 sekundy nebo déle, začne polohovací modul zrychlovat na specifikovanou JOG_SPEED. Je–li detekován přechod do nečinného stavu, zastaví polohovací modul se zpomalením.
Příkaz 125 *Záporný manuální chod*	Tento příkaz umožňuje ručně vysílat pulzy pro posun krokového motoru v záporném směru. Pokud zůstane příkaz aktivní po dobu kratší než 0,5 sekundy, vysílá polohovací modul pulzy pro posun o vzdálenost specifikovanou v JOG_INCREMENT. Pokud příkaz zůstane aktivní po dobu 0,5 sekundy nebo déle, začne polohovací modul zrychlovat na rychlost specifikované JOG_SPEED. Je–li detekován přechod do nečinného stavu, zastaví polohovací modul se zpomalením.
Příkaz 126 *Vyhledat polohu referenčního bodu*	Je–li prováděn tento příkaz, iniciuje polohovací modul operaci vyhledání referenčního bodu pomocí specifikovaného způsobu hledání. Když je referenční bod lokalizován a pohyb se zastavil, nahraje polohovací modul hodnotu načtenou z pole RP_OFFSET interaktivního bloku na aktuální polohu a generuje na výstupu CLR pulz o délce 50 milisekund.
Příkaz 127 *Opětovně nahrát konfiguraci*	Je–li prováděn tento příkaz, načte polohovací modul ukazatel tabulky konfigurace profilu z příslušného místa paměti SM, a potom načte konfigurační blok z místa specifikovaného ukazatelem tabulky konfigurace profilu. Polohovací modul porovná právě získaná konfigurační data se stávající konfigurací modulu a provede nutné změny nastavení nebo nové výpočty. Všechny profily uložené v rychlé vyrovnávací paměti jsou smazány.

Vysvětlení rychlé vyrovnávací paměti polohovacího modulu pro profily

Polohovací modul uchovává prováděcí data až pro 4 profily v rychlé vyrovnávací paměti. Když polohovací modul dostane příkaz k provedení nějakého profilu, kontroluje, zda je požadovaný profil uložen v rychlé vyrovnávací paměti. Pokud jsou prováděcí data profilu v této paměti přítomna, polohovací modul profil okamžitě provede. Pokud prováděcí data profilu v této paměti přítomna nejsou, načte polohovací modul informace profilového bloku z tabulky konfigurace profilů v automatu S7-200 a před jeho provedením vypočítá prováděcí data profilu.

Příkaz 122 (Provést pohyb specifikovaný v interaktivním bloku) nepoužívá pro uchovávání prováděcích dat rychlou vyrovnávací paměť, ale vždy načítá interaktivní blok z tabulky konfigurace profilů v automatu S7-200 a vypočítává prováděcí data pohybu.

Opětovná konfigurace polohovacího modulu vymaže všechna prováděcí data uložená v rychlé vyrovnávací paměti.

Tvorba vlastních instrukcí pro řízení polohy

Polohovací instrukce pro řízení práce polohovacího modulu vytváří ”Průvodce polohováním”; můžete ale také vytvořit své vlastní instrukce. Následující segment kódu STL je příklad toho, jak můžete vytvořit své vlastní řídicí instrukce pro polohovací modul.

V tomto příkladu je použit automat S7-200 CPU 224 s polohovacím modulem na místě 0. Polohovací modul je konfigurován při studeném startu. CMD_STAT je symbol pro SMB234, CMD je symbol pro QB2 a NEW_CMD je symbol pro profil.

Vzorový program: Řízení polohovacího modulu

```
Network 1       //Stav nového pohybového příkazu
LSCR    State_0

Network 2       //CMD_STAT je symbol pro SMB234.
                //CMD je symbol pro QB2.
                //NEW_CMD je symbol pro profil.
                //
                //1. Vymaže bit ”Done” (Dokončeno) polohovacího modulu.
                //2. Vymaže příkazový byte polohovacího modulu.
                //3. Vydá nový příkaz.
                //4. Čeká na provedení příkazu.
LD      SM0.0
MOVB    0, CMD_STAT
BIW     0, CMD
BIW     NEW_CMD, CMD
SCRT    State_1

Network 3
SCRE

Network 4       //Čeká na dokončení příkazu.
LSCR    State_1

Network 5       //Pokud je příkaz dokončen bez chyby, přejde do nečinného stavu.
LDB=    CMD_STAT, 16#80
SCRT    Idle_State

Network 6       //Pokud je příkaz dokončen s chybou, přejde do stavu správy chyb.
LDB>    CMD_STAT, 16#80
SCRT    Error_State

Network 7
SCRE
```

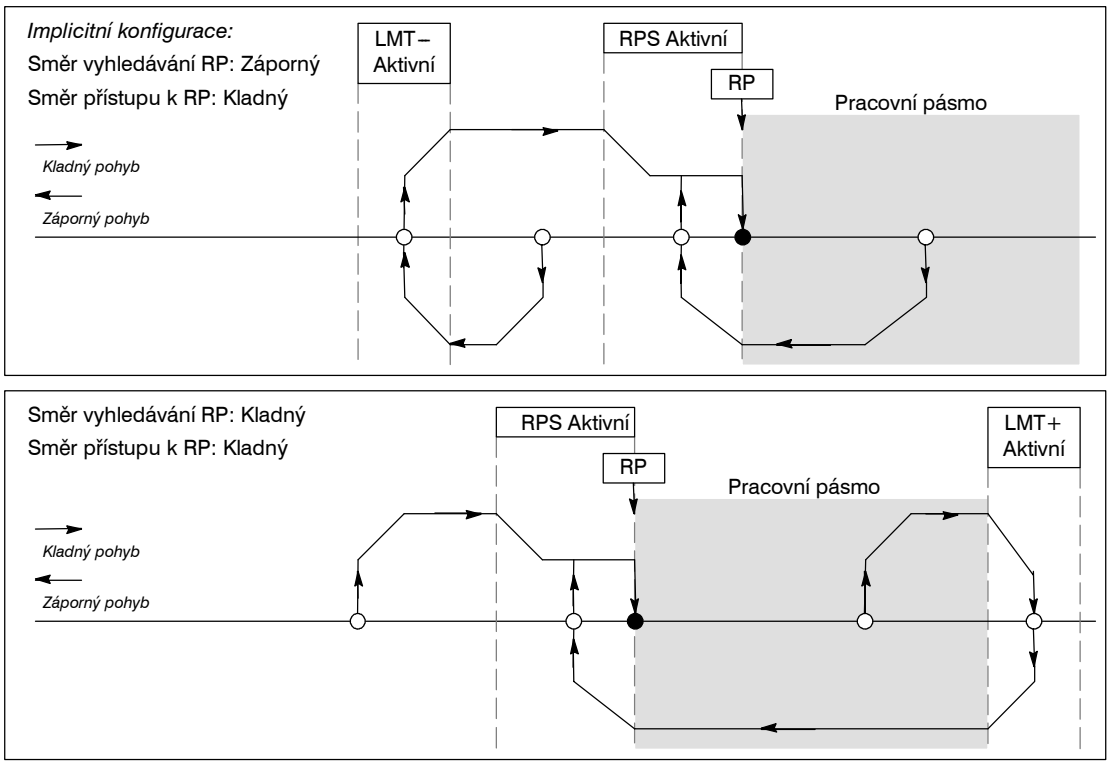
Vysvětlení režimů vyhledávání RP podporovaných polohovacím modulem

Následující obrázky ukazují schémata různých voleb pro každý režim vyhledávání RP.

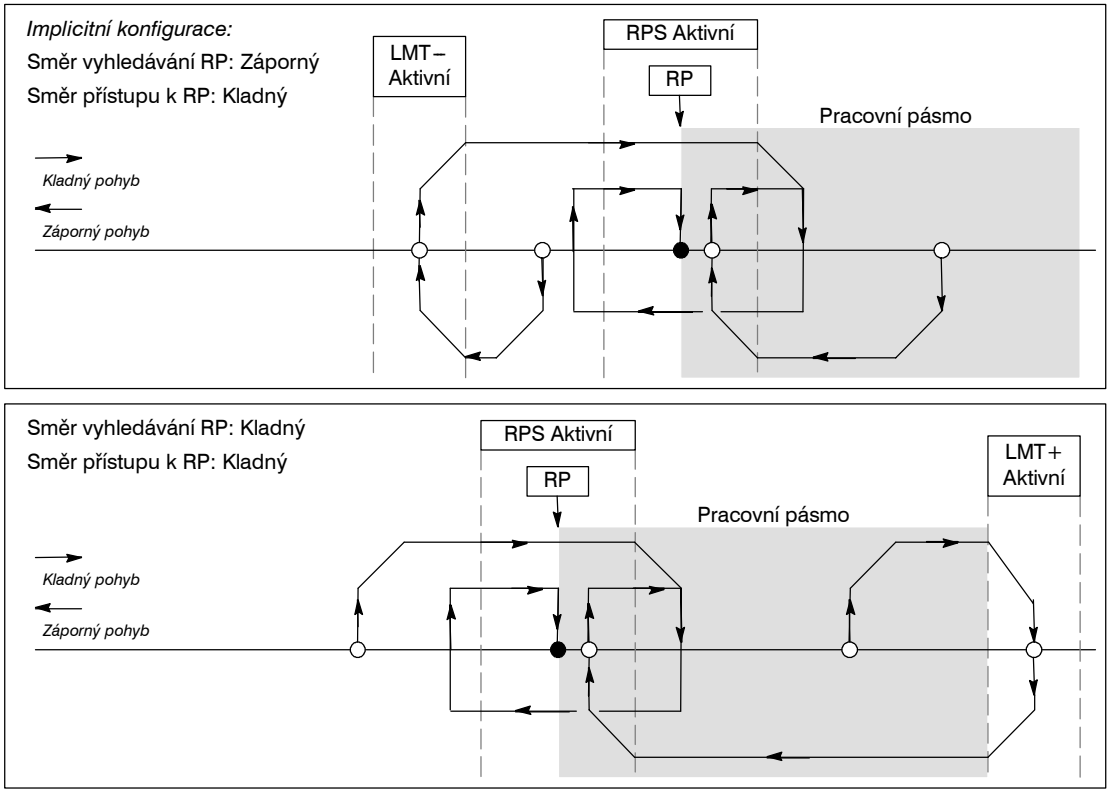
- ❑ Obrázek 9-21 ukazuje dvě z možností volby pro vyhledávání RP, režim 1. V tomto režimu je RP lokalizován tam, kde vstup RPS přejde do aktivního stavu při přiblížení ze strany pracovního pásma.
- ❑ Obrázek 9-22 ukazuje dvě z možností volby pro vyhledávání RP, režim 2. V tomto režimu je RP umístěn ve středu aktivní oblasti vstupu RPS.
- ❑ Obrázek 9-23 ukazuje dvě z možností volby pro vyhledávání RP, režim 3. V tomto režimu je RP umístěn o specifikovaný počet pulzů na vstupu (ZP) vně aktivní oblasti vstupu RPS.
- ❑ Obrázek 9-24 ukazuje dvě z možností volby pro vyhledávání RP, režim 4. V tomto režimu je RP umístěn o specifikovaný počet pulzů na vstupu (ZP) uvnitř aktivní oblasti vstupu RPS.

Pro každý režim existují čtyři kombinace směru vyhledávání RP a směru přístupu k RP. (Na obrázku jsou ukázány pouze dvě kombinace.) Tyto kombinace určují tvar operace vyhledávání RP. Pro každou kombinaci také existují čtyři různé počáteční body:

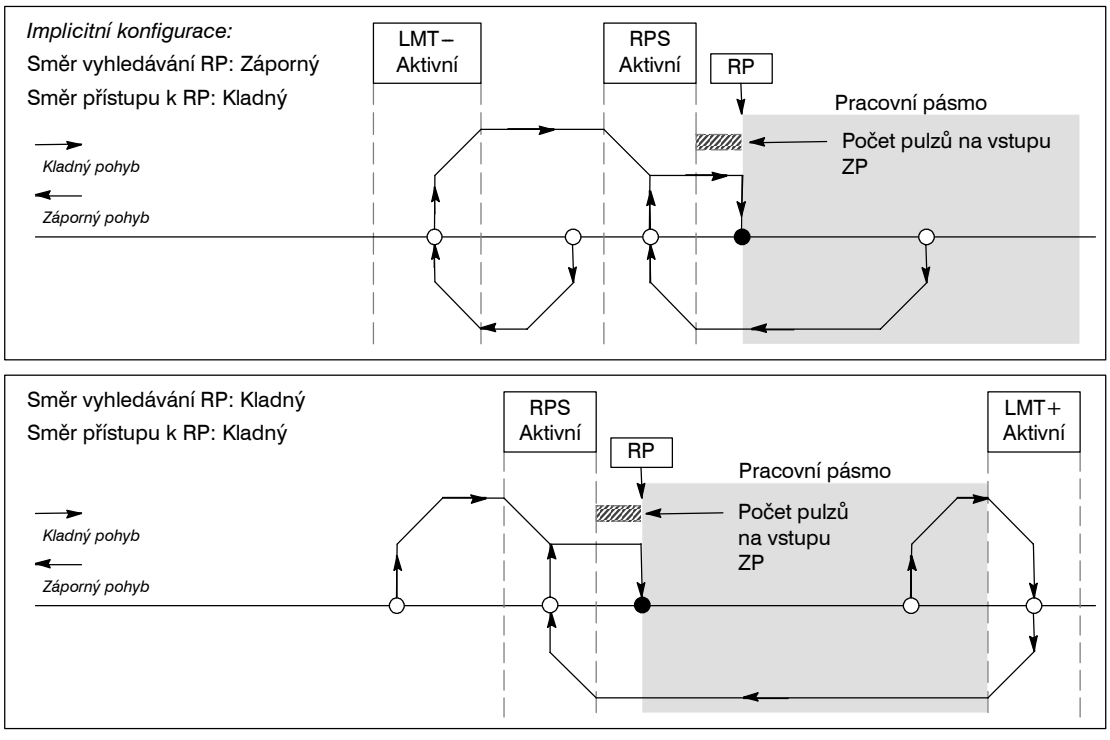
Pracovní pásma každého diagramu byla umístěna tak, aby pohyb od referenčního bodu k pracovnímu pásmu vyžadoval pohyb ve stejném směru, jako je směr přístupu k RP. Díky tomu, že je umístění pracovního pásma vybráno tímto způsobem, jsou eliminovány vůle mechanického převodového systému pro první pohyb směrem k pracovnímu pásmu po hledání referenčního bodu.



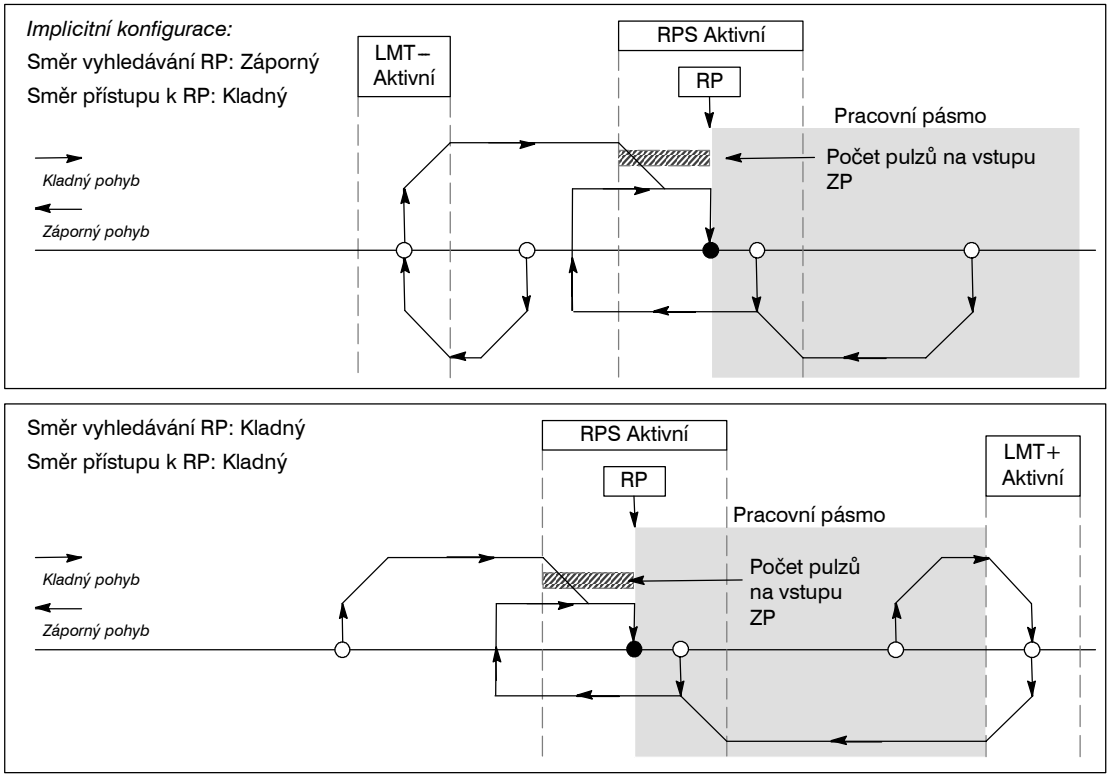
Obr. 9-21 Vyhledávání RP, režim 1



Obr. 9-22 Vyhledávání RP: Režim 2



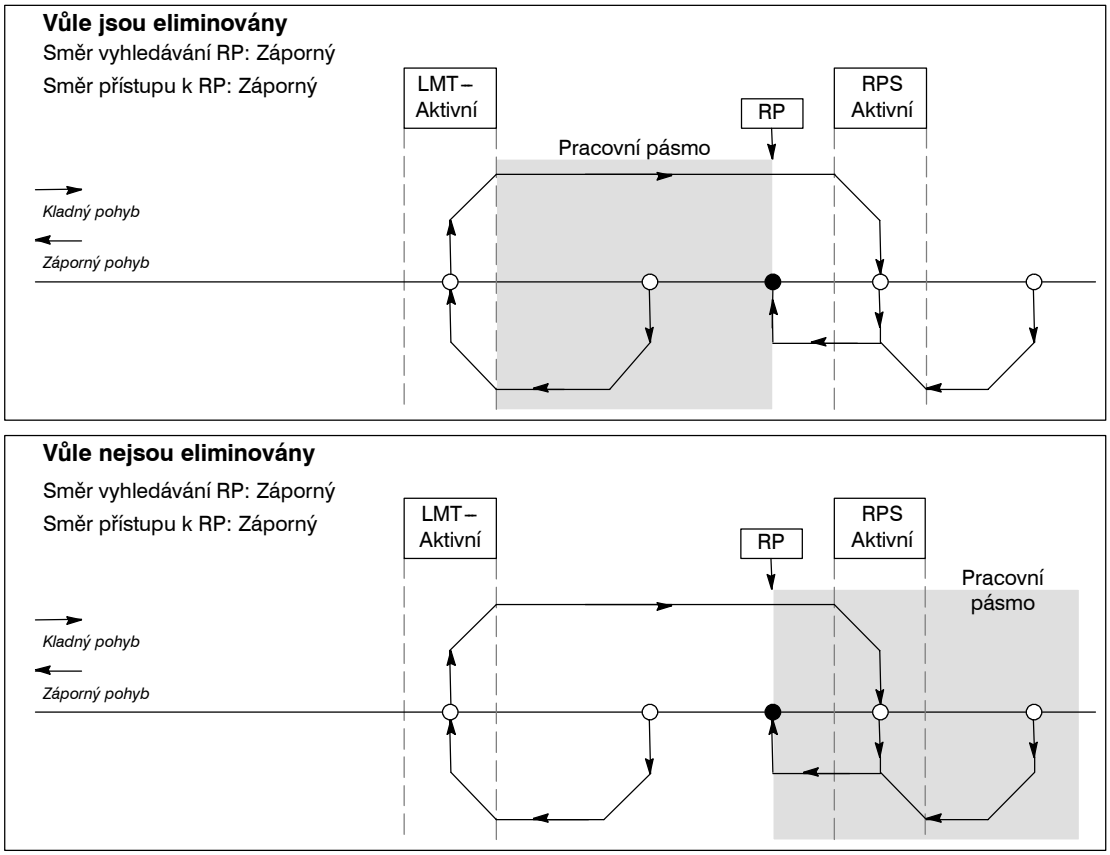
Obr. 9-23 Vyhledávání RP: Režim 3



Obr. 9-24 Vyhledávání RP: Režim 4

Výběr takového umístění pracovního pásma, které eliminuje vůli

Obrázek 9-25 ukazuje pracovní pásmo ve vztahu k referenčnímu bodu (RP), aktivní zóně RPS a koncovým spínačům (LMT+ a LMT–) pro směr přístupu, který vylučuje mechanickou vůli. V druhé části obrázku je pracovní pásmo umístěno tak, že nejsou eliminovány mechanické vůle. Na obrázku 9-25 je vyhledávání RP, režim 3. Obdobné umístění pracovního pásma je možné, i když se nedoporučuje, pro každou vyhledávací sekvenci všech ostatních režimů vyhledávání RP.



Obr. 9-25 Umístění pracovního pásma s eliminací vůle a bez eliminace

10 Tvorba programu pro modemový modul

Modemový modul EM 241 umožňuje přímé připojení automatu S7-200 k analogové telefonní lince a podporuje komunikaci mezi S7-200 a STEP 7–Micro/WIN. Modemový modul také podporuje protokol Modbus slave RTU. Komunikace mezi modemovým modulem a automatem S7-200 probíhá přes rozšiřovací sběrnici.

STEP 7–Micro/WIN obsahuje průvodce, který pomáhá nastavit vzdálený modem nebo modemový modul pro připojení lokálního S7-200 ke vzdálenému zařízení.

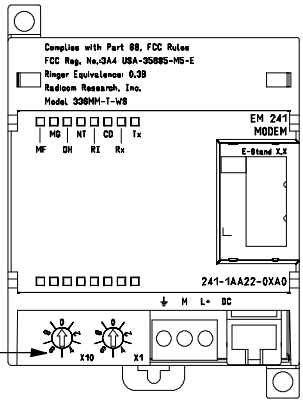
Přehled kapitoly

Funkce modemového modulu 308
Použití průvodce pro konfiguraci modemového modulu 314
Přehled instrukcí a omezení pro modem 318
Instrukce pro modemový modul 319
Vzorový program 323
CPU S7-200 podporující inteligentní moduly 323
Byty speciální paměti pro modemový modul 323
Témata pro pokročilé 325
Formát telefonního čísla pro zprávu 327
Formát textové zprávy 328
Formát zprávy pro přenos dat CPU 329

Funkce modemového modulu

Modemový modul umožňuje připojení S7-200 přímo k analogové telefonní lince; má následující funkce:

- ❏ Připojení mezinárodní telefonní linky
- ❏ Modemové spojení se STEP 7 – Micro/WIN pro programování a hledání chyb (teleservis)
- ❏ Podporuje protokol Modbus RTU
- ❏ Podporuje přenos numerických a textových zpráv na pager
- ❏ Podporuje zprávy SMS
- ❏ Umožňuje přesun dat z CPU do CPU nebo z CPU do Modbus
- ❏ Umožňuje ochranu heslem
- ❏ Zabezpečení zpětným voláním
- ❏ Konfigurace modemového modulu je uložena v CPU

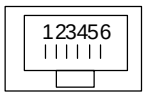


Obr. 10-1 Modemový modul EM 241

Pro konfiguraci modemového modulu můžete použít STEP 7 – Micro/WIN a jeho funkci ”Průvodce modemem”. Specifikaci modemového modulu najdete v příloze A.

Připojení mezinárodní telefonní linky

Modemový modul je standardní 10bitový modem V.34 (33,6 kBaud) a je kompatibilní s většinou interních a externích modemů pro PC. Modemový **modul nekomunikuje s 11bitovými** modemy.



Vývod	Popis	
3	Vyzvánění	Je povoleno obrácené zapojení.
4	Hovor	

Obr. 10-2 Pohled na jack RJ11

Modemový modul se připojuje k telefonní lince pomocí šestipólového čtyřdrátového konektoru RJ11, instalovaného na přední straně modulu. Viz obrázek 10-2.

Pro připojení konektoru RJ11 ke standardní telefonní zásuvce může být v různých zemích nutný adaptér. Více informací najdete v dokumentaci přechodového konektoru.

Rozhraní mezi modemem a telefonní linkou je napájeno z vnějšího zdroje 24 V DC. Ten může být připojen ke zdroji snímačů v CPU nebo k vnějšímu zdroji. Zemnicí svorku modemového modulu spojte s uzemněním systému.

Pokud je modemový modul připojen ke zdroji, nakonfiguruje automaticky telefonní připojení pro provoz v dané zemi. Kód země se volí dvěma otočnými přepínači na přední straně modulu. Dříve než modemový modul připojíte ke zdroji, musíte přepínači nastavit potřebnou volbu země. Nastavení přepínačů pro podporované země najdete v tabulce 10-1.

Tabulka 10-1 Země podporované EM 241

Nastavení přepínačů	Země
00	Austrálie
01	Rakousko
02	Belgie
05	Kanada
06	Čína
08	Dánsko
09	Finsko
10	Francie
11	Německo
12	Řecko
16	Irsko
18	Itálie
22	Lucembursko
25	Nizozemí
26	Nový Zéland
27	Norsko
30	Portugalsko
34	Španělsko
35	Švédsko
36	Švýcarsko
38	Velká Británie
39	USA

Připojení ke STEP 7–Micro/WIN

Modemový modul umožňuje komunikaci se STEP 7–Micro/WIN prostřednictvím telefonní linky (teleservis). Pokud používáte STEP 7–Micro/WIN, nemusíte CPU S7-200 na použití modemového modulu jako vzdáleného modemu ani konfigurovat, ani programovat.

Při použití modemového modulu spolu se STEP 7–Micro/WIN proveďte následující kroky:

1. Vypněte napájení CPU S7-200 a připojte modemový modul na rozšiřovací vstupní/výstupní sběrnici. Nepřipojujte žádné moduly, pokud je CPU S7-200 pod napětím.
2. Připojte modemový modul k telefonní lince. V případě potřeby použijte adaptér.
3. Připojte 24 V DC ke svorkovnici modemového modulu.
4. Připojte uzemnění svorkovnice modemového modulu k uzemnění systému.
5. Nastavte přepínače pro kód země.
6. Připojte CPU S7-200 a modemový modul ke zdroji napájecího napětí.
7. Nakonfigurujte STEP 7–Micro/WIN pro komunikaci s 10bitovým modemem.

Protokol Modbus RTU

Modemový modul můžete nakonfigurovat tak, aby komunikoval jako Modbus RTU slave. Modemový modul přijímá požadavky Modbus přes modemové rozhraní, interpretuje tyto požadavky a přenáší data do CPU nebo z něj. Modul pak generuje odpověď Modbus a odešle ji přes modemové rozhraní.



Tip

Pokud je modemový modul nakonfigurován, aby komunikoval jako Modbus RTU slave, není STEP 7–Micro/WIN schopen komunikovat s modemovým modulem přes telefonní linku.

Modemový modul podporuje funkce Modbus, uvedené v tabulce 10-2.

Funkce Modbus 4 a 16 umožňují čtení nebo zápis maximálně 125 paměťových registrů (250 bytu paměti V) v jednom požadavku. Funkce 5 a 15 zapisují hodnoty do registru obrazu výstupů CPU. Tyto hodnoty mohou být přepsány uživatelským programem.

Adresy Modbus se běžně zapisují jako 5 nebo 6 znakových hodnot, které obsahují typ dat a posunutí. První jeden nebo dva znaky určují typ dat a poslední čtyři znaky volí příslušnou hodnotu v daném typu dat. Zařízení Modbus master mapuje adresy na správné funkce Modbus.

Tabulka 10-2 Funkce Modbus podporované modemovým modulem

Funkce	Popis
Funkce 01	Čtení stavu výstupu
Funkce 02	Čtení stavu vstupu
Funkce 03	Čtení paměťových registrů
Funkce 04	Čtení registrů vstupů (analogových vstupů)
Funkce 05	Zápis na jeden výstup
Funkce 06	Předvolení jednoho registru
Funkce 15	Zápis na více výstupů
Funkce 16	Předvolení více registrů

Tabulka 10-3 uvádí adresy Modbus podporované modemovým modulem a mapování adres Modbus na adresy CPU S7-200.

Pro vytvoření konfiguračního bloku pro protokol Modbus RTU pomocí modemového modulu použijte ”Průvodce modemem”. Dříve než můžete použít protokol Modbus, musíte provést download konfiguračního bloku modemového modulu do datového bloku CPU.

Tabulka 10-3 Mapování adres Modbus do CPU S7-200

Adresy Modbus	Adresy CPU S7-200
000001	Q0.0
000002	Q0.1
000003	Q0.2
...	...
000127	Q15.6
000128	Q15.7
010001	I0.0
010002	I0.1
010003	I0.2
...	...
010127	I15.6
010128	I15.7
030001	AIW0
030002	AIW2
030003	AIW4
...	...
030032	AIW62
040001	VW0
040002	VW2
040003	VW4
...	...
04xxxx	VW 2*(xxxx – 1)

Přenos zpráv pro pager a SMS zpráv

Modemový modul podporuje posílání numerických a textových zpráv pro pager a SMS zpráv (Short Message Service – služba krátkých textových zpráv) na mobilní telefony (pokud je podporuje operátor). Zprávy a telefonní čísla jsou uloženy v konfiguračním bloku modemového modulu, který musí být nahrán do datového bloku CPU S7-200. Pro vytvoření zpráv a telefonních čísel pro konfigurační blok modemového modulu můžete použít průvodce. Ten také vytváří programový kód, který v uživatelském programu umožňuje posílat zprávy.

Přenos numerických zpráv

Přenos numerických zpráv používá tóny tlačítkového tónového telefonu pro posílání numerických hodnot na pager. Modemový modul vytočí požadovanou službu přenosu, vyčká ukončení hlasové zprávy a pak odešle tóny odpovídající číslicím ve zprávě. Ve stránkové zprávě jsou povoleny číslice 0 až 9, hvězdička (*), A, B, C a D. Skutečné znaky zobrazené na pageru pro hvězdičku a znaky A, B, C a D nejsou standardizované; určuje je však pager a operátor.

Přenos textových zpráv

Přenos textových zpráv umožňuje posílání alfanumerických zpráv operátorovi a odtud na pager. Poskytovatelé přenosu textových zpráv mají běžně modemovou linku, která přijímá textové zprávy. Modemový modul používá pro přenos textových zpráv na operátora protokol ”Telelocator Alphanumeric Protocol” (TAP). Tento protokol používá pro příjem zpráv mnoho operátorů.

Short Message Service – Služba krátkých zpráv (SMS)

Přenos SMS zpráv podporují některé mobilní telefony; obecně ty, které jsou GSM kompatibilní. SMS umožňuje modemovému modulu zaslat zprávu poskytovateli SMS prostřednictvím analogové telefonní linky. Poskytovatel SMS pak zprávu vyšle na mobilní telefon a ta se zobrazí na textovém displeji telefonu. Pro posílání zpráv poskytovateli SMS používá modemový modul protokoly ”Telelocator Alphanumeric Protocol” (TAP) a ”Universal Computer Protocol” (UCP). SMS zprávy můžete posílat pouze těm poskytovatelům SMS, kteří na modemové lince tyto protokoly podporují.

Vložené proměnné v textových zprávách pro pager a SMS zprávách

Modemový modul umí vkládat hodnoty dat z CPU do textových zpráv a formátovat hodnoty dat na základě specifikace ve zprávě. Můžete zadat počet číslic vlevo a vpravo od desetinné čárky a také to, zda použít desetinnou čárku, nebo tečku. Pokud uživatelský program přikáže modemovému modulu, aby odeslal textovou zprávu, vyhledá tyto hodnoty v CPU a pak tyto hodnoty zformátuje a vloží je do textové zprávy dříve, než ji odešle operátorovi.

Telefonní číslo operátora zpráv, zpráva a proměnné vložené do zprávy jsou načteny z CPU během několika programových cyklů CPU. V průběhu posílání zprávy by uživatelský program neměl měnit telefonní čísla nebo text zprávy. Aktualizace proměnných vložených do zprávy může v průběhu odesílání pokračovat. Pokud zpráva obsahuje více proměnných, jsou tyto proměnné načítány během více programových cyklů CPU. Pokud chcete, aby byly všechny vložené proměnné ve zprávě jednotné, nesmíte po odeslání zprávy žádnou z vložených proměnných měnit.

Přenos dat

Modemový modul umožňuje pomocí uživatelského programu přesouvat data do jiného CPU nebo zařízení Modbus po telefonní lince. Přenosy dat a telefonní čísla se konfigurují pomocí "Průvodce modemem" a jsou uloženy v konfiguračním bloku modemového modulu. Pak je proveden download konfiguračního bloku do datového bloku CPU S7-200. Průvodce vytváří také programový kód, který umožňuje, aby uživatelský program spustil přenos dat.

Přenos dat může znamenat jednak požadavek načtení dat ze vzdáleného zařízení, jednak požadavek zápisu dat do vzdáleného zařízení. Při datovém přenosu je možné načíst nebo zapsat mezi 1 a 100 wordů dat. Data se přesouvají do paměti V připojeného CPU nebo z něj.

Průvodce umožňuje vytvořit přenos dat, skládající se z jednoho čtení ze vzdáleného zařízení, jednoho zápisu do vzdáleného zařízení nebo kombinace čtení i zápisu vzdáleného zařízení.

Přenos dat používá nakonfigurovaný protokol modemového modulu. Jestliže je modemový modul nakonfigurován, aby podporoval protokol PPI (tam, kde odpovídá STEP 7 – Micro/WIN), použije protokol PPI i pro přenos dat. Pokud je modemový modul nakonfigurován, aby podporoval protokol Modbus RTU, jsou přenosy dat uskutečněny s použitím protokolu Modbus.

Telefonní číslo vzdáleného zařízení, požadavek na přenos dat a přenášená data jsou načtena z CPU během několika programových cyklů CPU. V průběhu posílání zprávy by uživatelský program neměl měnit telefonní čísla nebo text zprávy. Také byste v době posílání zprávy neměli měnit přenášená data.

Pokud je vzdáleným zařízením jiný modemový modul, je možné při přesunu dat použít funkci ochrany heslem vložením hesla vzdáleného modemového modulu do konfigurace telefonního čísla. Při přesunu dat není možné použít funkci zpětného volání.

Ochrana heslem

Zabezpečení modemového modulu heslem je volitelné; aktivuje se pomocí průvodce. Heslo používané modemovým modulem není stejné jako heslo pro CPU. Heslo modemového modulu je samostatné heslo o 8 znacích, které musí volající zadat do modemového modulu dříve, než je mu umožněn přístup k připojenému CPU. Heslo je uloženo v paměti V CPU jako součást konfiguračního bloku modemového modulu. Konfigurační blok modemového modulu musí být nahrán do datového bloku připojeného CPU.

Pokud má CPU aktivováno v systémovém bloku zabezpečení heslem, musí volající zadat heslo pro CPU, aby získal přístup k jakékoliv funkci chráněné heslem.

Zabezpečení zpětným voláním

Zabezpečení modemového modulu zpětným voláním je volitelné a konfiguruje se pomocí průvodce. Funkce zpětného volání poskytuje další zabezpečení připojeného CPU tím, že umožňuje přístup k CPU pouze z předdefinovaných telefonních čísel. Když je povolena funkce zpětného volání, odpoví modemový modul na jakékoliv příchozí volání, ověří volajícího a pak linku odpojí. Jestliže má volající oprávnění, vytočí potom modemový modul předdefinované telefonní číslo volajícího a povolí přístup k CPU.

Modemový modul podporuje tři režimy zpětného volání:

- Zpětné volání na jedno předdefinované telefonní číslo
- Zpětné volání na více předdefinovaných zpětných čísel
- Zpětné volání na libovolné telefonní číslo

Režim zpětného volání se volí zaškrtnutím příslušné volby v průvodci a následným definováním telefonních čísel pro zpětné volání. Telefonní čísla pro zpětné volání jsou uložena v konfiguračním bloku modemového modulu, uloženém v datovém bloku připojeného CPU.

Nejjednodušší forma zpětného volání je na jediné předdefinované telefonní číslo. Pokud je v konfiguračním bloku uloženo pouze jedno číslo pro zpětné volání, odpovídá modemový modul na příchozí volání kdykoliv, uvědomí volajícího, že je aktivováno zpětné volání, odpojí volajícího a potom vytočí číslo pro zpětné volání, specifikované v konfiguračním bloku.

Modemový modul také podporuje zpětné volání na více předdefinovaných telefonních čísel. V tomto režimu je volající požádán o telefonní číslo. Pokud zadané číslo odpovídá jednomu z předdefinovaných telefonních čísel v konfiguračním bloku modemového modulu, odpojí modul volajícího a zavolá zpět s použitím odpovídajícího telefonního čísla z konfiguračního bloku. Uživatel může nakonfigurovat až 250 čísel pro zpětné volání.

Tam, kde existuje více předdefinovaných čísel pro zpětné volání, musí číslo pro zpětné volání zadané při připojování k modemovému modulu přesně odpovídat číslu v konfiguračním bloku modemového modulu s výjimkou prvních dvou číslic. Jestliže je nakonfigurované číslo pro zpětné volání například 91(123)4569999 vzhledem k nutnosti vytočit vnější linku (9) a meziměsto (1), může být číslo zadané pro zpětné volání kterékoliv z následujících:

- 91(123)4569999
- 1(123)4569999
- (123)4569999

Všechna výše uvedená telefonní čísla jsou považována za odpovídající číslo pro zpětné volání. Modemový modul použije při provádění zpětného volání telefonní číslo ze svého konfiguračního bloku, v tomto příkladu 91(123)4569999. Při konfigurování více čísel pro zpětné volání se ujistěte, že jsou všechna telefonní čísla jedinečná s výjimkou prvních dvou číslic. Při porovnávání telefonních čísel pro zpětné volání se používají pouze číselné znaky telefonního čísla. Při tomto porovnávání jsou speciální znaky (jako čárky nebo závorky) ignorovány.

Zpětné volání na jakékoliv telefonní číslo se v průvodci nastavuje pomocí volby ”Použít zpětné volání pro všechna čísla” během konfigurace zpětného volání. Pokud je tato volba vybrána, odpovídá modemový modul na příchozí volání a požaduje telefonní číslo pro zpětné volání. Poté, co volající zadá telefonní číslo, odpojí ho modemový modul a vytočí toto telefonní číslo. Tento režim zpětného volání umožňuje účtovat telefonní poplatky pouze za telefonní připojení modulu a nezabezpečuje CPU S7-200. Pokud použijete tento režim zpětného volání, měl by být modul zabezpečen heslem.

Funkce zabezpečení heslem a zpětného volání mohou být pro modemový modul aktivovány zároveň. Modul požaduje po volajícím, aby zadal správné heslo dříve, než začne zpracovávat zpětné volání.

Tabulka konfigurace modemového modulu

Všechny textové zprávy, telefonní čísla, informace o přenosu dat, čísla pro zpětné volání a ostatní nastavení jsou uloženy v tabulce konfigurace modemového modulu, která musí být nahrána do paměti V CPU S7-200. Při tvorbě tabulky konfigurace modemového modulu vás provede průvodce. Potom STEP 7–Micro/WIN umístí tabulku konfigurace modemového modulu do datového bloku, který je následně nahrán do CPU S7-200.

Modemový modul načte tuto tabulku konfigurace z CPU při spuštění a také do pěti sekund po jakémkoliv přechodu CPU z režimu STOP do režimu RUN. Modemový modul nenačte novou tabulku konfigurace z CPU, dokud je modul spojen online se STEP 7–Micro/WIN. Pokud je proveden download nové tabulky konfigurace, když je modemový modul online, načte modul novou tabulku konfigurace až po skončení online relace.

Pokud modemový modul zjistí v tabulce konfigurace chybu, začne na přední straně modulu blikat LED ”MG – modul v pořádku” (MG). Zkontrolujte informační dialogové okno STEP 7–Micro/WIN programovatelného automatu nebo přečtěte hodnotu v SMW220 (modul na pozici 0), abyste získali informace o chybě konfigurace. Chyby konfigurace modemového modulu jsou uvedeny v tabulce 10-4. Pokud pro vytvoření tabulky konfigurace modemového modulu použijete průvodce, STEP 7–Micro/WIN data před vytvořením tabulky konfigurace zkontroluje.

Tabulka 10-4 Chyby konfigurace EM 241 (hexadecimální)

Error	Popis
0000	Bez chyby
0001	Chybí vnější napájení 24 V DC
0002	Porucha modemu
0003	Chybí ID konfiguračního bloku – Identifikace EM 241 na začátku tabulky konfigurace pro tento modul neplatí.
0004	Konfigurační blok mimo rozsah – ukazatel tabulky konfigurace neukazuje na paměť V nebo je některá část tabulky mimo rozsah paměti V připojené CPU.
0005	Chyba konfigurace – je povoleno zpětné volání, počet telefonních čísel pro zpětné volání se rovná 0 nebo je větší než 250. Počet zpráv je větší než 250. Počet telefonních čísel pro přenos zpráv je větší než 250 nebo je délka telefonních čísel pro přenos zpráv větší než 120 bytů.
0006	Chyba výběru kódu země – výběr země pomocí dvou otočných přepínačů není platný.
0007	Příliš dlouhé telefonní číslo – je povoleno zpětné volání a délka čísla pro něj je větší než maximální.
0008 až 00FF	Rezervováno
01xx	Chyba v čísle xx pro zpětné volání – v telefonním čísle pro zpětné volání xx jsou nepřípustné znaky. Hodnota xx je 1 pro první číslo pro zpětné volání, 2 pro druhé atd.
02xx	Chyba v telefonním čísle xx – jedno z polí v čísle xx telefonu pro přenos zpráv nebo v čísle xx telefonu pro pro přesun dat obsahuje nepřípustnou hodnotu. Hodnota xx je 1 pro první telefonní číslo, 2 pro druhé atd.
03xx	Chyba ve zprávě xx – zpráva nebo přesun dat číslo xx překračuje maximální délku. Hodnota xx je 1 pro první zprávu, 2 pro druhou atd.
0400 až FFFF	Rezervováno

Stavové LED modemového modulu

Modemový modul má na předním panelu 8 LED diod pro zobrazení stavu. Tabulka 10-5 Stavové kontrolkz EM 241

Tabulka 10-5 Stavové kontrolky EM 241

Kontrolka	Popis
MF	Porucha modulu – tato kontrolka svítí, pokud modul zjistí poruchový stav jako například: • Chybí vnější napájení 24 V DC. • Vypršel čas obvodu sledování funkce vstupů a výstupů. • Porucha modemu • Chyba komunikace s lokálním CPU
MG	Modul v pořádku – tato kontrolka svítí, pokud modul nehlásí žádnou chybu. LED "Modul v pořádku" bliká, pokud je chyba v tabulce konfigurace nebo uživatel zvolil nepřípustné nastavení kódu země pro rozhraní telefonní linky. Zkontrolujte u programovatelného automatu informační dialogové okno STEP 7–Micro/WIN nebo přečtěte hodnotu v SMW220 (pro modul na pozici 0), abyste získali informace o chybě konfigurace.
OH	Hovor – tato LED svítí, pokud EM 241 aktivně používá telefonní linku.
NT	Není oznamovací tón (No Dial Tone) – indikuje chybu a rozsvítí se, pokud dostal EM 241 příkaz odeslat zprávu a na telefonní lince není oznamovací tón. Chyba je to pouze tehdy, když byl EM 241 nakonfigurován, aby před vytáčením kontroloval přítomnost oznamovacího tónu. LED zůstane po neúspěšném pokusu o vytočení čísla svítit přibližně 5 sekund.
RI	Indikátor vyzvánění – tato LED znamená, že EM 241 přijímá příchozí hovor.
CD	Detekce nosné – tato LED znamená, že bylo navázáno spojení se vzdáleným modemem.
Rx	Příjem dat – tato LED bliká, když modem přijímá data.
Tx	Odesílání dat – tato LED bliká, když modem odesílá data.

Použití průvodce pro konfiguraci modemového modulu



Průvodce modemem

Spusťte průvodce z menu STEP 7–Micro/WIN nebo z části "Nástroje" v navigační liště.

Abyste mohli tohoto průvodce použít, musí být projekt zkompilován a nastaven na režim symbolického adresování. Pokud jste svůj program ještě nezkompilovali, udělejte to.

1. V prvním dialogovém okně "Průvodce modemem" vyberte "Konfigurovat modemový modul EM 241" a klikněte na "Další >".
2. Průvodce vyžaduje pozici modemového modulu vzhledem k CPU S7-200, aby generoval správný programový kód. Pro automatické načtení inteligentních modulů připojených k CPU klikněte na tlačítko "Přečíst manuály". Rozšiřovací moduly jsou číslovány postupně počínaje nulou. Dvakrát klikněte na modemový modul, který chcete konfigurovat, nebo nastavte hodnotu do pole pozice modemového modulu. Klikněte na "Další >".

   U CPU S7-200 s verzí firmware nižší než 1.2 musíte instalovat inteligentní modul přímo vedle CPU, aby průvodce modul konfiguroval.
3. Dialogové okno pro ochranu heslem umožňuje povolit tuto ochranu a přiřadit modulu heslo o 1 až 8 znacích. Toto heslo je nezávislé na hesle pro CPU S7-200. Když je modul chráněn heslem, je od každého, kdo se pokouší spojit s CPU S7-200 přes modem, požadováno zadání správného hesla. Pokud chcete, vyberte ochranu heslem a vložte heslo. Klikněte na "Další >".

4. Modemový modul podporuje dva komunikační protokoly: Protokol PPI (pro komunikaci se STEP 7–Micro/WIN) a protokol Modbus RTU. Výběr protokolu závisí na typu zařízení, které se používá jako vzdálený komunikační partner. Toto nastavení řídí použitý komunikační protokol, pokud modemový modul odpovídá na volání a pokud iniciuje přesun dat. Vyberte příslušný protokol a klikněte na "Další >".
5. Modul můžete nakonfigurovat pro posílání numerických a textových zpráv na pager nebo SMS zpráv mobilním telefonům. Zaškrtněte políčko "Povolit v této konfiguraci zprávy" a klikněte na tlačítko "Konfigurovat zprávy" pro definování zpráv a telefonních čísel příjemců.
6. Když konfigurujete zprávu, která má být odeslána na pager nebo mobilnímu telefonu, musíte definovat text zprávy a telefonní číslo. Vyberte záložku "Zprávy" v dialogovém okně "Konfigurace zprávy"a klikněte na tlačítko "Nová zpráva". Vložte text zprávy a specifikujte hodnoty všech dat CPU, která mají být vložena do zprávy. Chcete–li vložit do zprávy nějakou hodnotu dat CPU, klikněte na tlačítko "Vložit data...". Specifikujte adresu hodnoty dat CPU (např. VW100), formát zobrazení (např. signed integer) a číslice nalevo a napravo od desetinné čárky. Můžete také specifikovat, zda chcete použít desetinnou čárku, nebo desetinnou tečku.
   - Numerické stránkové zprávy jsou omezeny na číslice 0 až 9, písmena A, B, C a D a hvězdičku (*). Maximální povolená délka numerické zprávy pro pager se mění podle operátora.
   - Textové zprávy mohou mít délku až 119 znaků a obsahovat libovolné alfanumerické znaky.
   - Textové zprávy mohou obsahovat libovolný počet vložených proměnných.
   - Proměnné se mohou vkládat z paměti V, M, SM, I, Q, S, T, C nebo AI připojené CPU.
   - Před hexadecimálními daty je zobrazeno "16#". Počet znaků v hodnotě je závislý na velikosti proměnné. Například VW100 se zobrazí jako 16#0123.
   - Počet číslic vlevo od desetinné čárky musí být dostatečně velký na to, aby zobrazil předpokládaný rozsah hodnot včetně záporného znaménka, pokud je hodnota dat integer se znaménkem nebo hodnota s plovoucí desetinnou čárkou.
   - Pokud je formát dat integer a počet číslic vpravo od desetinné čárky není nulový, je hodnota integer zobrazena jako integer s upraveným měřítkem. Pokud například VW100 = 1234 a vy zadáte dvě číslice vpravo od desetinné čárky, jsou data zobrazena jako "12,34".
   - Pokud je hodnota dat větší, než je možné zobrazit ve specifikovaném poli, uvede modemový modul znak # na všech místech pro znaky hodnoty dat.
7. Telefonní čísla se konfigurují vybráním záložky "Telefonní čísla" v dialogovém okně "Konfigurace zprávy". Chcete–li přidat nové telefonní číslo, klikněte na tlačítko "Nové telefonní číslo...". Když je telefonní číslo nakonfigurováno, musí být přidáno do projektu. Vyberte telefonní číslo ve sloupci "Dostupná telefonní čísla" a klikněte na okénko se šipkou vpravo, abyste telefonní číslo přidali do aktuálního projektu. Jakmile jste jednou telefonní číslo přidali do aktuálního projektu, můžete toto číslo vybrat a přidat k němu symbolický název pro používání ve vašem programu.

   Telefonní číslo se skládá z několika polí, která závisí na typu přenosu zpráv vybraném uživatelem.
   - Volba protokolu pro přenos zpráv říká modulu, který protokol má použít při odesílání zprávy operátorovi. Numerický pager podporuje pouze numerický protokol. Textové služby obvykle vyžadují TAP (Telelocator Alphanumeric Protocol). Operátoři SMS zpráv podporují TAP nebo UCP (Universal Computer Protocol). Pro přenos SMS zpráv se normálně používají tři různé služby UCP. Většina poskytovatelů podporuje příkaz 1 nebo 51. Ověřte si u operátora SMS zpráv, jaký protokol a příkazy vyžaduje.
   - Pole "Popis tohoto kontaktu" umožňuje přidat textový popis telefonního čísla.

– Pole ”Telefonní číslo” je telefonní číslo operátora služby přenosu zpráv. U textových zpráv je to telefonní číslo modemové linky, kterou operátor používá pro příjem těchto zpráv. U přenosu numerických zpráv je to přímo telefonní číslo pageru. Modemový modul povoluje maximální počet 40 znaků pro pole telefonního čísla. V telefonních číslech, která modemový modul vytáčí, jsou povoleny následující znaky:

0 až 9	zadání z klávesnice telefonu
A, B, C, D, *, #	znaky DTMF (pouze tónová tlačítková volba)
,	pozastaví vytáčení na dobu 2 sekund
!	dá modemu příkaz generovat rychlé zavěšení
@	počkat 5 sekund
W	před pokračováním počkat na oznamovací tón
( )	ignoruje se (je možné použít pro formátování telefonního čísla)

– Pole ”Telefonní číslo nebo ID pageru”je určeno pro vložení čísla pageru nebo čísla mobilního telefonu příjemce zprávy. Toto číslo nesmí obsahovat jiné znaky než číslice 0 až 9. Je přípustných maximálně 20 znaků.

– Pole ”Heslo” je volitelné pro zprávu TAP. Někteří operátoři požadují heslo, ale normálně by toto pole mělo být necháno prázdné. Modemový modul povoluje heslo až o 15 znacích.

– Pole ”Číslo odchozího telefonu” umožňuje identifikaci modemového modulu v SMS zprávě. Toto pole požadují někteří operátoři, kteří používají příkazy UCP. Někteří operátoři mohou v tomto poli požadovat minimální počet znaků. Modemový modul povoluje až 15 znaků.

– Pole ”Standard modemu”se používá v případech, kdy se modemový modul a modem operátora nemohou dohodnout na standardu modemu. Implicitní hodnota je V.34 (33,6 kBaud).

– Pole ”Datový formát” umožňuje nastavit počet datových bitů a paritu, použité modemem při odesílání zprávy operátorovi. TAP běžně používal 7 datových bitů a sudou paritu, ale někteří operátoři používají 8 datových bitů a žádnou paritu. UCP vždy používá 8 datových bitů a žádnou paritu. Ověřte si u operátora, jaké nastavení použít.

8. Modemový modul můžete nakonfigurovat, aby přenášel data jiné CPU S7-200 (pokud byl vybrán protokol PPI) nebo aby přenášel data zařízení Modbus (pokud byl vybrán protokol Modbus). Zaškrtněte políčko ”Povolit v této konfiguraci přesun dat CPU–CPU” a klikněte na tlačítko ”Konfigurovat CPU–CPU”, abyste mohli definovat přesuny dat a telefonní čísla vzdálených zařízení.

9. Když nastavujete přesun dat z CPU na CPU nebo z CPU na Modbus, musíte definovat data, která se mají přesouvat, a telefonní číslo vzdáleného zařízení. Vyberte záložku ”Přenosy dat“ v dialogovém okně ”Konfigurace datových přenosů” a klikněte na tlačítko ”Nový přenos”. Přenos dat se skládá ze čtení dat ze vzdáleného zařízení, zápisu dat do vzdáleného zařízení nebo ze čtení i zápisu do vzdáleného zařízení. Pokud je vybráno čtení i zápis, je nejprve provedeno čtení a potom zápis.

   Při každém čtení nebo zápisu může být přesunuto až 100 wordů. Data se musí přesouvat do paměti V lokálního CPU nebo z něj. Průvodce vždy popisuje paměťová místa ve vzdáleném zařízení, jako kdyby bylo vzdálené zařízení CPU S7-200. Pokud je vzdálené zařízení Modbus, probíhá přenos do paměťových registrů v zařízení Modbus nebo z nich (adresa 04xxxx). Ekvivalentní adresa Modbus (xxxx) se určí následujícím způsobem:

   Adresa Modbus = 1 + (adresa v paměti V / 2)
   Adresa v paměti V = (adresa Modbus – 1) * 2

10. Záložka ”Telefonní čísla” v dialogovém okně ”Konfigurace datových přenosů” umožňuje definovat telefonní čísla pro přenosy dat z CPU do CPU nebo z CPU do Modbus. Chcete–li přidat nové telefonní číslo, klikněte na tlačítko ”Nové telefonní číslo...”. Když je telefonní číslo nakonfigurováno, musí být přidáno do projektu. Vyberte telefonní číslo ve sloupci ”Dostupná telefonní čísla” a klikněte na okénko se šipkou vpravo, abyste telefonní číslo přidali do aktuálního projektu. Jakmile jste jednou telefonní číslo přidali do aktuálního projektu, můžete toto číslo vybrat a přidat k němu symbolický název pro použití ve vašem programu.

    Pole ”Popis” a ”Telefonní číslo” jsou stejná jako ta, která byla popsána výše pro přenos zpráv. Pole ”Heslo” se vyžaduje, jestliže je vzdálené zařízení modemový modul a byla aktivována ochrana heslem. Pole ”Heslo” v lokálním modemovém modulu musí být nastaveno na heslo vzdáleného modemového modulu. Lokální modemový modul toto heslo zadá, když to požaduje vzdálený modemový modul.

11. Zpětné volání způsobí, že modemový modul automaticky odpojí a vytočí předdefinované telefonní číslo po tom, co přijme příchozí volání ze vzdáleného STEP 7–Micro/WIN. Abyste mohli konfigurovat telefonní čísla pro zpětné volání, zaškrtněte políčko ”Povolit v této konfiguraci zpětné volání” a klikněte na tlačítko ”Konfigurace zpětného volání...”. Klikněte na ”Další >”.

12. Dialogové okno ”Konfigurace zpětného volání...” umožňuje vkládat telefonní čísla, která modemový modul používá, když odpovídá na příchozí volání. Zaškrtněte ”Povolit zpětná volání pouze pro některá čísla”, jestliže mají být čísla pro zpětné volání předem definována. Pokud má modemový modul zpětně volat na kterékoliv číslo (kvůli hovorům na účet volaného), zaškrtněte volbu ”Povolit zpětné volání pro všechna čísla”.

    Pokud chcete pro zpětně volání použít pouze specifikovaná telefonní čísla, klikněte na tlačítko ”Nové telefonní číslo”, požadujete–li přidat telefonní čísla pro zpětné volání. Dialogové okno ”Vlastnosti zpětného volání” umožňuje vložit předdefinovaná telefonní čísla pro zpětné volání a popis těchto čísel. Číslo pro zpětné volání je takové telefonní číslo, které modemový modul vytáčí při zpětném volání. Toto telefonní číslo by mělo obsahovat všechny číslice potřebné pro připojení k vnější lince, pauzu pro čekání na vnější linku, připojení na meziměsto atd.

    Po vložení nového telefonního čísla pro zpětné volání musíte toto číslo přidat do projektu. Vyberte telefonní číslo ve sloupci ”Dostupná čísla pro zpětné volání” a klikněte na okénko se šipkou vpravo, abyste telefonní číslo přidali do aktuálního projektu.

13. Můžete nastavit počet pokusů o vytočení čísla, které má modemový modul uskutečnit při odesílání zprávy nebo při přesunu dat. Modemový modul hlásí uživatelskému programu chybu pouze tehdy, jsou–li všechny pokusy o vytočení čísla a odeslání zprávy neúspěšné.

    U některých telefonních linek není slyšet oznamovací tón, když se zvedne sluchátko. Běžně modemový modul hlásí uživatelskému programu chybu, pokud není přítomen oznamovací tón, když modemový modul dostane příkaz odeslat zprávu nebo provést zpětné volání. Abyste umožnili vytáčení čísla na lince bez oznamovacího tónu, zaškrtněte okno ”Povolit vytáčení bez oznamovacího tónu”.

14. ”Průvodce modemem” vytvoří pro modemový modul konfigurační blok a požaduje po uživateli, aby vložil počáteční adresu, kde jsou uložena konfigurační data modemového modulu. Konfigurační blok modemového modulu je uložen ve V paměti CPU. STEP 7–Micro/WIN zapíše konfigurační blok do datového bloku projektu. Velikost konfiguračního bloku se mění v závislosti na počtu konfigurovaných zpráv a telefonních čísel. Můžete vybrat adresu v paměti V, kam chcete konfigurační blok uložit, nebo klikněte na tlačítko ”Navrhnout adresu”, chcete–li, aby průvodce navrhl adresu volného bloku paměti o správné velikosti v paměti V. Klikněte na ”Další >”.

15. Závěrečným krokem při konfigurování modemového modulu je zadání adresy v paměti Q pro příkazový byte modemového modulu. Adresu v paměti Q můžete stanovit spočítáním výstupních bytů použitých digitálními výstupními moduly připojenými k S7-200 před modemovým modulem. Klikněte na ”Další >”.

16. Nyní průvodce generuje projektové komponenty pro vaši zvolenou konfiguraci (programový blok a datový blok) a zpřístupní tento kód pro použití ve vašem programu. Poslední dialogové okno průvodce zobrazí komponenty vámi požadované konfigurace projektu. Musíte provést download konfiguračního bloku modemového modulu (datový blok) a programového bloku do CPU S7-200.

Přehled instrukcí a omezení pro modem

Průvodce usnadňuje řízení modemového modulu tak, že vytváří jednoznačné instrukční podprogramy na základě pozice modulu a nastavení, které jste si vybrali. Každá instrukce má prefix ”MODx_”, v němž x je pozice modulu.

Požadavky na použití instrukcí modemového modulu EM 241

Pokud používáte instrukce modemového modulu, vezměte v úvahu tyto požadavky:

- Instrukce modemového modulu používají tři podprogramy.
- Díky instrukcím modemového modulu se zvětšuje část paměti potřebná pro uživatelský program až o 370 bytů. Pokud vymažete podprogram, který nepoužíváte, můžete v případě potřeby opět spustit průvodce a instrukci znovu vytvořit.
- Musíte zajistit, aby v danou dobu byla aktivní pouze jedna instrukce.
- Instrukce není možné používat v podprogramu přerušení.
- Modemový modul načte informace z tabulky konfigurace při prvním studeném startu a po přechodu z režimu STOP do režimu RUN. Jakoukoliv změnu provedenou uživatelským programem v tabulce konfigurace modul neregistruje až do změny režimu nebo příštího vypnutí a obnovení napájení.

Použití instrukcí modemového modulu EM 241

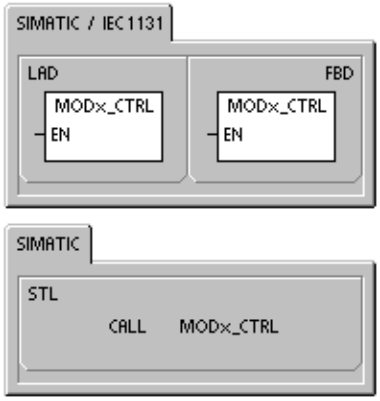
Pokud chcete použít instrukce pro modemový modul, proveďte v programu S7-200 tyto kroky:

1. Použijte průvodce a vytvořte tabulku konfigurace modemového modulu.
2. Vložte do svého programu instrukci MODx_CTRL a použijte kontakt SM0.0 pro její provedení v každém programovém cyklu (scan).
3. Vložte instrukci MODx_MSG pro každou zprávu, kterou potřebujete odeslat.
4. Vložte instrukci MODx_XFR pro každý přenos dat.

Instrukce pro modemový modul

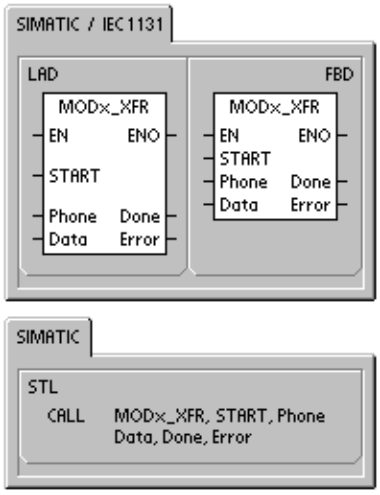
Instrukce MODx_CTRL

Instrukce MODx_CTRL (Řízení) se používá pro aktivaci a inicializaci modemového modulu. Tuto instrukci je třeba volat v každém programovém cyklu; musí být v projektu použita pouze jednou.



Instrukce MODx_XFR

Instrukce MODx_XFR (Přenos dat) se používá jako příkaz pro modemový modul, aby načítal data z jiné CPU S7-200 či zařízení Modbus nebo je do nich zapisoval. Tato instrukce vyžaduje 20 až 30 sekund od doby, kdy je spuštěn vstup START do nastavení bitu "Done" (Dokončeno).



Bit EN musí být zapnutý, aby byl příkaz aktivován, a musí zůstat zapnutý, dokud se nenastaví bit "Done", který signalizuje ukončení procesu. Příkaz XFR je poslán modemovému modulu v každém programovém cyklu, když je zapnutý vstup START a modul právě nepracuje. Pulzy by měly být na vstupu START generovány pomocí detekce hrany, která aktivuje odeslání pouze jednoho příkazu.

"Phone" (Telefon) je jedno z telefonních čísel pro přenos dat. Můžete použít symbolický název, který jste přiřadili každému telefonnímu číslu pro přenos dat, když bylo číslo definováno pomocí průvodce.

"Data" je číslo jednoho z definovaných přenosů dat. Můžete použít symbolický název, který jste přiřadili přenosu dat, když byl požadavek definován pomocí průvodce.

"Done" (Dokončeno) je bit, který se zapne, když modul ukončí přenos dat.

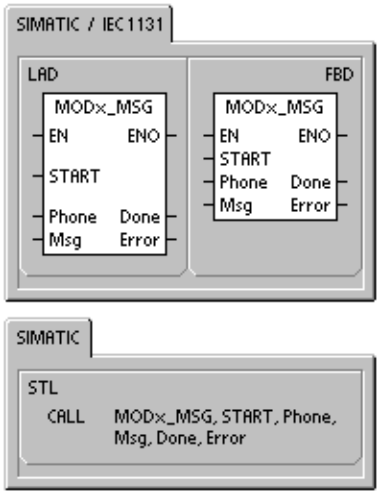
"Error" (Chyba) je byte, který obsahuje výsledek přenosu dat. Tabulka 10-4 definuje možné chyby, které mohou vyplynout z provedení této instrukce.

Tabulka 10-6 Parametry instrukce MODx_XFR

Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, M, S, SM, T, C, V, L, signálový tok
Phone, Data	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, konstanta, *VD, *AC, *LD
Done	BOOL	I, Q, M, S, SM, T, C, V, L
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD

Instrukce MODx_MSG

Instrukce MODx_MSG (Odeslat zprávu) se používá pro odeslání zprávy z modemového modulu. Tato instrukce vyžaduje 20 až 30 sekund od doby, kdy je spuštěn vstup START do nastavení bitu "Done" (Dokončeno).



Bit EN musí být zapnutý, aby byl příkaz aktivován, a musí zůstat zapnutý, dokud se nenastaví bit "Done", který signalizuje ukončení procesu. Příkaz MSG je poslán modemovému modulu v každém programovém cyklu, když je zapnutý vstup START a modul právě nepracuje. Pulzy by měly být na vstupu START generovány pomocí detekce hrany, která aktivuje odeslání pouze jednoho příkazu.

"Phone" je jedno z telefonních čísel pro přenos zpráv. Můžete použít symbolický název, který jste přiřadili každému telefonnímu číslu pro přenos zpráv, když bylo číslo definováno pomocí průvodce.

"Msg" je číslo jedné z definovaných zpráv. Můžete použít symbolický název, který jste zprávě přiřadili, když byla definována pomocí průvodce.

"Done" (Dokončeno) je bit, který se zapne, když modemový modul ukončí odesílání zprávy operátorovi.

"Error" (Chyba) je byte, který obsahuje výsledek tohoto požadavku na modul. Tabulka 10-8 definuje možné chyby, které mohou vyplynout z provedení této instrukce.

Tabulka 10-7 Parametry instrukce MODx_MSG

Vstupy/Výstupy	Typ dat	Operandy
START	BOOL	I, Q, M, S, SM, T, C, V, L, signálový tok
Phone, Msg	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, konstanta, *VD, *AC, *LD
Done	BOOL	I, Q, M, S, SM, T, C, V, L
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD

Tabulka 10-8 Chybové kódy hlášené instrukcemi MODx_MSG a MODx_XFR

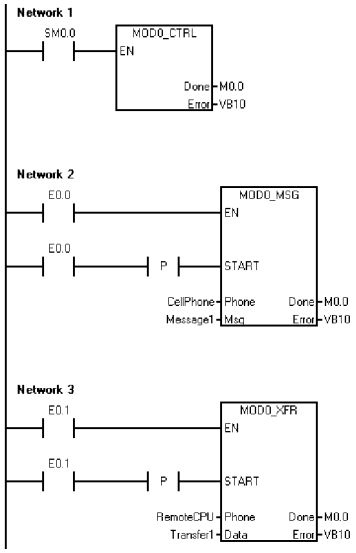
Error	Popis
0	Bez chyby
Chyby telefonní linky	
1	Není přítomen oznamovací tón
2	Linka obsazena
3	Chyba vytáčení
4	Volané číslo neodpovídá
5	Vypršení spojovací doby (ke spojení nedošlo během 1 minuty)
6	Spojení přerušeno nebo neznámá odezva
Chyby v příkazu	
7	Numerická stránková zpráva obsahuje nepřípustné znaky
8	Telefonní číslo (vstup Phone) mimo rozsah
9	Zpráva nebo přesun dat (vstup Msg nebo Data) mimo rozsah
10	Chyba v textové zprávě nebo zprávě přesunu dat
11	Chyba v telefonním čísle pro přenos zpráv nebo přesun dat
12	Nepřípustná operace (např. počet pokusů o vytočení nastaven na nulu)
Chyby operátora	
13	Bez odezvy (vypršení času) od služby přenosu zpráv
14	Služba přenosu zpráv odpojena z neznámého důvodu
15	Zpráva zrušena uživatelem (zablokován příkazový bit)
TAP – chyby zpráv ohlášené operátorem	
16	Příjem vzdáleného odpojení (operátor zrušil relaci)
17	Přihlášení nebylo přijato službou přenosu zpráv (nesprávné heslo)
18	Blok nebyl přijat službou přenosu zpráv (chyba kontrolního součtu nebo přenosu)
19	Blok nebyl přijat službou přenosu zpráv (neznámý důvod)
UCP – Poskytovatel služby ohlášené operátorem	
20	Neznámá chyba
21	Chyba kontrolního součtu
22	Syntaktická chyba
23	Operace není systémem podporována (nepřípustný příkaz)
24	V této době není operace povolena
25	Aktivní blokování účastníka (černá listina)
26	Neplatná adresa volajícího
27	Neúspěšné prokazování totožnosti
28	Neúspěšný legitimovací kód
29	Neplatný GA
30	Opakování není povoleno
31	Legitimovací kód pro opakování, neúspěch
32	Prioritní spojení nepovoleno
33	Legitimovací kód pro prioritní spojení, neúspěch
34	Urgentní zpráva nepovolena
35	Legitimovací kód pro urgentní zprávu, neúspěch
36	Spojení na účet volaného nepovoleno
37	Legitimovací kód pro spojení na účet volaného, neúspěch

Tabulka 10-8 Chybové kódy hlášené instrukcemi MODx_MSG a MODx_XFR, pokračování

Error	Popis
UCP – operátor hlásí chyby zprávy SMS (pokračování)	
38	Odložené doručení nepovoleno
39	Nový AC neplatný
40	Nový legitimovací kód nepovolen
41	Neplatný standardní text
42	Neplatná délka časového úseku
43	Systém nepodporuje typ zprávy
44	Zpráva příliš dlouhá
45	Požadovaný standardní text neplatný
46	Typ zprávy neplatný pro typ pageru
47	Zpráva nenalezena v SMSC
48	Rezervováno
49	Rezervováno
50	Účastník zavěsil
51	Nepodporována skupina faxu
52	Nepodporován typ faxové zprávy
Chyby přesunu dat	
53	Vypršení doby pro zprávu (není odezva od vzdáleného zařízení)
54	Vzdálené CPU provádí upload nebo download
55	Chyba přístupu (paměť mimo rozsah, nepřípustný typ dat)
56	Chyba komunikace (neznámá odpověď)
57	Chyba kontrolního součtu nebo CRC v odpovědi
58	Vzdálený EM 241 nastaven na zpětné volání (nepovoleno)
59	Vzdálený EM 241 zamítl zadané heslo
60 až 127	Rezervováno
Chyby v používání instrukcí	
128	Tento požadavek nelze zpracovat. Modemový modul zpracovává jiný požadavek, nebo pro tento požadavek nebyl pulz START.
129	Chyba modemového modulu: • Umístění modemového modulu nebo adresa v paměti Q, která byla nakonfigurována pomocí ”Průvodce modemem”, neodpovídá skutečnému umístění nebo adrese v paměti. • Viz SMB8 až SMB21 (ID vstupního/výstupního modulu a chybový registr)

Vzorový program

Příklad: Modemový modul




```
Network 1       //Volá podprogram
                //MOD0_CTRL v každém
                //programovém cyklu.
LD      SM0.0
CALL    MOD0_CTRL

Network 2       //Odešle textovou zprávu
                //na mobilní telefon.
LD      I0.0
EU
=       L63.7
LD      I0.0
CALL    MOD0_MSG, L63.7, Cell Phone,
        Message1, M0.0, VB10

Network 3       //Přenese data do
                //vzdáleného CPU.
LD      I0.1
EU
=       L63.7
LD      I0.1
CALL    MOD0_XFR, L63.7, Vzdálený CPU,
        Transfer1, M0.0, VB10
```

CPU S7-200 podporující inteligentní moduly

Modemový modul je inteligentní rozšiřovací modul určený pro spolupráci s CPU S7-200 uvedenými v tabulce 10-9.

Tabulka 10-9 Kompatibilita modulu EM 214 s CPU S7-200

CPU	Popis
Model CPU 222 verze 1.10 nebo vyšší	CPU 222 DC/DC/DC a CPU 222 AC/DC/relé
Model CPU 224 verze 1.10 nebo vyšší	CPU 224 DC/DC/DC a CPU 224 AC/DC/relé
Model CPU 224XP verze 2.00 nebo vyšší	CPU 224XP DC/DC/DC a CPU 224XP AC/DC/relé
Model CPU 226 verze 1.00 nebo vyšší	CPU 226 DC/DC/DC a CPU 226 AC/DC/relé

Byty speciální paměti pro modemový modul

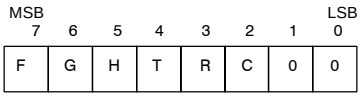
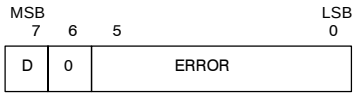
Každému inteligentnímu modulu je na základě jeho fyzického umístění v rozšiřovací sběrnici přiděleno padesát bytů speciální paměti (SM). Když je zjištěna chyba nebo změna stavu, modul na to reaguje aktualizací míst v paměti SM odpovídajících umístění modulu. Pokud je to první modul, aktualizuje SMB200 až SMB249 tak, jak je to potřeba pro hlášení stavových a chybových informací. Pokud je to druhý modul, aktualizuje SMB250 až SMB299 a tak dále. Viz tabulku 10-10.

Tabulka 10-10 Byty speciální paměti SMB200 až SMB549

Byty speciální paměti SMB200 až SMB549						
Inteligentní modul na pozici 0	**Inteligentní modul na pozici 1**	**Inteligentní modul na pozici 2**	**Inteligentní modul na pozici 3**	**Inteligentní modul na pozici4**	**Inteligentní modul na pozici 5**	**Inteligentní modul na pozici 6**
SMB200 až SMB249	SMB250 až SMB299	SMB300 až SMB349	SMB350 až SMB399	SMB400 až SMB449	SMB450 až SMB499	SMB500 až SMB549

Tabulka 10-11 ukazuje datovou oblast speciální paměti přidělenou modemovému modulu. Tato oblast je definována, jako kdyby inteligentní modul byl umístěn na pozici 0 rozšiřovacího systému.

Tabulka 10-11 Byty v paměti SM pro modemový modul EM 241

Adresa SM	Popis
SMB200 až SMB215	Název modulu (16 znaků ASCII), SMB200 je první znak. ”Modem EM241”
SMB216 až SMB219	Číslo verze firmwaru (4 znaky ASCII), SMB216 je první znak.
SMW220	Kód chyb 0000 – Bez chyby 0001 – Není napájení ze strany uživatele 0002 – Porucha modemu 0003 – Chybí ID konfiguračního bloku 0004 – Konfigurační blok mimo rozsah 0005 – Chyba konfigurace 0006 – Chyba ve volbě kódu země 0007 – Telefonní číslo příliš dlouhé 0008 – Zpráva příliš dlouhá 0009 až 00FF – Rezervováno 01xx – Chyba v čísle xx pro zpětné volání 02xx – Chyba v čísle xx pro pager 02xx – Chyba čísla zprávy xx 0400 až FFFF – Rezervováno
SMB222	Stav modulu – odráží stav LED  F – EM_FAULT 0 – bez poruchy 1 – porucha G – EM_GOOD 0 – není v pořádku 1 – v pořádku H – OFF_HOOK 0 – sluchátko zavěšené 1 – sluchátko zvednuté T – NO DIALTONE 0 – oznamovací tón 1 – není oznamovací tón R – RING 0 – nevyzvání 1 – telefon vyzvání C – CONNECT 0 – nespojen 1 – spojen
SMB223	Kód země, jak je nastaven přepínači (desítková hodnota).
SMW224	Přenosová rychlost stanovená pro spojení (desítková hodnota bez znaménka).
SMB226	Výsledek příkazu uživatele  D – Bit ”Done”; 0 – operace probíhá 1 – operace ukončena ERROR: Popis kódu chyby, viz tabulku 10-8.
SMB227	Volič telefonního čísla – Tento byte specifikuje, které telefonní číslo pro přenos zpráv se má při odeslání zprávy použít. Platné hodnoty jsou 1 až 250.
SMB228	Volič zprávy – Tento byte specifikuje, kterou zprávu odeslat. Platné hodnoty jsou 1 až 250.
SMB229 až SMB244	Rezervováno
SMB245	Offset prvního bytu Q použitého jako příkazové rozhraní tohoto modulu. Offset dodává CPU pro pohodlí uživatele a modul ho nepotřebuje.
SMD246	Ukazatel na tabulku konfigurace modulu v paměti V. Hodnota ukazatele na oblast jinou než paměť V není přijata a modul nadále kontroluje toto místo a čeká na jinou než nulovou hodnotu ukazatele.

Témata pro pokročilé

Vysvětlení tabulky konfigurace

Průvodce byl vyvinut, aby usnadnil řešení automatizační aplikace tím, že automaticky generuje tabulku konfigurace na základě odpovědí, které předáváte o svém systému. Informace o tabulce konfigurace uvádíme pro pokročilé uživatele, kteří si pro modemový modul chtějí vytvořit své vlastní řídicí podprogramy a sami formátovat zprávy.

Tabulka konfigurace je umístěna v paměti V programovatelného automatu S7-200. V tabulce 10-12 uvádí sloupec ”Offset bytu” offset bytu od místa, na které ukazuje ukazatel konfigurační oblasti v paměti SM. Informace tabulky konfigurace jsou rozděleny do čtyř částí.

- Konfigurační blok obsahuje informace pro konfiguraci modulu.
- Blok telefonních čísel pro zpětné volání obsahuje předdefinovaná telefonní čísla, která jsou přípustná při zabezpečení zpětným voláním.
- Blok telefonních čísel pro přenos zpráv obsahuje telefonní čísla používaná při vytáčení služeb přenosu zpráv nebo při přenosu dat CPU.
- Blok zpráv obsahuje předdefinované zprávy pro odesílání.

Tabulka 10-12 Tabulka konfigurace modemového modulu

Konfigurační blok	
Posunutí bytu	**Popis**
0 až 4	Identifikace modulu – pět znaků ASCII použitých pro přiřazení tabulky konfigurace inteligentnímu modulu. Verze 1.00 modemového modulu EM 241 očekává ”M241A”.
5	Délka konfiguračního bloku – zde 24.
6	Délka telefonního čísla pro zpětné volání – platné hodnoty jsou 0 až 40.
7	Délka telefonního čísla pro přenos zpráv – platné hodnoty jsou 0 až 120.
8	Počet telefonních čísel pro zpětné volání – platné hodnoty jsou 0 až 250.
9	Počet telefonních čísel pro přenos zpráv – platné hodnoty jsou 0 až 250.
10	Počet zpráv – platné hodnoty jsou 0 až 250.
11 až 12	Rezervováno (2 byty)
13	Tento byte obsahuje povolovací bity pro podporované funkce. MSB 7 6 5 4 3 2 1 LSB 0 PD | CB | PW | MB | BD | 0 | 0 | 0 PD – 0 = tónová volba; 1 = pulzní volba CB – 0 = zpětné volání blokováno; 1 = zpětné volání povoleno PW – 0 = heslo blokováno; 1 = heslo povoleno MB – 0 = povolen protokol PPI; 1 = protokol Modbus BD – 0 = vytáčení bez oznam. tónu blokováno; 1 = vytáčení bez oznam. tónu povoleno Bity 2, 1 a 0 jsou modulem ignorovány
14	Rezervováno
15	Počet pokusů – tato hodnota specifikuje, kolikrát se má modem pokusit o vytočení čísla a odeslání zprávy dříve, než ohlásí chybu. Hodnota 0 zabraňuje modemu ve vytáčení.
16 až 23	Heslo – osm znaků ASCII

Tabulka 10-12 Tabulka konfigurace modemového modulu, pokračování

Blok telefonních čísel pro zpětné volání (volitelný)	
Posunutí bytu	**Popis**
24	Telefonní číslo pro zpětné volání 1 – řetězec představující první telefonní číslo, které má oprávnění pro přístup ke zpětnému volání z modemového modulu EM 241. Každému telefonnímu číslu pro zpětné volání musí být přidělen takový prostor, jaký specifikuje pole pro délku telefonního čísla pro zpětné volání (offset 6 v konfiguračním bloku).
24+ číslo pro zpětné volání	Telefonní číslo pro zpětné volání 2
:	:
:	Telefonní číslo pro zpětné volání n
Blok telefonních čísel pro přenos zpráv (volitelný)	
Posunutí bytu	**Popis**
M	Telefonní číslo pro přenos zpráv 1 – řetězec představující telefonní číslo pro přenos zpráv, které obsahuje volby protokolu a tónové pulzní volby. Každému telefonnímu číslu musí být přidělena taková oblast paměti, jaká specifikuje pole pro délku telefonního čísla zpráv (offset 7 v konfiguračním bloku). Následuje popis formátu telefonního čísla pro přenos zpráv
M + délka čísla pro přenos zpráv	Telefonní číslo pro přenos zpráv 2
:	:
:	Telefonní číslo pro přenos zpráv n
Blok zpráv (volitelný)	
Posunutí bytu	**Popis**
N	Offset paměti (vzhledem k VB0) pro první zprávu (2 byty)
N+2	Délka zprávy 1
N+3	Délka zprávy 2
:	
:	Délka zprávy n
P	Zpráva 1 – řetězec (max. 120 bytů) představující první zprávu. Tento řetězec obsahuje specifikaci textu a vložené proměnné nebo může zadat přenos dat CPU. Viz formát textové zprávy a formát přenosu dat CPU níže.
P + délka zprávy 1	Zpráva 2
:	:
:	Zpráva n

Modul znovu načítá tabulku konfigurace, jestliže dojde k následujícím událostem:

- ❏ Do pěti vteřin po každém přechodu z režimu STOP do režimu RUN CPU S7-200 (pokud modem právě nepracuje online)
- ❏ Každých pět vteřin, dokud není nalezena platná konfigurace (pokud modem právě nepracuje online).
- ❏ Pokaždé, když modem přejde z práce online do práce offline.

Formát telefonního čísla pro přenos zpráv

Telefonní číslo pro přenos zpráv je struktura obsahující informace, které modemový modul potřebuje pro odeslání zprávy. Telefonní číslo pro přenos zpráv je řetězec ASCII znaků, v němž je na prvním místě byte obsahující délku a za ním následují ASCII znaky. Maximální délka telefonního čísla pro přenos zpráv je 120 bytů (včetně bytu délky).

Telefonní číslo pro přenos zpráv obsahuje až 6 polí oddělených lomítkem (/). Dvě lomítka vedle sebe označují prázdné (nulové) pole. V modemovém modulu jsou nulová pole nastavena na implicitní hodnoty.

Formát: <Telefonní číslo>/<ID>/<Heslo/<Protokol>/<Standard>/<Formát>

Pole ”Telefonní číslo” je telefonní číslo, které modemový modul vytáčí, když odesílá zprávu. Pokud je odesílanou zprávou textová zpráva pro pager nebo SMS zpráva, jde o telefonní číslo operátora. Pokud je zprávou numerická zpráva pro pager, obsahuje toto pole telefonní číslo pageru. Pokud je zpráva přenos dat CPU, jde o telefonní číslo vzdáleného zařízení. Maximální počet znaků v tomto poli je 40.

ID je číslo pageru nebo číslo mobilního telefonu. Toto pole může obsahovat pouze číslice od 0 do 9. Pokud je protokol nastaven na přenos dat CPU, použije se toto pole pro zadání adresy vzdáleného zařízení. V tomto poli je povoleno až 20 znaků.

Pole ”Heslo” se používá pro zadání hesla u zpráv odesílaných prostřednictvím TAP v případě, že heslo požaduje operátor. U zpráv odesílaných prostřednictvím UCP se toto pole používá jako adresa nebo telefonní číslo odesílatele. Pokud zpráva znamená přenos dat CPU do jiného modemového modulu, může být toto pole použito pro zadání hesla pro vzdálený modul. Heslo může mít délku až 15 znaků.

Pole ”Protokol” obsahuje jeden znak ASCII, který říká modemovému modulu, jak má formátovat a vysílat zprávu. Přípustné jsou následující hodnoty:

1 – Numerický protokol pro pager (implicitně)
2 – TAP
3 – UCP příkaz 1
4 – UCP příkaz 30
5 – UCP příkaz 51
6 – přesun dat CPU

Pole ”Standard” nutí modemový modul používat konkrétní standard pro modem. Pole ”Standard” obsahuje jeden ASCII znak . Přípustné jsou následující hodnoty:

1 – Bell 103
2 – Bell 212
3 – V.21
4 – V.22
5 – bit V.22
6 – V.23c
7 – V.32
8 – bit V.32
9 – V.34 (implicitní)

Pole ”Formát” obsahuje tři znaky ASCII, které specifikují počet datových bitů a paritu použité při odesílání zprávy. Toto pole neplatí, pokud je protokol nastaven na numerické zprávy na pager. Přípustná jsou pouze dvě nastavení:

8N1 – 8 datových bitů, žádná parita, jeden stop bit (implicitní)
7E1 – 7 datových bitů, sudá parita, jeden stop bit

Formát textové zprávy

Formát textové zprávy definuje formát textových zpráv na pager nebo SMS. Tyto zprávy mohou obsahovat text a vložené proměnné. Textová zpráva je řetězec ASCII znaků, v němž je na prvním místě byte obsahující délku a za ním následují další ASCII znaky. Maximální délka textové zprávy je 120 bytů (včetně bytu délky).

Formát: <Text><Proměnná><Text><Proměnná>...

Pole "Text" se skládá z ASCII znaků.

Pole "Proměnná" definuje hodnotu vložených dat, které modemový modul čte z lokálního CPU, formátuje a vkládá do zprávy. Znak procenta (%) se používá pro označení začátku a konce pole proměnné. Adresa a pole "Vlevo" jsou odděleny dvojtečkou. Oddělovač mezi poli vlevo a vpravo může být buď tečka, nebo čárka; používá se jako oddělení desetinných míst ve formátované proměnné. Syntaxe pro pole proměnné je:

%Adresa:Formát vlevo.Vpravo%

Pole "Adresa" specifikuje adresu, typ dat a velikost vložených dat (napo. VD100, VW50, MB20 nebo T10). Přípustné jsou následující typy dat: I, Q, M, SM, V, T (pouze word), C (pouze word) a AI (pouze word). Přípustné jsou hodnoty byte, word a double word.

Pole "Vlevo" definuje počet číslic, které se mají zobrazit nalevo od desetinné čárky. Tato hodnota musí být dostatečně velká, aby obsáhla předpokládaný rozsah vložené proměnné včetně případného znaménka minus. Pokud je pole "Vlevo" nulové, je hodnota zobrazena s nulou nalevo od desetinné čárky. Platný rozsah pro "Vlevo" je od 0 do 10.

Pole "Vpravo" definuje počet číslic, které se mají zobrazit napravo od desetinné čárky. Nuly napravo od desetinné čárky se vždy zobrazují. Pokud je pole "Vpravo" nulové, je číslo zobrazeno bez desetinné čárky. Platný rozsah pro "Vpravo" je od 0 do 10.

Pole "Formát" specifikuje formát zobrazení vložené proměnné. V poli "Formát" jsou přípustné následující znaky:

i – integer se znaménkem
u – integer bez znaménka
h – hexadecimální
f – plovoucí desetinná čárka/reálné

Příklad: "Teplota = %VW100:3.1i% Tlak = %VD200:4.3f%"

Formát zprávy pro přenos dat CPU

Přenos dat CPU, ať už z CPU do CPU nebo z CPU do Modbus, se specifikuje pomocí formátu zprávy. Zpráva s přenosem dat CPU je řetězec ASCII znaků, který může specifikovat libovolný počet přesunů dat mezi zařízeními; tento počet je omezen maximální délkou zprávy 120 bytů (119 znaků plus byte délky zprávy). Pro oddělení specifikací přenosu dat může být použit znak mezery ASCII, ale není vyžadován. Všechny specifikace přesunu dat jsou provedeny v rámci jednoho spojení. Přenosy dat jsou provedeny v pořadí definovaném ve zprávě. Pokud je v přenosu dat zjištěna chyba, je spojení se vzdáleným zařízením ukončeno a následující transakce nejsou zpracovány.

Pokud je operace specifikována jako čtení, je počet wordů "Count" načten ze vzdáleného zařízení počínaje Remote_address, a pak zapsán do paměti V lokální CPU počínaje Local_address.

Pokud je operace specifikována jako zapsání, je počet wordů "Count" načten z lokálního CPU počínaje Local_address, a pak zapsán do vzdáleného zařízení počínaje Remote_address.

Formát: <Operace>=<Count>,<Local_address>,<Remote_address>

Pole "Operace" se skládá z jednoho ASCII znaku a definuje typ přenosu.

R – Načíst data ze vzdáleného zařízení
W – Zapsat data do vzdáleného zařízení

Pole "Count" specifikuje počet hodnot wordů, které mají být přesunuty. Platný rozsah pro pole "Count" je 1 až 100 wordů.

Pole Local_address specifikuje adresu pro přenos dat v paměti V lokální CPU (napo. VW100).

Pole Remote_address specifikuje adresu pro přesun dat ve vzdáleném zařízení (napo. VW500). Tato adresa se vždy specifikuje jako adresa v paměti V, dokonce i když přesun dat probíhá do zařízení Modbus. Pokud je vzdálené zařízení Modbus, konverze mezi adresou v paměti V a adresou Modbus vypadá takto:

Adresa Modbus = 1 + (adresa v paměti V / 2)
Adresa v paměti V = (adresa Modbus – 1) * 2

Příklad: R=20,VW100, VW200 W=50,VW500,VW1000 R=100,VW1000,VW2000

11 Použití knihovny USS protokolu pro řízení pohonu MicroMaster

Knihovny instrukcí STEP 7–Micro/WIN usnadňují řízení pohonu MicroMaster, protože obsahují již nakonfigurované podprogramy přerušení a jiné podprogramy, které jsou speciálně navrženy pro použití USS protokolu při komunikaci s pohonem. Pomocí instrukcí USS můžete řídit fyzický pohon a číst, popř. zapisovat jeho parametry.

Tyto instrukce najdete ve složce ”Knihovny” instrukčního stromu STEP 7–Micro/WIN. Když vyberete nějakou instrukci USS, automaticky se k projektu přidá jeden nebo více příslušných podprogramů (USS1 až USS7).

Knihovny Siemens se prodávají na samostatném CD STEP 7–Micro/WIN Add-On: Knihovna instrukcí, objednací číslo 6ES7 830–2BC00–0YX0. Po zakoupení a instalaci verze 1.1 knihovny Siemens bude každý následující nainstalovaný upgrade programovacího balíku STEP 7–Micro/WIN V3.2x a V4.0 také automaticky aktualizovat knihovny bez dalších poplatků (po doplnění a modifikaci knihovny).

Přehled kapitoly

Požadavky použití USS protokolu 332
Výpočet doby potřebné pro komunikaci s pohonem 332
Použití instrukcí USS 333
Instrukce USS protokolu 334
Vzorové programy USS protokolu 341
Chybové kódy pro provádění USS 342
Připojení a nastavení pohonu MicroMaster řady 3 342
Připojení a nastavení pohonu MicroMaster řady 4 345

Požadavky použití USS protokolu

Knihovny instrukcí ve STEP 7–Micro/WIN obsahují 14 podprogramů, 3 podprogramy přerušení a 8 instrukcí, které podporují USS protokol. Instrukce USS používají v automatu S7-200 následující prostředky:

- Inicializace USS protokolu vyhradí port 0 pro komunikace USS.

  Instrukce USS_INIT se používá pro volbu USS nebo PPI pro port 0. (USS se vztahuje na USS protokol pro pohony SIMOTION MicroMaster.) Poté, co zvolíte pro komunikaci s pohony použití USS protokolu, nemůžete port 0 používat pro žádný jiný účel včetně komunikace se STEP 7–Micro/WIN.

  Během programování aplikace používající USS protokol byste měli používat CPU 224XP, CPU 226 nebo modul EM 277 PROFIBUS–DP s připojením na kartu PROFIBUS CP. Tento druhý komunikační port umožňuje STEP 7–Micro/WIN monitorovat aplikaci během chodu USS protokolu.
- Instrukce USS ovlivňují všechna místa paměti SM spojená s komunikací Freeport na portu 0.
- Instrukce USS používají 14 podprogramů a 3 podprogramy přerušení.
- Díky instrukcím USS se zvětšuje část paměti potřebná pro uživatelský program až o 3600 bytů. V závislosti na konkrétních použitých instrukcích USS mohou podpůrné podprogramy pro tyto instrukce zvýšit režii pro řídicí program nejméně o 2300 bytů, až na 3600 bytů.
- Proměnné pro instrukce USS potřebují 400bytový blok v paměti V. Počáteční adresa tohoto bloku je přidělena uživatelem a je rezervována pro proměnné USS.
- Některé z instrukcí USS také vyžadují 16bytový komunikační zásobník. Uživatel přiděluje počáteční adresu v paměti V pro tento zásobník jako parametr instrukce. Doporučuje se přiřadit vlastní zásobník každé instrukci USS.
- Při provádění výpočtů používají instrukce USS akumulátory AC0 až AC3. Ve vašem programu můžete také použít akumulátory, ale hodnoty v akumulátorech budou instrukcemi USS změněny.
- Instrukce USS není možné používat v podprogramu přerušení.



Tip

Chcete–li změnit činnost portu 0 zpět na PPI, abyste mohli komunikovat se STEP 7–Micro/WIN, použijte další instrukci USS_INIT, která změní přiřazení portu 0.

Můžete také přepnout přepínač režimu na S7-200 na režim STOP. Tím se resetují parametry portu 0. Uvědomte si ale, že zastavení komunikace s pohony zastaví i pohony samotné.

Výpočet doby potřebné pro komunikaci s pohonem

Komunikace s pohonem je asynchronní s programovým cyklem automatu S7-200. Běžně S7-200 dokončí několik programových cyklů dříve, než je ukončena jedna komunikační transakce. Následující faktory pomáhají při stanovení potřebné délky času: počet připojených pohonů, přenosová rychlost a doba programového cyklu S7-200.

Některé pohony potřebují při použití instrukcí pro přístup k parametrům delší čas. Čas potřebný pro přístup k parametrům závisí na typu pohonu a parametru, ke kterému se přistupuje.

Poté, co instrukce USS_INIT přikáže portu 0 používat USS protokol, provádí S7-200 pravidelné dotazování všech aktivních pohonů v intervalech uvedených v tabulce 11-1. Pro každý pohon musíte nastavit parametr časové prodlevy s přihlédnutím k této skutečnosti.

Tabulka 11-1 Časy komunikace

Přenosová rychlost	Doba mezi dotazováním aktivních pohonů (když není aktivní žádná instrukce pro přístup k parametrům)
1200	240 25 ms (maximálně) krát počet pohonů
2400	130 25 ms (maximálně) krát počet pohonů
4800	75 25 ms (maximálně) krát počet pohonů
9600	50 25 ms (maximálně) krát počet pohonů
19200	35 25 ms (maximálně) krát počet pohonů
38400	30 25 ms (maximálně) krát počet pohonů
57600	25 25 ms (maximálně) krát počet pohonů
115200	25 25 ms (maximálně) krát počet pohonů



Tip

V jednu dobu může být aktivní pouze jedna instrukce USS_RPM_x nebo USS_WPM_x. Výstup "Done" (Dokončeno) každé instrukce musí signalizovat ukončení dříve, než uživatelský program spustí novou instrukci.

Pro každý pohon použijte pouze jednu instrukci USS_CTRL.

Použití instrukcí USS

Pro použití instrukcí USS protokolu v programu programovatelného automatu S7-200 proveďte tyto kroky:

1. Vložte do svého programu instrukci USS_INIT a provádějte ji pouze po dobu jednoho programového cyklu. Instrukci USS_INIT můžete použít buď pro inicializaci, nebo pro změnění parametru komunikace USS.

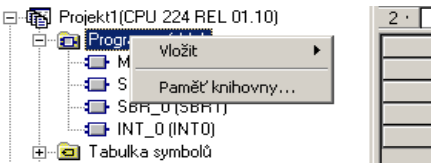
   Když vložíte instrukci USS_INIT, přidá se k vašemu programu automaticky několik skrytých podprogramů a podprogramů přerušení.

2. Pro každý aktivní pohon vložte do svého programu pouze jednu instrukci USS_CTRL.

   Instrukcí USS_RPM_x a USS_WPM_x můžete vložit tolik, kolik potřebujete, ale pouze jedna z nich může být v určitou dobu aktivní.

3. Paměť V pro knihovnu instrukcí přidělíte tak, že kliknete pravým tlačítkem myši (abyste rozbalili menu) na uzel programového bloku v instrukčním stromě.

   Aby se zobrazilo dialogové okno "Přidělení paměti pro knihovnu", vyberte "Paměť knihovny".



Obr. 11-1 Přidělení paměti V knihovně instrukcí

4. Nakonfigurujte parametry pohonu tak, aby odpovídaly přenosové rychlosti a adrese použité programem.

5. Zapojte komunikační kabel mezi automatem S7-200 a pohony.

   Zajistěte, aby všechna řídicí zařízení, jako je například S7-200, která jsou propojena s pohonem, byla spojena krátkým silným kabelem se stejným uzemněním nebo nulovým bodem jako pohon.

Upozornění

Jestliže propojíte zařízení s různými referenčními potenciály, mohou spojovacím kabelem protékat nežádoucí proudy. Tyto proudy mohou způsobit chyby v komunikaci nebo poškodit zařízení.

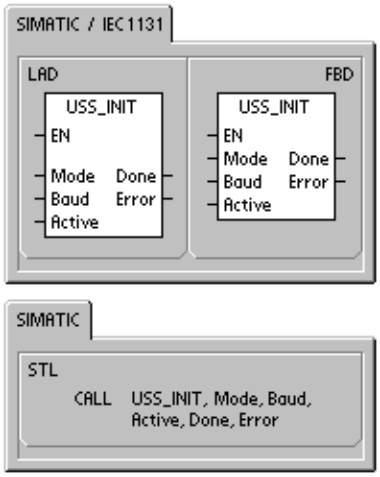
Zkontrolujte si, že všechna zařízení, která jsou propojena komunikačním kabelem, buďto mají společný referenční potenciál, nebo jsou napěťově izolována, aby se zabránilo průtoku nežádoucích proudů.

Stínění musí být zapojeno na uzemnění rozvaděče nebo k pinu 1v 9pinovém konektoru. Doporučuje se připojit svorku 2 – 0V na pohonu MicroMaster k uzemnění rozvaděče.

Instrukce USS protokolu

Instrukce USS_INIT

Instrukce USS_INIT se používá pro povolení a inicializaci nebo pro ukončení komunikace s pohony MicroMaster. Instrukce USS_INIT musí být dokončena bez chyb před použitím jakékoliv jiné instrukce USS. Dříve než může být provedena další instrukce, musí být prováděná instrukce ukončena a nastaven bit ”Done”.



Tato instrukce je provedena v každém programovém cyklu, když je zapnutý vstup EN.

Instrukci USS_INIT proveďte pro každou změnu ve stavu komunikace pouze jednou. Proto by vstup EN měl být zapojený pulzně, přes vyhodnocující náběžnou hranu. Chcete–li změnit parametry inicializace, proveďte novou instrukci USS_INIT.

Parametr ”Mode” volí komunikační protokol: vstupní hodnota 1 přiřazuje port 0 USS protokolu a aktivuje protokol, vstupní hodnota 0 přiřazuje port 0 protokolu PPI a blokuje USS protokol.

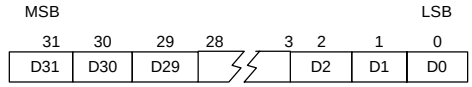
”Baud” nastavuje přenosovou rychlost na 1200, 2400, 4800, 9600, 19200, 38400, 57600 nebo 115200. Přenosové rychlosti 57600 a 115200 jsou podporovány S7-200 CPU verze 1.2 nebo vyšší.

”Active” indikuje, které pohony jsou aktivní. Některé pohony podporují pouze adresy 0 až 30.

Tabulka 11-2 Parametry instrukce USS_INIT

Vstupy/Výstupy	Typ dat	Operandy
Mode	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, konstanta, *VD, *AC, *LD
Baud, Active	DWORD	VD, ID, QD, MD, SD, SMD, LD, konstanta, AC *VD, *AC, *LD
Done	BOOL	I, Q, M, S, SM, T, C, V, L
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD

Obrázek 11-2 ukazuje popis a formát vstupu aktivního pohonu. Kterýkoliv pohon označený jako ”Active” je automaticky dotazován na pozadí tak, aby řídil pohon, shromažďoval stav a zabránil časovému odpojení sériové linky pohonu.



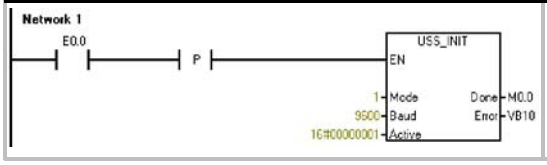
D0 Bit aktivity pro pohon 0; 0 – pohon není aktivní, 1 – pohon aktivní
D1 Bit aktivity pro pohon 1; 0 – pohon není aktivní, 1 – pohon aktivní
...

Obr. 11-2 Formát parametru aktivního pohonu

Pokud chcete vypočítat časový interval mezi dotazy na stav, nahlédněte do tabulky 11-1.

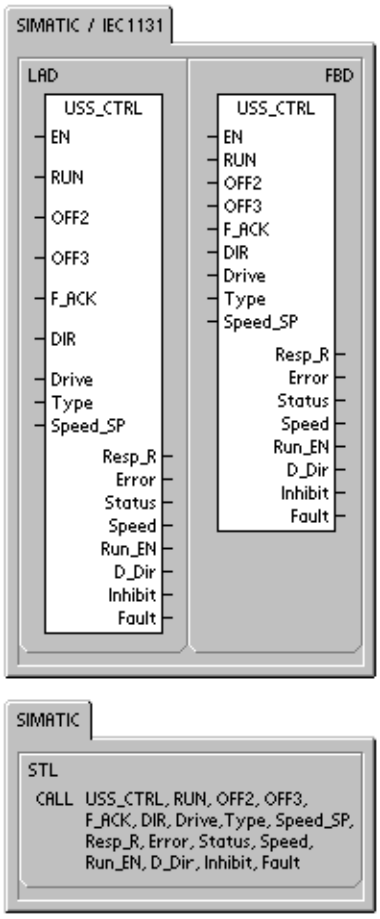
Když je dokončena instrukce USS_INIT, zapne se výstup ”Done”. Výstupní byte ”Error” obsahuje výsledek provádění této instrukce. Tabulka 11-6 definuje chyby, které mohou vyplynout z provedení této instrukce.

Příklad: Podprogram USS_INIT




```
Network 1
LD      I0.0
EU
CALL    USS_INIT, 1, 9600, 16#00000001,
        M0.0, VB10
```

Instrukce USS_CTRL



Instrukce USS_CTRL se používá na řízení aktivního pohonu MicroMaster. Tato instrukce uloží vybrané příkazy do komunikačního zásobníku, který je pak odeslán adresovanému pohonu (parametr ”Drive” – pohon), pokud byl tento pohon vybrán v parametru ”Active” instrukce USS_INIT.

Každému pohonu by měla být přiřazena pouze jedna instrukce USS_CTRL.

Některé pohony hlásí rychlost pouze jako kladnou hodnotu. Pokud je rychlost záporná, hlásí ji pohon jako kladnou, ale invertuje se bit D_Dir (směr).

Aby byla povolena instrukce USS_CTRL, musí být zapnutý bit EN. Tato instrukce by měla být vždy povolena.

RUN (RUN/STOP) indikuje, zda je pohon zapnutý (1), nebo vypnutý (0). Když je bit RUN zapnutý, dostane pohon MicroMaster příkaz, aby se rozběhl se zadanou rychlostí a směrem. Aby byl pohon uveden do chodu, musí platit následující:

- Pohon musí být vybrán jako ”Active” v USS_INIT.
- OFF2 a OFF3 musí být nastaveny na 0.
- ”Fault” (Porucha) a ”Inhibit” (Blokování) musí být 0.

Když je bit RUN vypnutý, je pohonu MicroMaster poslán příkaz, aby zastavil se zpomalením. Bit OFF2 se používá, aby umožnil pohonu MicroMaster volně doběhnout až do zastavení. Bit OFF3 se používá, jestliže chceme pohonu MicroMaster přikázat, aby zastavil okamžitě.

Bit Resp_R (Odpověď přijata) potvrzuje odpověď pohonu. Všechny aktivní pohony jsou cyklicky dotazovány na aktuální informace o stavu pohonu. Pokaždé, když automat S7-200 dostane odpověď od pohonu, zapne se bit Resp_R po dobu jednoho programového cyklu a všechny hodnoty jsou aktualizovány.

Bit F_ACK (Potvrzení poruchy) se používá pro potvrzení poruchy pohonu. Pohon vymaže poruchu (Fault), když F_ACK přejde z 0 na 1.

Bit DIR (Směr) ukazuje, kterým směrem by se měl pohon pohybovat.

Tabulka 11-3 Parametry instrukce USS_CTRL

Vstupy/Výstupy	Typy dat	Operandy
RUN, OFF 2, OFF 3, F_ACK, DIR	BOOL	I, Q, M, S, SM, T, C, V, L, signálový tok
Resp_R, Run_EN, D_Dir, Inhibit, Fault	BOOL	I, Q, M, S, SM, T, C, V, L
Drive, Type	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD, konstanta
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD
Status	WORD	VW, T, C, IW, QW, SW, MW, SMW, LW, AC, AQW, *VD, *AC, *LD
Speed_SP	REAL	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD, konstanta
Speed	REAL	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD

Vstup ”Drive” (Adresa pohonu) je adresa pohonu MicroMaster, na kterou má být poslán příkaz USS_CTRL. Platné adresy: 0 až 31.

Vstup ”Type” (Typ pohonu) volí typ pohonu. Pro pohon MicroMaster 3 (nebo předchozí verzi) nastavte ”Type” na 0. Pro pohon MicroMaster 4 nastavte ”Type” na 1.

“Speed_SP” (Nastavená hodnota rychlosti) je rychlost motoru jako procento plných otáček. Záporné hodnoty Speed_SP způsobí, že se pohon bude pohybovat v opačném směru. Dosah: –200,0 % až 200,0 %.

”Error” (Chyba) je chybový byte, který obsahuje výsledek posledního požadavku na komunikaci s modulem. Tabulka 11-6 definuje chyby, které mohou vyplynout z provedení této instrukce.

”Status” (Stav) je nezpracovaná hodnota stavového slova, odeslaného zpět pohonem. Obrázek 11-3 ukazuje stavové bity pro ”Standard Status Word” (Standardní stavové word) a ”Main Feedback” (Hlavní zpětná vazba).

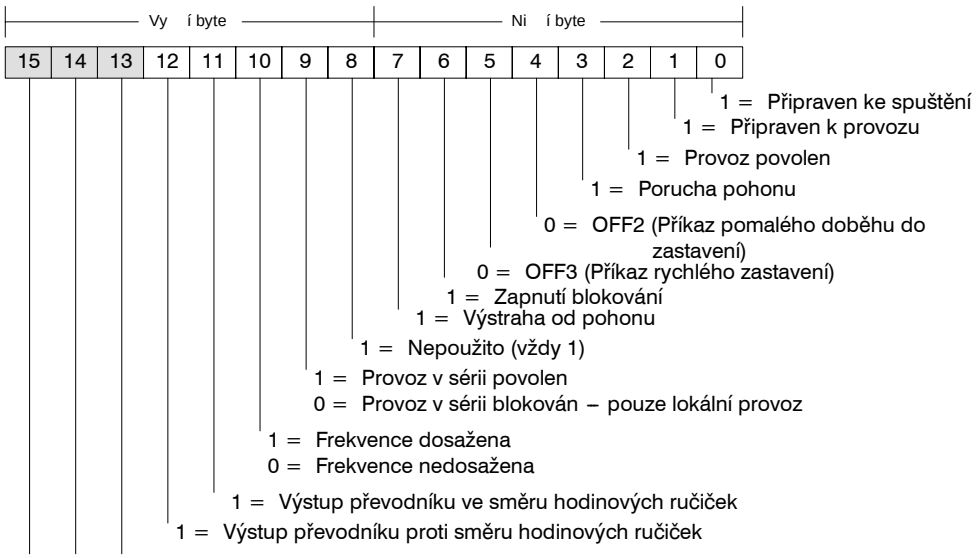
”Speed” (Rychlost) je rychlost pohonu jako procento plných otáček. Dosah: –200,0 % až 200,0 %.

RUN_EN (Povolení RUN) indikuje, zda je pohon v chodu (1), nebo vypnutý (0).

D_Dir indikuje směr pohybu pohonu.

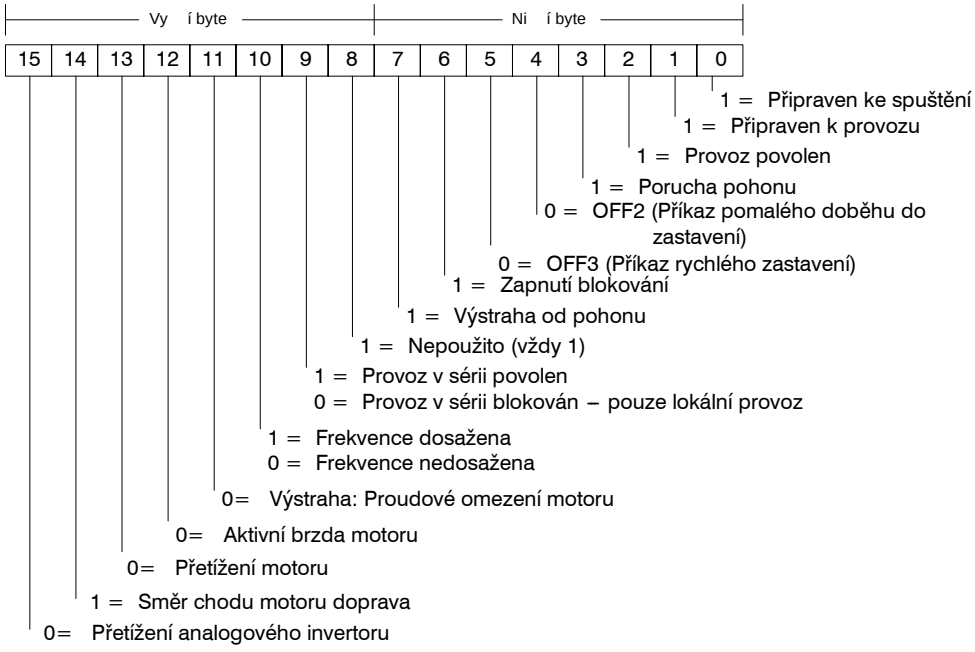
”Inhibit” indikuje stav blokovacího bitu pro pohon (0 – není blokován, 1 – blokován). Chcete–li vymazat blokovací bit, musí být vypnutý bit ”Fault”; vstupy RUN, OFF2 a OFF3 musí být také vypnuté.

”Fault” indikuje stav bitu pro poruchu (0 – bez poruchy, 1 – porucha). Poruchový kód odráží stav pohonu. (Viz příručku pro pohon.) Abyste vymazali bit ”Fault”, opravte příčinu poruchy a nastavte bit F_ACK.



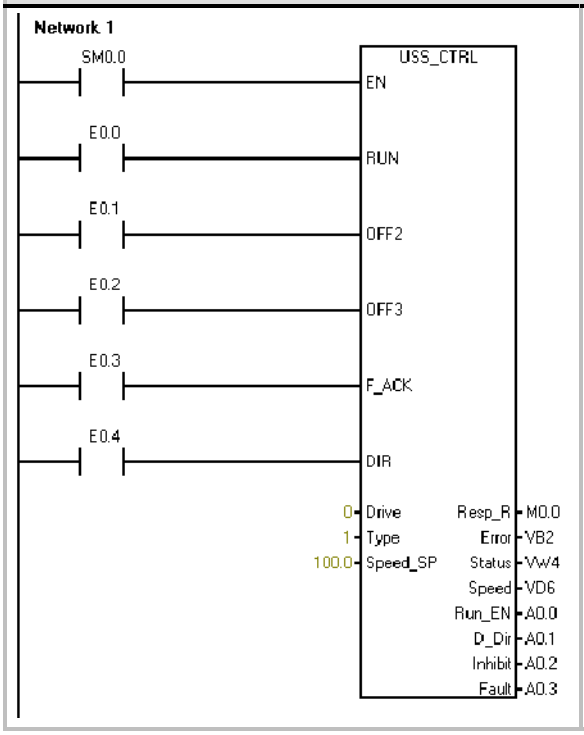
Rezervováno pro použití v budoucnosti: Tyto bity nemusí vždy být nulové.

Obr. 11-3 Stavové bity pro standardní stavové word pro MicroMaster 3 a hlavní zpětná vazba



Obr. 11-4 Stavové bity pro standardní stavové word pro MicroMaster 4 a hlavní zpětná vazba

Příklad: Podprogram USS_CTRL



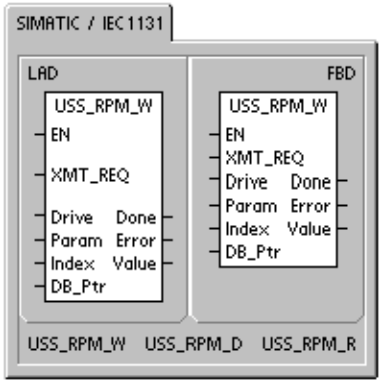
Pro zobrazení pouze v STL:

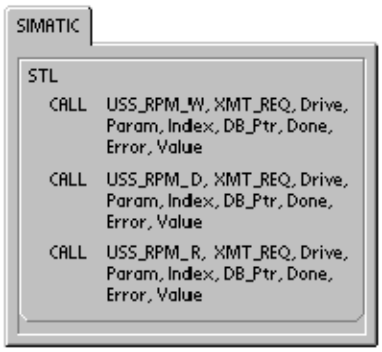
```
Network 1       //Řídicí blok pro pohon 0
LD      SM0.0
CALL    USS_CTRL, I0.0, I0.1, I0.2, I0.3,
        I0.4, 0, 1, 100.0, M0.0, VB2, VW4,
        VD6, Q0.0, Q0.1, Q0.2, Q0.3
```

Pro zobrazení v LAD nebo FBD:

```
Network 1       //Řídicí blok pro pohon 0
LD      SM0.0
=       L60.0
LD      I0.0
=       L63.7
LD      I0.1
=       L63.6
LD      I0.2
=       L63.5
LD      I0.3
=       L63.4
LD      I0.4
=       L63.3
LD      L60.0
CALL    USS_CTRL, L63.7, L63.6, L63.5,
        L63.4, L63.3, 0, 1, 100.0, M0.0,
        VB2, VW4, VD6, Q0.0, Q0.1, Q0.2,
        Q0.3
```

Instrukce USS_RPM_x





V protokolu USS existují tři instrukce pro čtení:

- Instrukce USS_RPM_W čte parametr word bez znaménka.
- Instrukce USS_RPM_D čte parametr double word bez znaménka.
- Instrukce USS_RPM_R čte parametr s plovoucí desetinnou čárkou.

V jednu dobu může být aktivní pouze jedna instrukce pro čtení (USS_RPM_x) nebo zápis (USS_WPM_x).

Transakce USS_RPM_x je ukončena, když pohon MicroMaster potvrdí příjem příkazu nebo když je ohlášena chyba. Programový cyklus se nadále provádí i v době, kdy tento proces čeká na odezvu.

Aby byl povolen přenos požadavku, musí být bit EN zapnutý a musí zůstat zapnutý do nastavení bitu ”Done”, který signalizuje ukončení procesu. Například požadavek USS_RPM_x je vysílán pohonu MicroMaster v každém programovém cyklu, kdy je zapnutý vstup XMT_REQ. Proto by měl být vstup XMT_REQ spouštěn pulzem od náběžné hrany, aby byl požadavek vyslaný pouze jednou v každém kladném přechodu vstupu EN.

Vstup ”Drive” je adresa pohonu MicroMaster, na kterou má být poslán příkaz USS_RPM. Platné adresy jednotlivých pohonů jsou 0 až 31.

”Param” je číslo parametru. ”Index” je indexová hodnota parametru, který má být načten. ”Value” je vrácená hodnota parametru. Adresa 16bytového zásobníku musí být zadána na vstup DB_Ptr. Tento zásobník používá instrukce USS_RPM_x na uložení výsledku příkazu daného pohonu MicroMaster.

Po ukončení instrukce USS_RPM_x se zapne výstup ”Done” a výstupní byte ”Error” a výstup ”Value” obsahují výsledky provedení instrukce. Tabulka 11-6 definuje chyby, které mohou vyplynout z provedení této instrukce. Výstupy ”Error” a ”Value” nejsou platné, dokud se nezapne výstup ”Done”.

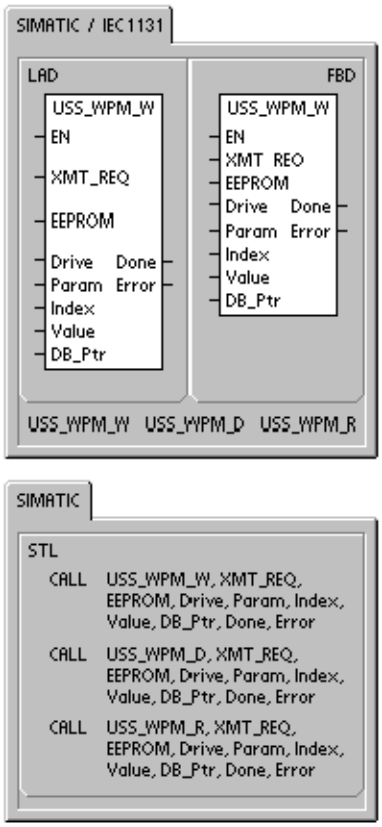
Tabulka 11-4 Operandy platné pro USS_RPM_x

Vstupy/Výstupy	Typ dat	Operandy
XMT_REQ	BOOL	I, Q, M, S, SM, T, C, V, L, signálový tok podmíněný detekcí náběžné hrany
Drive	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD, konstanta
Param, Index	WORD	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, AIW, *VD, *AC, *LD, konstanta
DB_Ptr	DWORD	&VB
Value	WORD DWORD, REAL	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, AQW, *VD, *AC, *LD VD, ID, QD, MD, SD, SMD, LD, *VD, *AC, *LD
Done	BOOL	I, Q, M, S, SM, T, C, V, L
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC. *VD, *AC, *LD

Instrukce USS_WPM_x

V protokolu USS existují tři instrukce pro zápis:

- ❏ Instrukce USS_WPM_W zapisuje parametr word bez znaménka.
- ❏ Instrukce USS_WPM_D zapisuje parametr double word bez znaménka.
- ❏ Instrukce USS_WPM_R zapisuje parametr s plovoucí desetinnou čárkou.



V jednu dobu může být aktivní pouze jedna instrukce pro čtení (USS_RPM_x) nebo zápis (USS_WPM_x).

Transakce USS_WPM_x je ukončena, když pohon MicroMaster potvrdí příjem příkazu nebo když je ohlášena chyba. Programový cyklus se nadále provádí i v době, kdy tento proces čeká na odezvu.

Aby byl povolen přenos požadavku, musí být Bit EN zapnutý a musí zůstat zapnutý do nastavení bitu "Done", který signalizuje ukončení procesu. Například požadavek USS_WPM_x je vysílán pohonu MicroMaster v každém programovém cyklu, kdy je zapnutý vstup XMT_REQ. Proto by měl být vstup XMT_REQ spouštěn pulzem od náběžné hrany, aby byl požadavek vyslaný pouze jednou v každém kladném přechodu vstupu EN.

Vstup "Drive" je adresa pohonu MicroMaster, na kterou má být poslán příkaz USS_WPM. Platné adresy jednotlivých pohonů jsou 0 až 31.

"Param" je číslo parametru. "Index" je indexová hodnota parametru, který má být zapsán. "Value" je hodnota parametru, která má být zapsána do RAM v pohonu. U pohonů MicroMaster 3 můžete tuto hodnotu také zapsat do EEPROM pohonu; záleží na tom, jak jste nakonfigurovali P971 (řízení ukládání do paměti EEPROM).

Adresa 16bytového zásobníku musí být zadána na vstup DB_Ptr. Tento zásobník instrukce USS_WPM_x používá na uložení výsledku příkazu daného pohonu MicroMaster.

Po ukončení instrukce USS_WPM_x se zapne výstup "Done" a výstupní byte "Error" obsahuje výsledky provedení instrukce. Tabulka 11-6 definuje chyby, které mohou vyplynout z provedení této instrukce.

Pokud je vstup EEPROM zapnutý, zapisuje instrukce do RAM i EEPROM pohonu. Když je vstup vypnutý, zapisuje instrukce pouze do RAM pohonu. Abyste mohli tuto instrukci s pohonem používat, musíte zajistit vypnutí tohoto vstupu, protože pohon MicroMaster 3 tuto funkci nepodporuje.

Tabulka 11-5 Operandy platné pro instrukce USS_WPM_x

Vstupy/Výstupy	Typ dat	Operandy
XMT_REQ	BOOL	I, Q, M, S, SM, T, C, V, L, signálový tok podmíněný detekcí náběžné hrany
EEPROM	BOOL	I, Q, M, S, SM, T, C, V, L, signálový tok
Drive	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD, konstanta
Param, Index	WORD	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, AIW, *VD, *AC, *LD, konstanta
DB_Ptr	DWORD	&VB
Value	WORD DWORD, REAL	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, AQW, *VD, *AC, *LD VD, ID, QD, MD, SD, SMD, LD, *VD, *AC, *LD
Done	BOOL	I, Q, M, S, SM, T, C, V, L
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC. *VD, *AC, *LD

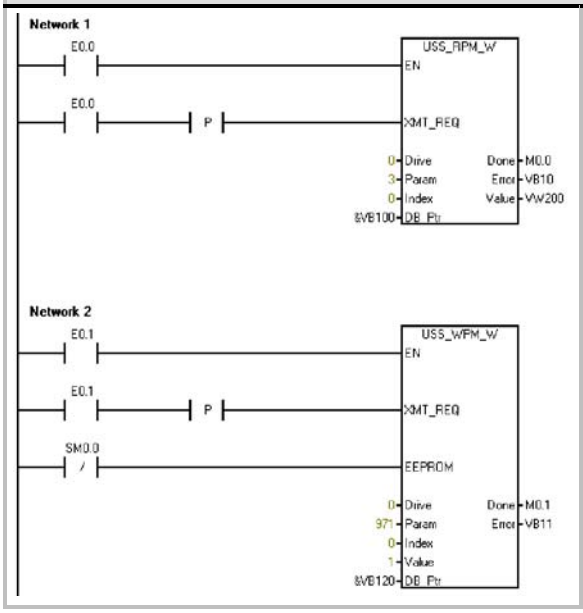
Upozornění

Pokud použijete instrukci USS_WPM_x na aktualizaci sady parametru uložených v paměti EEPROM pohonu, musíte zajistit, aby nebyl překročen maximální počet cyklů zápisu do EEPROM (přibližně 50.000).

Překročení maximálního počtu zapisovacích cyklů bude mít za následek poškození uložených dat a jejich následnou ztrátu. Počet cyklů čtení není omezen.

Pokud požadujete častý zápis do parametrů pohonů, měli byste u pohonů MicroMaster 4 nejprve nastavit řídicí parametr paměti EEPROM v pohonu na nulu (u pohonů MicroMaster 3) nebo vypnout vstup EEPROM.

Příklad: USS_RPM_x a USS_WPM_x

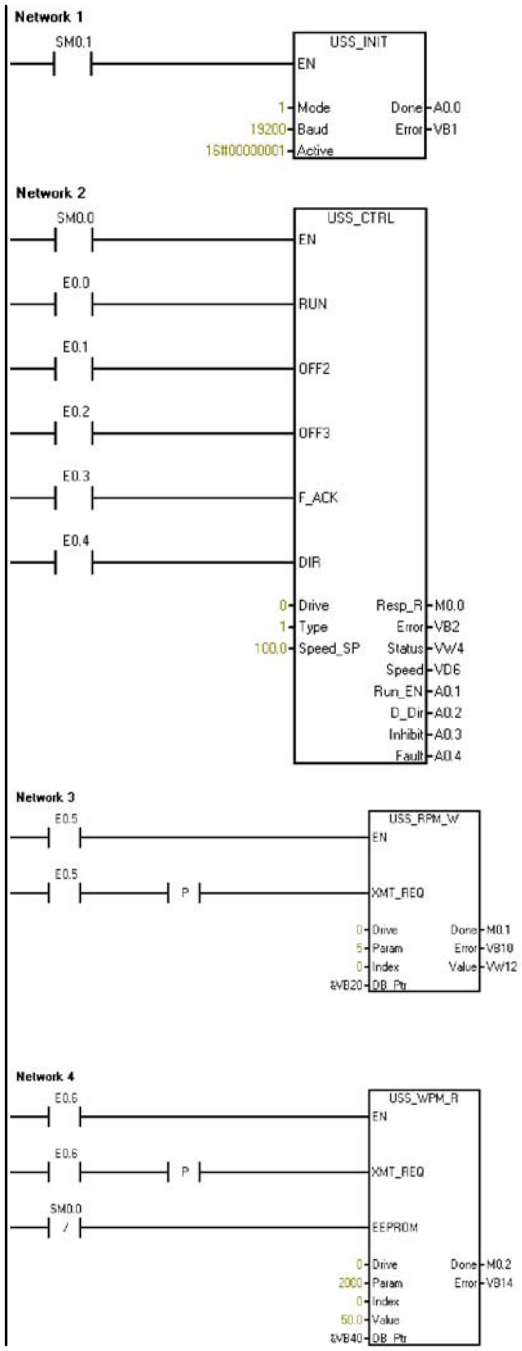



```
Network 1       //Oba kontakty musí mít stejnou adresu.
LD      I0.0
=       L60.0
LD      I0.0
EU
=       L63.7
LD      L60.0
CALL    USS_RPM_W, L63.7, 0, 3, 0, &VB100,
        M0.0, VB10, VW200

Network 2       //Oba kontakty musí mít stejnou adresu.
LD      I0.1
=       L60.0
LD      I0.1
EU
=       L63.7
LDN     SM0.0
=       L63.6
LD      L60.0
CALL    USS_WPM_W, L63.7, L63.6, 0, 971, 0, 1,
        &VB120, M0.1, VB11
```

Vzorové programy USS protokolu

Příklad: Vzorový program instrukcí USS , který se správně zobrazí v STL




```
Network 1       //Inicializuje protokol USS:
                //V prvním programovém cyklu povolí
                //USS protokol pro port 0 při 19200
                //s adresou pohonu "0" aktivní.
LD      SM0.1
CALL    USS_INIT, 1, 19200, 16#00000001, Q0.0,
        VB1

Network 2       //Řídicí parametry pro pohon 0
LD      SM0.0
CALL    USS_CTRL, I0.0, I0.1, I0.2, I0.3, I0.4, 0, 1,
        100.0, M0.0, VB2, VW4, VD6, Q0.1, Q0.2,
        Q0.3, Q0.4

Network 3       //Načte parametr "Word" z pohonu 0.
                //Načte parametr 5, index 0.
                //1. Uloží stav I0.5 na dočasné paměťové
                //   místo, takže tento network se zobrazí
                //   v LAD.
                //2. Uloží náběžnou hranu pulzu I0.5
                //   na dočasné paměťové místo L, aby
                //   mohla být předána do podprogramu.
LD      I0.5
=       L60.0
LD      I0.5
EU
=       L63.7
LD      L60.0
CALL    USS_RPM_W, L63.7, 0, 5, 0, &VB20, M0.1,
        VB10, VW12
Network 4       //Zapíše parametr "Word" do pohonu 0.
                //Zapíše parametr 2000 index 0.
LD      I0.6
=       L60.0
LD      I0.6
EU
=       L63.7
LDN     SM0.0
=       L63.6
LD      L60.0
CALL    USS_WPM_R, L63.7, L63.6, 0, 2000, 0, 50.0,
        &VB40, M0.2, VB14
```

Poznámka: Tento kód STL se nekompiluje na LAD nebo FBD.

Chybové kódy pro provádění USS

Tabulka 11-6 Chybové kódy při běhu instrukcí USS

Chybové kódy	Popis
0	Bez chyby
1	Pohon neodpověděl
2	Byla zjištěna chyba kontrolního součtu v odpovědi pohonu
3	Byla zjištěna chyba parity v odpovědi pohonu
4	Chyba způsobená zásahem uživatelského programu
5	Pokus o nepřípustný příkaz
6	Zadána nepřípustná adresa pohonu
7	Komunikační port nebyl nastaven pro USS protokol
8	Komunikační port je zaneprázdněn zpracováváním instrukce
9	Vstupní rychlost pohonu je mimo rozsah
10	Nesprávná délka odpovědi pohonu
11	Nesprávný první znak v odpovědi pohonu
12	Znak délky v odpovědi pohonu není podporován instrukcemi USS
13	Odpověděl nesprávný pohon
14	Zadaná adresa DB_Ptr je nesprávná
15	Zadané číslo parametru je nesprávné
16	Byl vybrán neplatný protokol
17	USS je aktivní; změna není povolena
18	Byla specifikována neplatná přenosová rychlost
19	Nekomunikuje se: pohon není AKTIVNÍ
20	Parametr nebo hodnota v odpovědi pohonu jsou nesprávné nebo obsahují chybový kód
21	Namísto požadované hodnoty word byla vrácena hodnota double word
22	Namísto požadované hodnoty double word byla vrácena hodnota word

Připojení a nastavení pohonu MicroMaster řady 3

Připojení pohonu MicroMaster 3

Pro spojení automatu S7-200 s pohonem MicroMaster řady 3 (MM3) můžete použít standardní kabel a konektory PROFIBUS. Správnou symetrizaci a zakončení propojovacího kabelu najdete na obrázku 11-5.

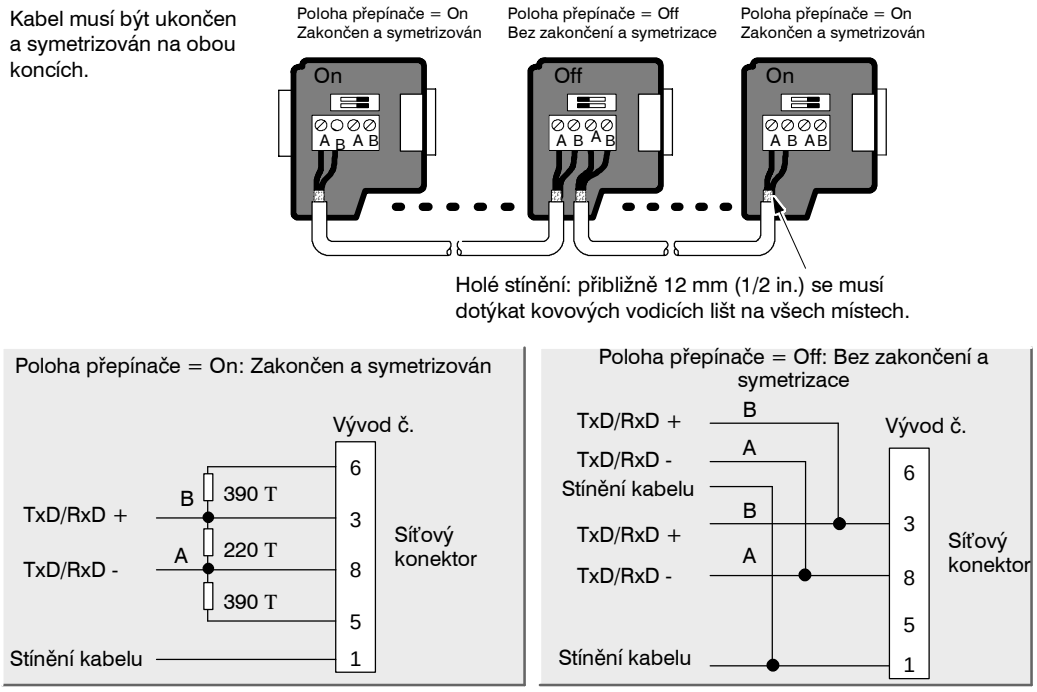
Upozornění

Jestliže propojíte zařízení s různými referenčními potenciály, mohou spojovacím kabelem protékat nežádoucí proudy.

Tyto nežádoucí proudy mohou způsobit chyby v komunikaci nebo poškodit zařízení.

Aby se zabránilo průtoku nežádoucích proudů, zkontrolujte, zda všechny části zařízení, které chcete propojit komunikačním kabelem, mají buď společný referenční potenciál, nebo jsou napěťově odděleny, aby se zabránilo průtoku nežádoucích proudů.

Stínění musí být zapojeno na uzemnění rozvaděče nebo k pinu 1 v 9pinovém konektoru. Doporučuje se připojit svorku 2 – 0V na pohonu MicroMaster k uzemnění rozvaděče.



Obr. 11-5 Symetrizace a zakončení síťového kabelu

Nastavení pohonu MicroMaster 3

Dříve než pohon připojíte k automatu S7-200, musíte zkontrolovat, zda má pohon následující systémové parametry. Pro nastavení parametrů použijte číslicovou klávesnici na pohonu:

1. Resetujte pohon na tovární nastavení. Stiskněte klávesu P: Zobrazí se P000. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí P944. Stiskněte P, abyste vložili parametr.

   P944 = 1

2. Povolte přístup ke čtení/zápisu všech parametrů. Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí P009. Stiskněte P, abyste vložili parametr.

   P009 = 3

3. Zkontrolujte nastavení motoru vašeho pohonu. Nastavení se budou měnit podle použitého motoru/použitých motorů. Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí nastavení motoru vašeho pohonu. Stiskněte P, abyste vložili parametr.

   P081 = Jmenovitá frekvence motoru (Hz)
   P082 = Jmenovitá rychlost motoru (RPM)
   P083 = Jmenovitý proud motoru (A)
   P084 = Jmenovité napětí motoru (V)
   P085 = Jmenovitý výkon motoru (kW/HP)

4. Nastavte režim řízení. Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí P910. Stiskněte P, abyste vložili parametr.

   P910 = 1 Režim řízení ”Remote” (vzdálené)

5. Nastavte přenosovou rychlost sériového rozhraní RS−485. Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se neobjeví P092. Stiskněte P, abyste vložili parametr. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí číslo, které odpovídá přenosové rychlosti sériového rozhraní RS−485. Stiskněte P, abyste je vložili.

   P092 3 (1200 baud)
   4 (2400 baud)
   5 (4800 baud)
   6 (9600 baud − implicitní nastavení)
   7 (19200 baud)

6. Vložte adresu zařízení typu slave. Každý pohon (maximálně 31) může být řízen přes sběrnici. Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se neobjeví P091. Stiskněte P, abyste vložili parametr. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí adresa zařízení typu slave. Stiskněte P, abyste je vložili.
   P091 = 0 až 31.

7. Doba rozběhu (volitelné). Je to doba v sekundách, kterou motor potřebuje na to, aby zrychlil na maximální frekvenci. Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se neobjeví P002. Stiskněte P, abyste vložili parametr. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí doba rozběhu. Stiskněte P, abyste je vložili.
   P002=0−650,00

8. Doba doběhu (volitelné). Je to doba v sekundách, kterou motor potřebuje na to, aby zpomalil až do úplného zastavení. Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se neobjeví P003. Stiskněte P, abyste vložili parametr. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí doba doběhu. Stiskněte P, abyste je vložili.
   P003=0−650,00

9. Časová prodleva sériové linky. Je to maximální povolený interval mezi dvěma příchozími datovými zprávami. Tato funkce se používá pro vypnutí invertoru v případě poruchy komunikace.

   Odpočítávání času začne od příjmu platné datové zprávy. Pokud ve specifikovaném časovém intervalu není přijata další datová zpráva, invertor se vypne a zobrazí se chybový kód F008. Nastavení hodnoty na nulu vypne řízení. Pro výpočet doby mezi dotazy na stav pohonu použijte tabulku 11-1.

   Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se neobjeví P093. Stiskněte P, abyste vložili parametr. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí časová prodleva sériové linky. Stiskněte P, abyste je vložili.
   P093 = 0−240 (0 je implicitní nastavení; čas je v sekundách)

10. Nominální systémová žádaná hodnota sériové linky. Tato hodnota se může měnit, ale běžně odpovídá 50 Hz nebo 60 Hz, což je 100% hodnota PV (procesní hodnota) nebo SP (žádaná hodnota). Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se neobjeví P094. Stiskněte P, abyste vložili parametr. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí nominální systémová žádaná hodnota sériové linky. Stiskněte P, abyste je vložili.

    P094 = 0−400,00

11. Kompatibilita s USS (volitelné). Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se neobjeví P095. Stiskněte P, abyste vložili parametr. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí číslo odpovídající kompatibilitě USS. Stiskněte P, abyste je vložili.

    P095 = 0 rozlišení 0,1 Hz (implicitně nastaveno)
    1 rozlišení 0,01 Hz

12. Řízení ukládání do paměti EEPROM (volitelné). Stiskněte klávesu P. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se neobjeví P971. Stiskněte P, abyste vložili parametr. Stiskněte klávesu se šipkou nahoru nebo dolů, dokud se na displeji nezobrazí číslo odpovídající řízení paměti EEPROM. Stiskněte P, abyste je vložili.

    P971 = 0 Změny v nastavení parametrů (včetně P971) se ztratí, když je napájení odpojeno.
    1 Změny nastavení paramtetrů jsou uchovány i v době, kdy je napájení odpojeno (implicitní nastavení).

13. Pracovní displej. Stiskněte P, abyste ukončili režim nastavování parametru.

Připojení a nastavení pohonu MicroMaster řady 4

Připojení pohonu MicroMaster 4

Chcete–li připojit pohon MicroMaster řady 4 (MM4), zasuňte konce kabelu RS-485 do dvou speciálních svorek bez šroubů určených pro USS. Pro připojení automatu S7-200 je možné použít standardní kabel a konektory PROFIBUS.

Upozornění

Jestliže propojíte zařízení s různými referenčními potenciály, mohou spojovacím kabelem protékat nežádoucí proudy.

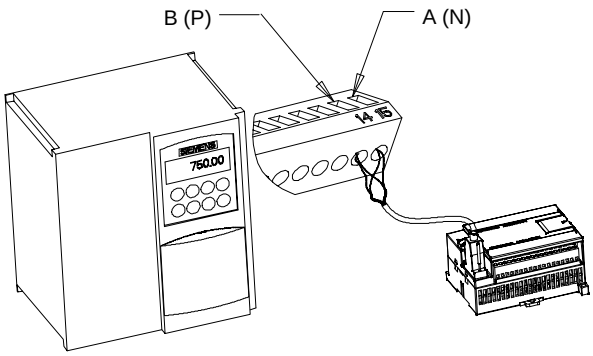
Tyto nežádoucí proudy mohou způsobit chyby v komunikaci nebo poškodit zařízení.

Aby se zabránilo průtoku nežádoucích proudů, zkontrolujte, zda všechny části zařízení, které chcete propojit komunikačním kabelem, mají buď společný referenční potenciál, nebo jsou napěťově odděleny, aby se zabránilo průtoku nežádoucích proudů.

Stínění musí být zapojeno na uzemnění rozvaděče nebo k pinu 1v 9pinovém konektoru. Doporučuje se připojit svorku 2–0V na pohonu MicroMaster k uzemnění rozvaděče.

Jak je vidět na obrázku 11-6, dva vodiče na opačném konci kabelu RS-485 musí být zasunuty do svorkovnic pohonu MM4. Chcete–li připojit pohon MM4 kabelem, sejměte kryt/kryty pohonu, abyste získali přístup ke svorkovnicím. Podrobnosti o tom, jak demontovat kryt/kryty vašeho konkrétního pohonu, najdete v příručce uživatele MM4.

Přípojné body jsou ve svorkovnici značeny čísly. Jestliže použijete konektor PROFIBUS na straně automatu S7-200, připojte kolík A kabelu do svorky 15 pohonu (pro MM420) nebo svorky 30 pohonu (MM440). Kolík B kabelového konektoru zapojte do svorky 14 (MM420) nebo svorky 29 (MM440).



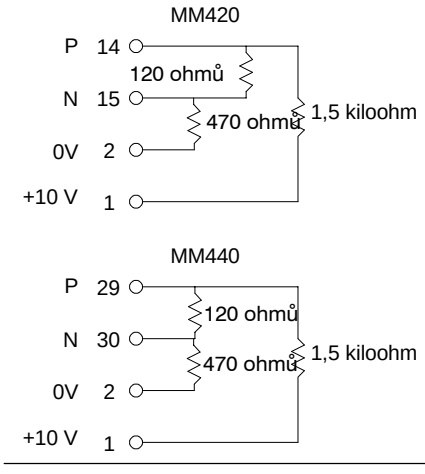
Obr. 11-6 Připojení ke svorkovnici MM420

Pokud je automat S7-200 koncovým uzlem v síti nebo pokud je spojení dvoubodové (point-to-point), je třeba použít kolíky A1 a B1 (nikoliv A2 a B2) konektoru, protože umožňují použít koncové zakončení (například u konektoru DP typ 6ES7 972–0BA40–0X40).

Upozornění

Před připojením napětí k pohonu zkontrolujte, zda jsou kryty správně připojeny na své místo.

Pokud je pohon nakonfigurován jako koncový uzel sítě, musí být zakončovací a symetrizační rezistory také zapojeny k příslušným svorkám svorkovnice. Obrázek 11-7 ukazuje příklad zapojení nutných pro zakončení a symetrizaci pohonu MM4.



Obr. 11-7 Příklad zakončení a symetrizace

Nastavení pohonu MM4

Dříve než pohon připojíte k automatu S7-200, musíte zkontrolovat, zda má pohon následující systémové parametry. Pro nastavení parametrů použijte číslicovou klávesnici na pohonu:

1. Resetujte pohon na tovární nastavení: P0010 = 30, P0970 = 1

   Pokud tento krok přeskočíte, zajistěte, aby byly následující parametry nastaveny na tyto hodnoty:
   Délka USS PZD: P2012 Index 0 = 2
   Délka USS PKW: P2013 Index 0 = 127

2. Povolte čtení/zápis všech parametrů (režim ”Expert”): P0003 = 3

3. Zkontrolujte nastavení motoru vašeho pohonu:
   P0304 = Jmenovité napětí motoru (V)
   P0305 = Jmenovitý proud motoru (A)
   P0307 = Jmenovitý výkon motoru (W)
   P0310 = Jmenovitá frekvence motoru (Hz)
   P0311 = Jmenovitá rychlost motoru (RPM)

   Nastavení se budou měnit podle použitého motoru/použitých motorů.

   Abyste mohli nastavovat parametry P304, P305, P307, P310 a P311, musíte nejprve nastavit parametr P010 na 1 (režim rychlého uvedení do provozu). Pokud jste dokončili nastavování parametrů, nastavte parametr P010 na 0. Parametry P304, P305, P307, P310 a P311 mohou být měněny pouze v režimu rychlého uvedení do provozu.

4. Nastavte režim řízení: P0700 Index 0 = 5

5. Nastavte volbu požadované hodnoty frekvence na USS na lince COM: P0700 Index 0 = 5

6. Doba rozběhu (volitelné): P1120 = 0 až 650,00

   Je to doba v sekundách, kterou motor potřebuje na to, aby zrychlil na maximální frekvenci.

7. Doba doběhu (volitelné): P1121 = 0 až 650,00

   Je to doba v sekundách, kterou motor potřebuje na to, aby zpomalil až do úplného zastavení.

8. Nastavte referenční frekvenci sériové linky: P2000 = 1 až 650 Hz

9. Nastavte normalizaci USS: P2009 Index 0 = 0

10. Nastavte přenosovou rychlost sériového rozhraní RS–485:
    P2010 Index 0 =
    4 (2400 baud)
    5 (4800 baud)
    6 (9600 baud)
    7 (19200 baud)
    8 (38400 baud)
    9 (57600 baud)
    12 (115200 baud)

11. Vložte adresu zařízení typu slave: P2011 Index 0 = 0 až 31

    Každý pohon (maximálně 31) může být řízen přes sběrnici.

12. Nastavte časovou prodlevu sériové linky: P2014 Index 0 = 0 až 65 535 ms (0 = časová prodleva blokována)

    Je to maximální povolený interval mezi dvěma příchozími datovými zprávami. Tato funkce se používá pro vypnutí invertoru v případě poruchy komunikace. Odpočítávání času začne od příjmu platné datové zprávy. Pokud ve specifikovaném časovém intervalu není přijata další datová zpráva, invertor se vypne a zobrazí se chybový kód F0070. Nastavení hodnoty na nulu vypne řízení. Pro výpočet doby mezi dotazy na stav pohonu použijte tabulku 11-1.

13. Přesuňte data z RAM do EEPROM:

    P0971 = 1 (Začátek přesunu) Uložte změny nastavení parametrů do EEPROM.

12 Použití knihovny protokolu Modbus

Knihovny instrukcí STEP 7–Micro/WIN usnadňují komunikaci se zařízeními Modbus master, protože obsahují předkonfigurované podprogramy přerušení i jiné programy, které jsou speciálně navrženy pro komunikaci Modbus. Pomocí instrukcí protokolu Modbus slave můžete nakonfigurovat automat S7-200, aby se choval jako zařízení Modbus RTU slave a komunikoval se zařízeními Modbus master.

Tyto instrukce najdete ve složce ”Knihovny” instrukčního stromu STEP 7–Micro/WIN. S těmito novými instrukcemi přimějete automat S7-200, aby se choval jako Modbus slave. Když vyberete instrukci Modbus Slave, je k projektu automaticky připojen jeden nebo více přidružených podprogramů.

Knihovny Siemens se prodávají na samostatném CD STEP 7–Micro/WIN Add-On: Knihovna instrukcí, objednací číslo 6ES7 830–2BC00–0YX0. Po zakoupení a instalaci verze 1.1 knihovny Siemens bude každý následující nainstalovaný upgrade programovacího balíku STEP 7–Micro/WIN V3.2x a V4.0 také automaticky aktualizovat knihovny bez dalších poplatků (po doplnění a modifikaci knihovny).

Přehled kapitoly

Požadavky použití protokolu Modbus 348
Inicializace a doba provádění protokolu Modbus 348
Adresování Modbus 349
Použití instrukcí protokolu Modbus slave 350
Instrukce protokolu Modbus slave 351

Požadavky použití protokolu Modbus

Instrukce protokolu Modbus slave používají v automatu S7-200 následující prostředky:

- ❏ Inicializace protokolu Modbus slave vyhradí port 0 pro komunikaci protokolu Modbus slave.

  Pokud je port 0 použit pro komunikaci protokolu Modbus slave, nemůže být použit pro žádný jiný účel včetně komunikace se STEP 7–Micro/WIN. Instrukce MBUS_INIT řídí přiřazení portu 0 protokolu Modbus slave nebo PPI.
- ❏ Instrukce protokolu Modbus slave ovlivňují všechna místa paměti SM spojená s komunikací Freeport na portu 0.
- ❏ Instrukce protokolu Modbus slave používají 3 podprogramy a 2 podprogramy přerušení.
- ❏ Instrukce protokolu Modbus slave potřebují volnou paměť o velikosti 1857 bytů pro dvě instrukce Modbus slave a podpůrné podprogramy.
- ❏ Proměnné pro instrukce protokolu Modbus slave potřebují 779bytový blok v paměti V. Počáteční adresa tohoto bloku je přidělena uživatelem a je rezervována pro proměnné Modbus.



Tip

Chcete–li změnit režim práce portu 0 zpět na PPI, abyste mohli komunikovat se STEP 7-Micro/WIN, použijte další instrukci MBUS_INIT, která změní přiřazení portu 0.

Můžete také přepnout přepínač režimu na S7-200 na režim STOP. Tím se resetují parametry portu 0.

Inicializace a doba provádění protokolu Modbus

Komunikace Modbus používá pro zajištění integrity přenášených zpráv kontrolu CRC (kontrolu cyklickým kódem). Protokol Modbus slave používá tabulku předem vypočítaných hodnot, aby se zkrátila doba potřebná pro zpracování zprávy. Inicializace této tabulky kódu CRC vyžaduje přibližně 425 milisekund. Inicializace se provádí v podprogramu MBUS_INIT. Běžně je provedena v prvním programovém cyklu uživatelského programu po přechodu do režimu RUN. Uživatel odpovídá za reset obvodu sledování funkce (watchdog) a udržování výstupů v povoleném stavu (pokud to vyžadují rozšiřovací moduly), jestliže doba potřebná pro podprogram MBUS_INIT a jakoukoliv jinou uživatelskou inicializaci překročí o 500 milisekund cyklu obvodu sledování funkce. Obvod sledování funkce výstupního modulu se resetuje zápisem na výstupy modulu. Viz instrukce pro reset obvodu sledování funkce v kapitole 6.

Doba cyklu se prodlužuje, když podprogram MBUS_SLAVE zpracovává požadavek. Protože se většina času spotřebuje na výpočet CRC, je doba cyklu prodloužena o cca 650 mikrosekund na každý byte v požadavku a v odpovědi. Nejdelší požadavek/odpověď (čtení nebo zápis 120 word) prodlužuje dobu cyklu přibližně o 165 milisekund.

Adresování Modbus

Adresy Modbus se běžně zapisují jako 5 nebo 6 znakových hodnot, které obsahují typ dat a posunutí. První jeden nebo dva znaky určují typ dat a poslední čtyři znaky volí příslušnou hodnotu v daném typu dat. Zařízení Modbus master pak mapuje adresy na příslušné funkce. Instrukce Modbus slave podporují následující adresy:

- 000001 až 000128 jsou digitální výstupy mapované na Q0.0 - Q15.7
- 010001 až 010128 jsou digitální vstupy mapované na I0.0 - I15.7
- 030001 až 030032 jsou analogové vstupní registry mapované na AIW0 až AIW62
- 040001 až 04xxxx jsou úložné registry mapované do paměti V.

Všechny adresy Modbus pracují stejně. Tabulka 12-1 ukazuje mapování adres Modbus na adresy S7-200.

Protokol Modbus slave umožňuje omezit počet vstupů, výstupů, analogových vstupů a úložných registrů (paměť V) přístupných pro Modbus master.

Parametr MaxIQ instrukce MBUS_INIT specifikuje maximální počet digitálních vstupů nebo výstupů (I nebo Q), ke kterým má Modbus master povolený přístup.

Parametr MaxAI instrukce MBUS_INIT specifikuje maximální počet vstupních registrů (AIW), ke kterým má Modbus master povolený přístup.

Parametr MaxHold instrukce MBUS_INIT specifikuje maximální počet úložných registrů (word paměti V), ke kterým má Modbus master povolený přístup.

Více informací o nastavování paměťových omezení pro Modbus slave viz popis instrukce MBUS_INIT.

Tabulka 12-1 Mapování adresy Modbus do S7-200

Adresy Modbus	Adresa S7-200
000001	Q0.0
000002	Q0.1
000003	Q0.2
...	...
000127	Q15.6
000128	Q15.7
010001	I0.0
010002	I0.1
010003	I0.2
...	...
010127	I15.6
010128	I15.7
030001	AIW0
030002	AIW2
030003	AIW4
...	...
030032	AIW62
040001	HoldStart
040002	HoldStart+2
040003	HoldStart+4
...	...
04xxxx	HoldStart+2 x (xxxx–1)

Konfigurace tabulky symbolů

Když vložíte adresu prvního symbolu, tabulka automaticky vypočítá a přiřadí zbytek symbolů v tabulce.

Tabulce, která zabírá 779 bytů, musíte přiřadit počáteční místo v paměti V. Ujistěte se, že se přiřazení symbolů Modbus slave nepřekrývá s částí paměti V přiřazené úložným registrům s parametry HoldStart a MaxHold u instrukce MBUS_INIT. Pokud by došlo k jakémukoliv překrytí paměťových oblastí, instrukce MBUS_INIT ohlásí chybu.

Použití instrukcí protokolu Modbus slave

Pro použití instrukcí protokolu Modbus slave v programu S7-200 proveďte tyto kroky:

1. Vložte do svého programu instrukci MBUS_INIT a provádějte ji pouze po dobu jednoho programového cyklu. Instrukci MBUS_INIT můžete použít buď pro inicializaci, nebo pro změnění parametrů komunikace Modbus.

   Když vložíte instrukci MBUS_INIT, přidá se k vašemu programu automaticky několik skrytých podprogramů a podprogramů přerušení.

2. Přidělte počáteční adresu pro 779 za sebou jdoucích bytů paměti V potřebných pro instrukce protokolu Modbus slave.
3. Do svého programu vložte pouze jednu instrukci MBUS_SLAVE. Tato instrukce je volána při každém programovém cyklu, aby zpracovala všechny přijaté požadavky.
4. Port 0 automatu S7–200 a zařízení Modbus master propojte komunikačním kabelem.

Upozornění
Jestliže propojíte zařízení s různými referenčními potenciály, mohou spojovacím kabelem protékat nežádoucí proudy. Tyto nežádoucí proudy mohou způsobit chyby v komunikaci nebo poškodit zařízení.

Zkontrolujte si, že všechna zařízení, která jsou propojena komunikačním kabelem, buďto mají společný referenční potenciál, nebo jsou napěťově izolována, aby se zabránilo průtoku nežádoucích proudů.

Akumulátory (AC0, AC1, AC2, AC3) jsou využívány instrukcemi Modbus slave a objevují se na seznamu křížových odkazů. Dříve než je provedena instrukce Modbus slave, jsou hodnoty uloženy v akumulátorech a pak do akumulátorů obnoveny; tím je zajištěno, že jsou v akumulátorech zachována všechna uživatelská data.

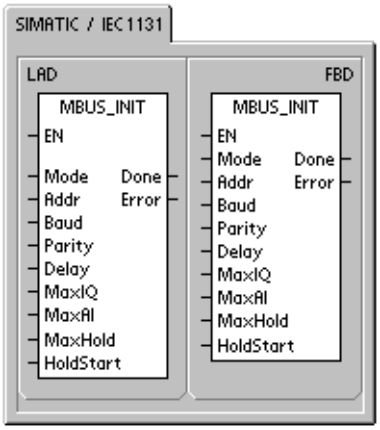
Instrukce protokolu Modbus slave podporují protokol Modbus RTU. Tyto instrukce využívají pro podporu nejběžnějších funkcí Modbus obslužné instrukce Freeport automatu S7-200. Jsou podporovány následující funkce Modbus:

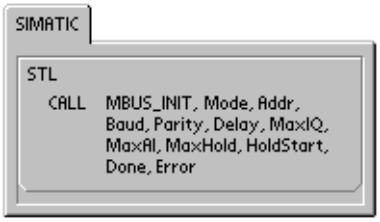
Tabulka 12-2 Podporované funkce protokolu Modbus slave

Funkce	Popis
1	Načtení stavu jednoho nebo více digitálních výstupů. Funkce 1 hlásí stav zapnuto/vypnuto libovolného počtu výstupů (Q).
2	Načtení stavu jednoho nebo více digitálních vstupů. Funkce 2 hlásí stav zapnuto/vypnuto libovolného počtu vstupů (I).
3	Načtení jednoho nebo více úložných registrů. Funkce 3 hlásí obsah paměti V. Úložné registry jsou hodnoty word pod Modbus a umožňují načtení až 120 wordů v jednom požadavku.
4	Načtení jednoho nebo více vstupních registrů. Funkce 4 hlásí hodnoty analogových vstupů.
5	Zapíše jeden digitální výstup. Funkce 5 nastavuje digitální výstup na zadanou hodnotu. Výstup není vnucený (forced) a program může přepsat hodnotu zapsanou požadavkem Modbus.
6	Zapíše jeden úložný registr. Funkce 6 zapisuje hodnotu jednoho úložného registru do paměti V automatu S7-200.
15	Zapíše více digitálních výstupů. Funkce 15 zapisuje hodnoty více digitálních výstupů do registru obrazu Q automatu S7-200. Počáteční výstupní bod musí začínat na hranici bytu (například Q0.0 nebo Q2.0); počet výstupů, na které se zapisuje, musí být násobkem osmi. To je omezení instrukcí protokolu Modbus slave. Výstupy nejsou vnucené (forced) a program může přepsat hodnoty zapsané požadavkem Modbus.
16	Zapíše více úložných registrů. Funkce 16 zapisuje více úložných registrů do paměti V automatu S7-200. V jednom požadavku může být zapsáno až 120 wordů.

Instrukce protokolu Modbus Slave

Instrukce MBUS_INIT





Instrukce MBUS_INIT se používá pro povolení a inicializaci nebo pro ukončení komunikace Modbus. Instrukce MBUS_INIT musí být provedena bez chyb dříve, než je možné použít instrukci MBUS_SLAVE. Dříve než může být provedena další instrukce, musí být prováděná instrukce ukončena a nastaven bit ”Done”.

Tato instrukce je provedena v každém programovém cyklu, když je zapnutý vstup EN.

Instrukce MBUS_INIT musí být pro každou změnu stavu komunikace provedena právě jednou. Proto by vstup EN měl být zapojen pomocí bloku vyhodnocující náběžnou hranu, nebo by měl být prováděn v prvním programovém cyklu.

Hodnota vstupu ”Mode” (Režim) volí komunikační protokol: vstupní hodnota 1 přiřazuje portu 0 protokol Modbus a povoluje protokol, vstupní hodnota 0 přiřazuje portu 0 protokol PPI a blokuje protokol Modbus.

Parametr ”Baud” nastavuje přenosovou rychlost na 1200, 2400, 4800, 9600, 19200, 38400, 57600 nebo 115200. Přenosové rychlosti 57600 a 115200 jsou podporovány S7-200 CPU verze 1.2 nebo vyšší.

Parametr ”Addr” (Adresa) nastavuje adresu na hodnoty mezi 1 a 247.

Tabulka 12-3 Parametry instrukce MBUS_INIT

Vstupy/Výstupy	Typ dat	Operandy
Mode, Addr, Parity	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, konstanta, *VD, *AC, *LD
Baud, HoldStart	DWORD	VD, ID, QD, MD, SD, SMD, LD, AC, konstanta, *VD, *AC, *LD
Delay, MaxIQ, MaxAI, MaxHold	WORD	VW, IW, QW, MW, SW, SMW, LW, AC, konstanta, *VD, *AC, *LD
Done	BOOL	I, Q, M, S, SM, T, C, V, L
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD

Parametr ”Parity” (parita) je nastaven tak, aby odpovídal paritě zařízení Modbus master. Všechna nastavení využívají jeden stop bit. Přijímané hodnoty jsou:

- 0 bez parity
- 1 lichá parita
- 2 sudá parita

Parametr ”Delay” (Zpoždění) prodlužuje standardní časovou prodlevu na konci zprávy Modbus přidáním specifikovaného počtu milisekund ke standardní časové prodlevě zprávy Modbus. Typickou hodnotou tohoto parametru při provozu v pevně zapojené síti by měla být 0. Pokud používáte modemy s opravou chyb, nastavte zpoždění na hodnotu 50 až 100 milisekund. Pokud používáte rádiové modemy s rozprostřeným pásmem, nastavte zpoždění na hodnotu 10 až 100 milisekund. Hodnota ”Delay” může být 0 až 32767 milisekund.

Parametr “MaxIQ” nastavuje počet bodů I a Q, které jsou k dispozici pro adresy Modbus 00xxxx a 01xxxx při hodnotách 0 až 128. Hodnota 0 zablokuje všechna čtení a zápisy na vstupy a výstupy. Navrhovaná hodnota MaxIQ je 128, což umožňuje přístup na všechny body I a Q automatu S7-200.

Parametr “MaxAI” nastavuje počet vstupních registrů o velikosti word (AI), které jsou k dispozici adrese Modbus 03xxx na hodnotách 0 až 32. Hodnota 0 zablokuje čtení analogových vstupů. Navrhovaná hodnota MaxAI, která umožňuje přístup ke všem analogovým vstupům automatu S7-200, je následující:

- 0 pro CPU 221
- 16 pro CPU 222
- 32 pro CPU 224, CPU 224XP a CPU 226

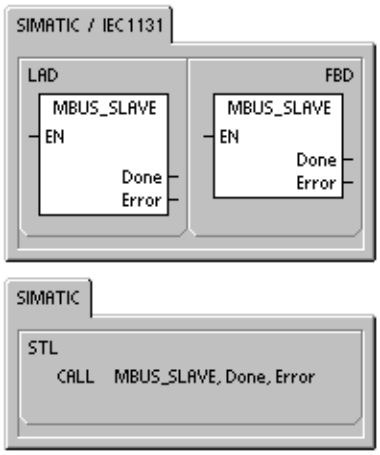
Parametr “MaxHold” nastavuje počet úložných registrů word v paměti V, které jsou k dispozici adrese Modbus 04xxx. Aby mělo například zařízení typu master přístup k 2000 bytům paměti V, nastavte MaxHold na hodnotu 1000 wordů (úložných registrů).

Parametr “HoldStart” je adresa začátku úložných registrů v paměti V. Tato hodnota se obvykle nastavuje na VB0, takže parametr HoldStart je nastaven na &VB0 (adresa VB0). Jako počáteční adresu úložných registrů je možné specifikovat jiné adresy v paměti V, aby bylo možné použít VB0 jinde v projektu. Zařízení Modbus master má přístup k počtu wordů uvedeném v MaxHold počínaje na HoldStart.

Když je instrukce MBUS_INIT dokončena, zapne se výstup ”Done”. Výstupní byte ”Error” obsahuje výsledek provádění této instrukce. Tabulka 12-5 definuje chyby, které mohou nastat při provádění této instrukce.

Instrukce MBUS_SLAVE

Instrukce MBUS_SLAVE se používá pro zpracování požadavku ze zařízení Modbus master; musí být prováděna v každém programovém cyklu, aby mu umožnila kontrolovat požadavky Modbus a odpovídat na ně.



Tato instrukce je provedena v každém programovém cyklu, když je zapnutý vstup EN.

Instrukce MBUS_SLAVE nemá žádné vstupní parametry.

Výstup ”Done” (Dokončeno) je zapnutý, pokud instrukce MBUS_SLAVE odpovídá na požadavek Modbus. Výstup ”Done” je vypnutý, když se nezpracovává žádný požadavek.

Výstup ”Error” (Chyba) obsahuje výsledek provedení této instrukce. Tento výstup je platný pouze tehdy, je–li ”Done” zapnutý. Pokud je vypnutý, chybový parametr se nemění. Tabulka 12-5 definuje chyby, které mohou nastat při provádění této instrukce.

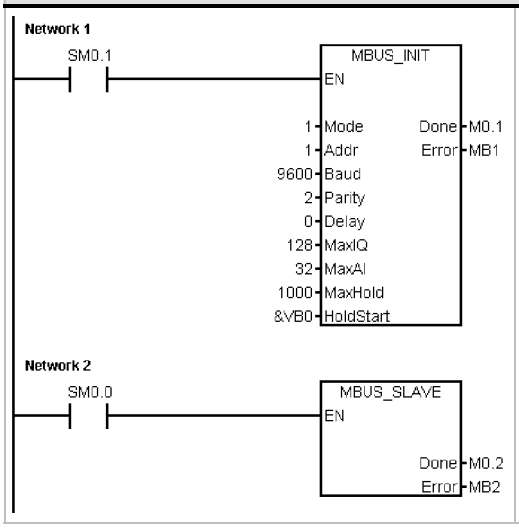
Tabulka 12-4 Parametry instrukce MBUS_SLAVE

Parametr	Typ dat	Operandy
Done	BOOL	I, Q, M, S, SM, T, C, V, L
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD

Tabulka 12-5 Chybové kódy provádění protokolu Modbus Slave

Chybové kódy	Popis
0	Bez chyby
1	Chyba rozsahu paměti
2	Nepřípustná přenosová rychlost nebo parita
3	Nepřípustná adresa zařízení slave
4	Nepřípustná hodnota parametru Modbus
5	Úložné registry překrývají symboly Modbus Slave
6	Chyba parity příjmu
7	Chyba CRC příjmu
8	Nepřípustný funkční požadavek nebo funkce není podporována
9	Nepřípustná adresa paměti v požadavku
10	Funkce slave není povolena

Příklad programování pro protokol Modbus slave




```
Network 1
        //Inicializuje protokol Modbus slave
        //v prvním programovém cyklu. Nastavte
        //adresu zařízení slave na 1, nastavte
        //port 0 na 9600 baud se sudou paritou,
        //všechny přístupy ke všem I, Q
        //a hodnotám AI, povolte přístup k úložným
        //registrům pro 1000 (2000 bytů) počínaje
        //na VB0.
LD      SM0.1
CALL    MBUS_INIT,1,1,9600,2,0,128,32,1000,
        &VB0,M0.1,MB1
Network 2
        //Provádí protokol Modbus slave
        //v každém programovém cyklu.
LD      SM0.0
CALL    MBUS_SLAVE,M0.2,MB2
```

Použití receptur

13

STEP 7 – Micro/Win má zabudován tzv. ”Průvodce recepturami”, který vám pomůže organizovat receptury a definice receptur. Receptury nejsou uloženy v PLC, ale v paměťovém modulu.

Přehled kapitoly

Přehled 356
Definice receptur a terminologie 357
Použití “Průvodce recepturami” 357
Instrukce vytvořené “Průvodcem recepturami” 361

Přehled

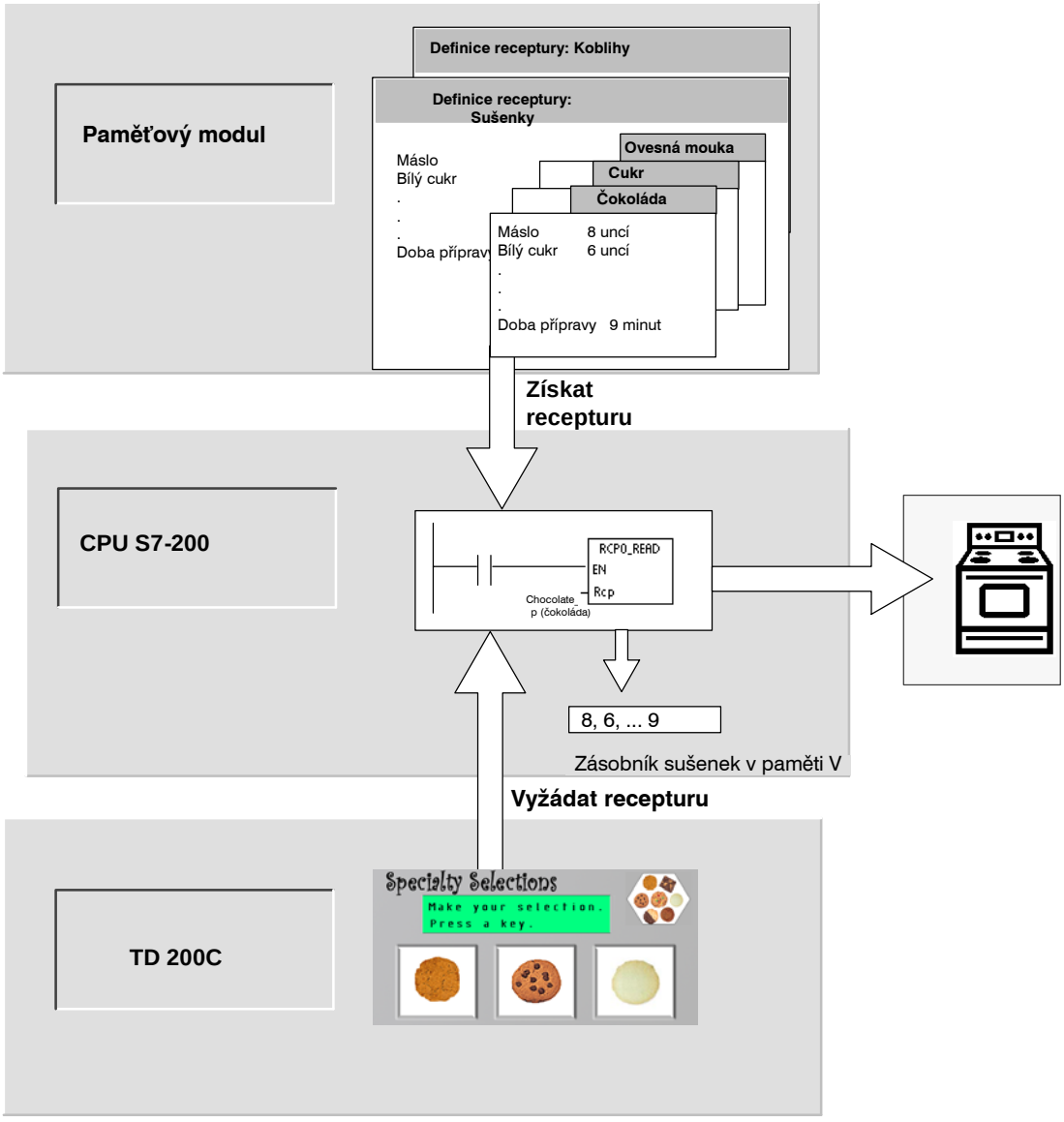

Receptura

Do programového balíku STEP 7–Micro/WIN a do S7-200 PLC byla zabudována podpora pro receptury. STEP 7–Micro/Win má zabudován “Průvodce recepturami”, který vám pomůže organizovat receptury a definice receptur.

Všechny receptury jsou uloženy v paměťovém modulu. Proto se musí do PLC nainstalovat doplňkový paměťový modul 64 kB nebo 256 kB, aby bylo možno používat funkce receptur. Další informace o paměťových modulech naleznete v příloze A.

Všechny receptury jsou uloženy v paměťovém modulu. Jednotlivá receptura se však načte do paměti PLC, jestliže uživatelský program tuto konkrétní recepturu zpracovává. Jestliže například vyrábíte sušenky, mohou být receptury pro čokoládové, cukrové a ovesné sušenky. Ve stejnou dobu lze vyrábět pouze jeden druh sušenek, takže se musí zvolit příslušná receptura a načíst do paměti PLC.

Na obrázku 13-1 je zobrazen proces pro výrobu více druhů sušenek pomocí receptur. Receptura pro každý druh sušenky je uložena v paměťovém modulu. Pomocí textového displeje TD 200C obsluha vybere druh vyráběné sušenky a uživatelský program načte tuto recepturu do paměti.



Obr. 13-1 Příklad použití receptury

Definice receptur a terminologie

Zde jsou vysvětleny definice a termíny, které vám pomohou porozumět “Průvodci recepturami”.

- ❏ Konfigurace receptury je sada komponentů projektu vytvořená “Průvodcem recepturami”. Tyto komponenty obsahují podprogramy instrukcí, záložky datových bloků a tabulky symbolů.
- ❏ Definice receptury je soubor receptur, které mají stejnou sadu parametrů. Hodnoty pro parametry se však mohou lišit podle receptury.
- ❏ Receptura je sada parametrů a hodnot parametrů, která poskytuje informace potřebné k výrobě výrobku nebo k řízení procesu.

Například lze vytvořit různé definice receptur, jako koblihy nebo sušenky. Definice receptury sušenek může obsahovat mnoho různých receptur, jako jsou čokoládové nebo cukrové sušenky. Příklady položek a jejich hodnot jsou zobrazeny v tabulce 13-1.

Tabulka 13-1 Příklad definice receptury – sušenky

Název pole	Typ dat	Chocolate_Chip (čokoláda) (Receptura 0)	Sugar (cukr) (Receptura 1)	Poznámka
Butter (máslo)	byte	8	8	unce
White_Sugar (bílý cukr)	byte	6	12	unce
Brown_Sugar (hnědý cukr)	byte	6	0	unce
Eggs (vejce)	byte	2	1	kus
Vanilla (vanilka)	byte	1	1	kávová lžička
Flour (mouka)	byte	18	32	unce
Baking_Soda (jedlá soda)	reálný	1,0	0,5	kávová lžička
Baking_Powder (prášek do pečiva)	reálný	0	1,0	kávová lžička
Salt (sůl)	reálný	1,0	0,5	kávová lžička
Chocolate_Chips (čokoláda)	reálný	16	0,0	unce
Lemon_Peel (citronová kůra)	reálný	0,0	1,0	polévková lžíce
Cook_Time (doba přípravy)	reálný	9,0	10,0	minuty

Použití “Průvodce recepturami”

“Průvodce recepturami” použijte k vytvoření receptur a definicí receptur. Receptury jsou uloženy v paměťovém modulu. Receptury a definice receptur lze zadávat přímo v “Průvodci recepturami”. Další změny jednotlivých receptur lze provádět opět “Průvodcem recepturami” nebo programováním pomocí podprogramu instrukce RCPx_WRITE.

“Průvodce recepturami” vytvoří konfiguraci receptury, která obsahuje:

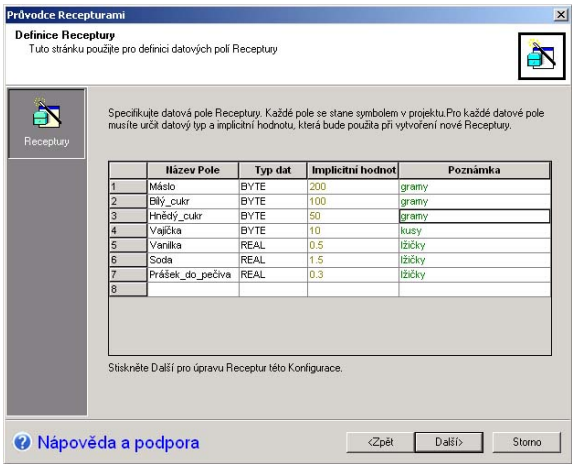
- ❏ Tabulku symbolů pro každou definici receptury. Každá tabulka obsahuje názvy symbolů, které jsou stejné jako názvy polí receptur. Tyto symboly definují adresy paměti V potřebné k přístupu k hodnotám receptury aktuálně načtené v paměti. Každá tabulka také obsahuje symbolickou konstantu k odkazu na jednotlivou recepturu.
- ❏ Záložku datového bloku pro každou definici receptury. Tato záložka definuje počáteční hodnoty pro každou adresu paměti V zastoupenou v tabulce symbolů.
- ❏ Podprogram instrukce RCPx_READ. Tato instrukce se používá k načtení specifikované receptury z paměťového modulu do paměti V.
- ❏ Podprogram instrukce RCPx_WRITE. Tato instrukce se používá k zápisu hodnot receptury z paměti V do paměťového modulu.

Definování receptur

Chcete–li vytvořit recepturu pomocí “Průvodce recepturami”, zvolte příkaz menu **Nástroje > Průvodce recepturami**. První obrazovkou je úvodní okno definující základní operace “Průvodce recepturami”. Kliknutím na tlačítko “Další” zahájíte konfiguraci vašich receptur.

K vytvoření definice receptury postupujte podle těchto kroků. Viz obrázek 13-2.

1. Specifikujte názvy polí pro každou definici receptury. Každý název se stane symbolem ve vašem dříve definovaném projektu.
2. Zvolte datový typ z rozbalovacího seznamu.
3. Pro každý název zadejte implicitní hodnotu a poznámku. Všechny nové receptury specifikované v této definici začnou těmito implicitními hodnotami.
4. Kliknutím na tlačítko “Další” vytvoříte a upravíte receptury pro tuto definici receptury.



Obr. 13-2 Definování receptur

K definování všech datových polí receptury použijte tolik řádků, kolik bude potřeba. Můžete mít až čtyři různé definice receptury. Počet receptur pro každou definici je omezen pouze volným místem v paměťovém modulu.

Vytvoření a editování receptur

Okno “Vytvořit a editovat receptury” vám umožní vytvářet jednotlivé receptury a specifikovat hodnoty pro tyto receptury. Každý editovatelný sloupec představuje jedinečnou recepturu.

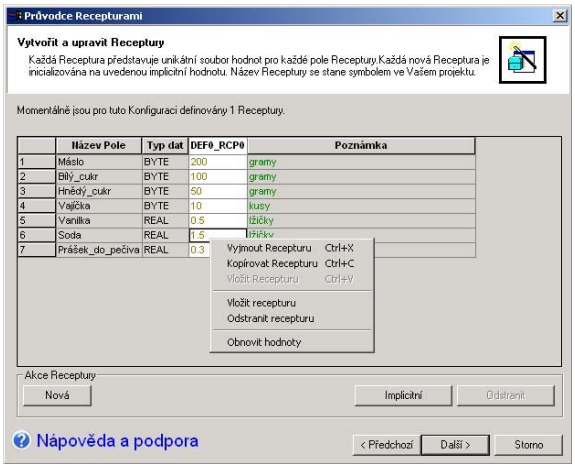
Receptury lze vytvořit stisknutím tlačítka “Nová”. Každá receptura začíná implicitními hodnotami, specifikovanými při vytváření definice receptury.

Receptury lze také vytvořit pomocí kontextového menu po stisknutí pravého tlačítka myši zkopírováním a vložením stávajících receptur. Nalevo od polohy kurzoru budou vloženy nové sloupce včetně pole poznámky.

Každá nová receptura dostane implicitní název, který obsahuje odkaz na definici receptury a číslo receptury. Tento název bude ve tvaru DEFx_RCPy.

K vytvoření a editaci receptury postupujte podle těchto kroků. Viz obrázek 13-3.

1. Kliknutím na tlačítko “Další” se otevře okno “Vytvořit a editovat recepturu”.
2. Podle potřeby vložte nový recept stisknutím tlačítka “Nová”.
3. Přejmenujte název receptury na příslušný explicitní název.
4. Podle potřeby změňte hodnoty dat jednotlivé receptury.
5. Klikněte na OK.



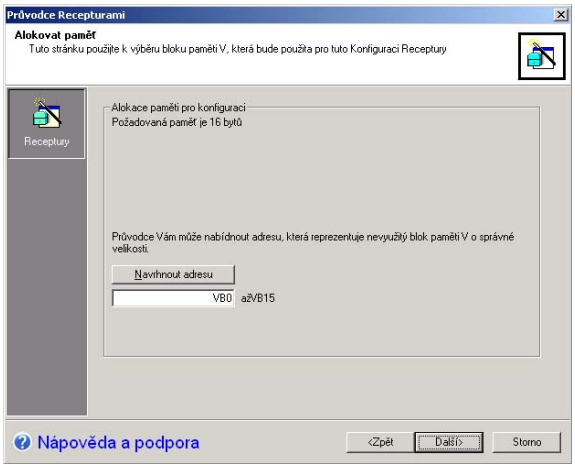
Obr. 13-3 Vytvoření a editování receptur

Přidělování paměti

Okno “Přidělování paměti” specifikuje počáteční adresu paměti V, do které se uloží receptura načtená z paměťového modulu. K přidělení adresy nevyužitého paměťového bloku V správné velikosti můžete buď zvolit adresu paměti V, nebo ji nechat přidělit “Průvodcem recepturami”.

K přidělení paměti postupujte podle následujících kroků. Viz obrázek 13-4.

1. Abyste vybrali adresu paměti V, kam chcete uložit recepturu, klikněte do okna a zadejte adresu.
2. Chcete–li nechat “Průvodce recepturami” vybrat nevyužitý paměťový blok správné velikosti, klikněte na tlačítko “Navrhnout adresu”.
3. Klikněte na tlačítko “Další”.



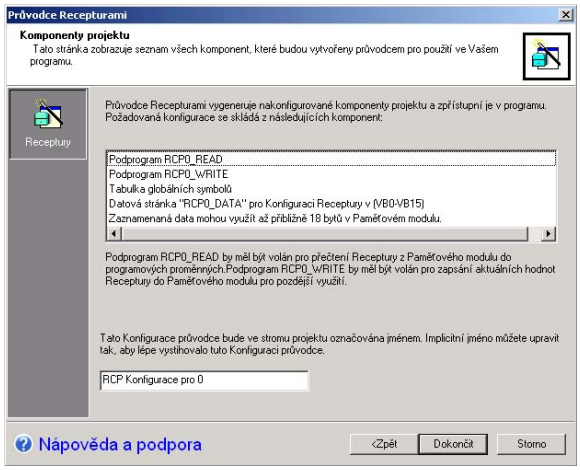
Obr. 13-4 Přidělování paměti

Komponenty projektu

V okně komponent projektu je seznam různých komponent, které budou přidány do vašeho projektu. Viz obrázek 13-5.

Klinutím na tlačítko “Dokončit” ukončíte “Průvodce recepturami” a přidáte tyto komponenty.

Každé konfiguraci receptury lze přidělit jedinečný název. Tento název bude zobrazen ve stromu projektu s každou konfigurací průvodce. Definice receptury (RCPx) bude připojena ke konci tohoto názvu.



Obr. 13-5 Komponenty projektu

Použití tabulky symbolů

Tabulka symbolů je vytvořena pro každou definici receptury. Každá tabulka definuje konstantní hodnoty, které představují každou recepturu. Tyto symboly lze použít jako parametry pro instrukce RCPx_READ a RCPx_WRITE k označení požadované receptury. Viz obrázek 13-6.

Každá tabulka také vytváří názvy symbolů pro každé pole receptury. Tyto symboly můžete použít k přístupu k hodnotám receptury v paměti V.



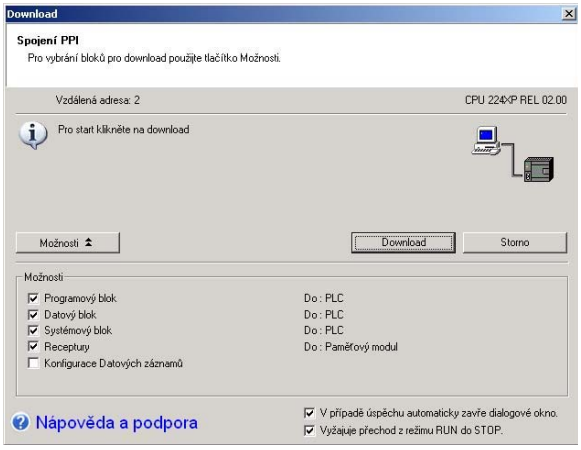
	Název	Adresa	Komentář
1	DEF0_RCP0	0	
2	Prášek_do_pečiva	VD12	lžičky
3	Soda	VD8	lžičky
4	Vanilka	VD4	lžičky
5	Vajíčka	VB3	kusy
6	Hnědý_cukr	VB2	gramy
7	Bílý_cukr	VB1	gramy
8	Máslo	VB0	gramy

Obr. 13-6 Tabulka symbolů

Download projektu s konfigurací receptury

Download projektu, který obsahuje konfiguraci receptury, proveďte podle následujících kroků. Viz obrázek 13-7.

1. Vyberte **Soubor > Download**.
2. V dialogovém okně pod záložkou “Možnosti” zaškrtněte políčka “Programový blok”, “Datový blok” a “Receptury”.
3. Klikněte na tlačítko “Download”.

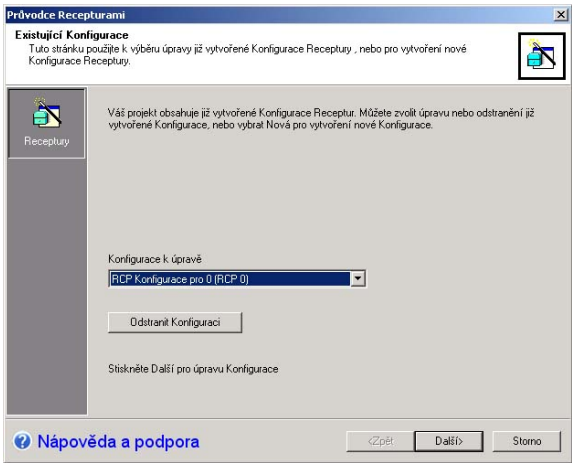


Obr. 13-7 Download projektu s konfigurací receptury

Editace stávajících konfigurací receptur

Editaci stávajících konfigurací receptur proveďte podle následujících kroků. Viz obrázek 13-8.

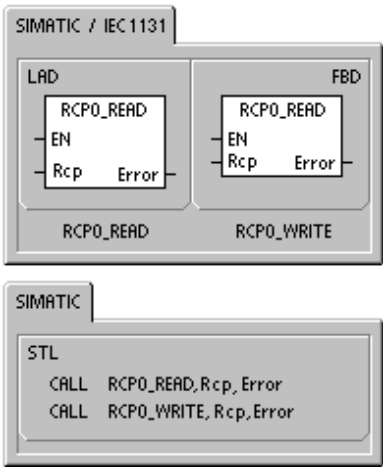
1. Klikněte na rozbalovací seznam konfigurací a vyberte stávající konfiguraci receptury.
2. Kliknutím na tlačítko “Vymazat konfiguraci” vymažete stávající konfiguraci receptury.



Obr. 13-8 Editování stávajících konfigurací receptur

Instrukce vytvořené “Průvodcem recepturami”

Podprogram RCPx_Read



Podprogram RCPx_READ je vytvořen “Průvodcem recepturami” a používá se k načtení jednotlivé receptury z paměťového modulu do specifikovaného místa v paměti V.

”x” v instrukci RCPx_READ odpovídá definici receptury, která obsahuje recepturu, kterou chcete načíst.

Vstup EN umožňuje provedení instrukce, když je vstup ve vysoké úrovni.

Vstup “Rcp” označuje recepturu, která bude načtena z paměťového modulu.

Výstup ”Error” (Chyba) vrací výsledek provedení této instrukce. Definice chybových kódů jsou popsány v tabulce 13-3.

Podprogram RCPx_Write

Podprogram RCPx_WRITE je vytvořen “Průvodcem recepturami” a používá se k nahrazení receptury v paměťovém modulu obsahem receptury obsaženým v paměti V .

”x” v instrukci RCPx_WRITE odpovídá definici receptury, která obsahuje recepturu, již chcete nahradit.

Vstup EN umožňuje provedení instrukce, když je vstup ve vysoké úrovni.

Vstup “Rcp” označuje recepturu, která bude nahrazena v paměťovém modulu.

Výstup ”Error” (Chyba) vrací výsledek provedení této instrukce. Definice chybových kódů jsou popsány v tabulce 13-3.

Tabulka 13-2 Operandy platné pro podprogram receptury

Vstupy/Výstupy	Typ dat	Operandy
Rcp	Word	VW, IW, QW, MW, SW, SMW, LW, AC, *VD, *AC, *LD, konstanta
Error	Byte	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD

Tabulka 13-3 Chybové kódy pro instrukce receptur

Chybové kódy	Popis
0	Bez chyby
132	Přístup do paměťového modulu neúspěšný

Tip

EEPROM používaná v paměťovém modulu podporuje omezený počet zápisů. Standardně je to jeden milion zapisovacích cyklů. Jakmile je tento limit dosažen, EEPROM nebude fungovat správně.

Zajistěte, aby instrukce RCPx_WRITE nebyla aktivována v každém programovém cyklu. Aktivací této instrukce při každém cyklu se paměťový modul opotřebuje v relativně krátké době.

Použití datových záznamů

14

STEP 7–Micro/Win obsahuje “Průvodce datovým záznamem” k uložení naměřených dat v procesu do paměťového modulu. Přesunem dat procesu do paměťového modulu se uvolní adresy paměti V, které by jinak byly nutné k uložení těchto dat.

Přehled kapitoly

Přehled 364
Použití “Průvodce datovým záznamem” 365
Instrukce vytvořená “Průvodcem datovým záznamem” 369

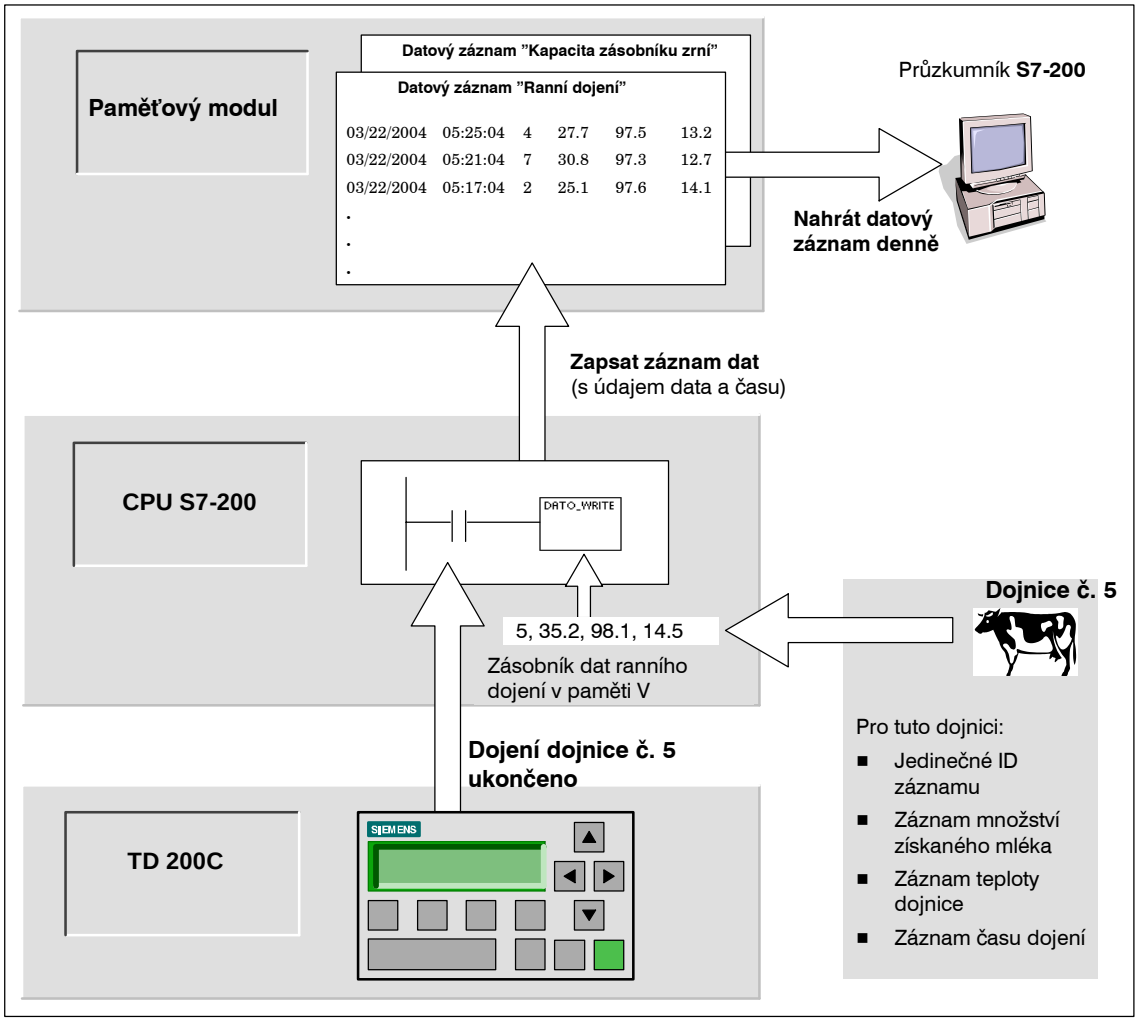
Přehled

Do programového balíku STEP 7–Micro/WIN a do S7-200 PLC byla zabudována podpora pro datové záznamy. Použitím této funkce můžete trvale ukládat záznamy obsahující data procesu při řízení programem. Tyto záznamy mohou volitelně obsahovat časový údaj a datum. Můžete konfigurovat až čtyři samostatné datové záznamy. Formát záznamu dat se definuje pomocí nového "Průvodce datovým záznamem".

Všechny datové záznamy jsou uloženy v paměťovém modulu. Abyste mohli využívat funkci datového záznamu, musíte mít ve svém PLC nainstalován doplňkový paměťový modul 64 kB nebo 256 kB. Informace o paměťových modulech naleznete v příloze A.

Abyste mohli načíst obsah datových záznamů do svého počítače, musíte použít Průzkumník S7-200.

Příklad použití datového záznamu je uveden na obrázku 14-1.



Obr. 14-1 Příklad použití datového záznamu

Definice a terminologie datového záznamu

Zde jsou vysvětleny definice a termíny, které vám pomohou porozumět "Průvodci dat. záznamem".

- Datový záznam je sada záznamů obvykle řazená podle data a času. Každý záznam představuje nějakou událost procesu, která zaznamenává sadu dat procesu. Organizace těchto dat je definována Průvodcem datovým záznamem.
- Položka datového záznamu je jedna řada dat zapsaná do datového záznamu.

Použití “Průvodce datovým záznamem”

“Průvodce datovým záznamem” použijte ke konfiguraci až čtyř datovývh záznamů. “Průvodce datovým záznamem” se používá:


Datový záznam

- ❑ k definování formátu položky datového záznamu
- ❑ k volbě možností datového záznamu, jako je časový údaj a datum a vymazání datového záznamu při nahrání
- ❑ ke specifikaci maximálního počtu položek, které lze uložit v datovém záznamu
- ❑ k vytvoření kódu projektu použitého k ukládání položek v datovém záznamu.

“Průvodce datovým záznamem” vytvoří konfiguraci datového záznamu, která obsahuje:

- ❑ Tabulku symbolů pro každou konfiguraci datového záznamu. Každá tabulka obsahuje názvy symbolů, které jsou stejné jako názvy polí datových záznamů. Každý symbol definuje adresy paměti V, potřebné k uložení aktuálního datového záznamu. Každá tabulka také obsahuje symbolickou konstantu k odkazu na jednotlivý datový záznam.
- ❑ Záložku bloku dat pro každou položku datového záznamu, která přiděluje adresy paměti V pro každé pole datového záznamu. Váš program používá tyto adresy paměti V ke shromažďování aktuální sady datového záznamu.
- ❑ Podprogram DATx_WRITE. Tato instrukce kopíruje specifikovanou položku datového záznamu z paměti V do paměťového modulu. Každé provedení instrukce DATx_WRITE přidá novou položku do datového záznamu uloženého v paměťovém modulu.

Možnosti datového záznamu

Můžete konfigurovat následující možnosti chování datového záznamu. Viz obrázek 14-2.

Časový údaj

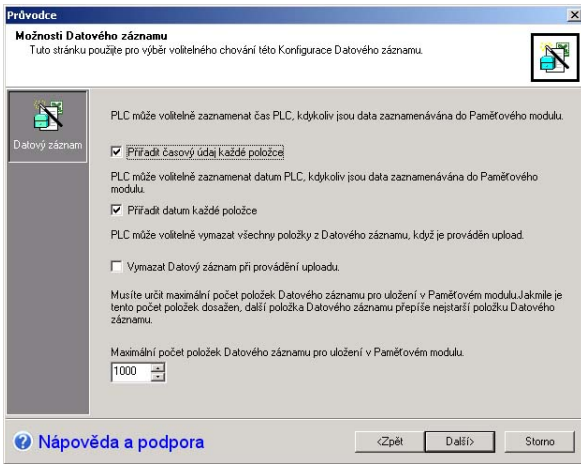
Ke každé položce datového záznamu můžete přidat časový údaj. Je–li tato možnost zvolena, CPU automaticky zahrne časový údaj ke každé položce, jestliže uživatelský program nařídí zápis datového záznamu.

Datum

Ke každé položce datového záznamu můžete přidat datum. Je–li tato možnost zvolena, CPU automaticky zahrne datum ke každé položce, jestliže uživatelský program nařídí zápis datového záznamu.

Vymazání datového záznamu

Vymazání datového záznamu – můžete vymazat všechnypoložky z datového záznamu při každém nahrávání. Jestliže zvolíte možnost vymazání datového záznamu, datový záznam bude vymazán při každém nahrávání dat.



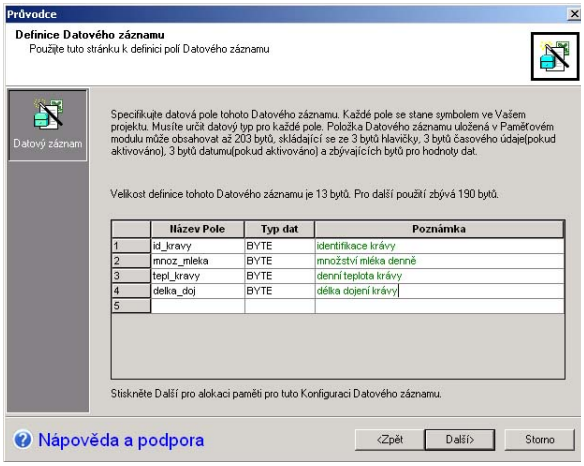
Obr. 14-2 Možnosti datového záznamu

Datové záznamy jsou realizovány jako cyklická fronta (když je záznam plný, nová položka nahradí nejstarší položku). Musíte specifikovat maximální počet položek, které mají být uloženy v datovém záznamu. Maximální počet položek povolených v datovém záznamu je 65 535. Implicitní hodnota pro počet položek je 1000.

Definování datového záznamu

Specifikujete pole pro datový záznam a každé pole se stává symbolem ve vašem projektu. Pro každé pole musíte specifikovat typ dat. Položka datového záznamu může obsahovat 4 až 203 bytů dat. Definování datových polí v datovém záznamu proveďte následujícím způsobem. Viz obrázek 14-3.

1. Kliknutím na “Název pole” zadejte název. Název se stane symbolem, na který se odkazuje uživatelský program.
2. Klikněte na “Typ dat” a vyberte typ dat z rozbalovacího seznamu.
3. Kliknutím na “Poznámku” zadáte poznámku.
4. K definování položky použijte tolik řad, kolik je potřeba.
5. Klikněte na OK.



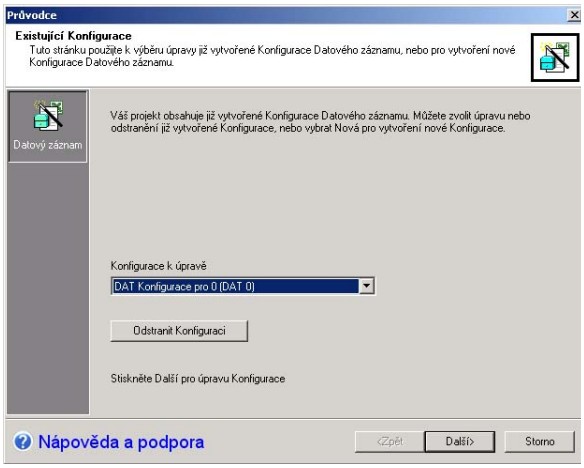
Obr. 14-3 Definování položky datového záznamu

Editace stávající konfigurace datového záznamu

Editaci stávající konfigurace datového záznamu proveďte následujícím způsobem:

1. Klikněte na rozbalovací seznam konfigurace a vyberte stávající konfiguraci datového záznamu, jak je uvedeno na obrázku 14-4.
2. Kliknutím na tlačítko “Vymazat konfiguraci” vymažete stávající konfiguraci datového záznamu.

Můžete mít až čtyři různé datové záznamy.



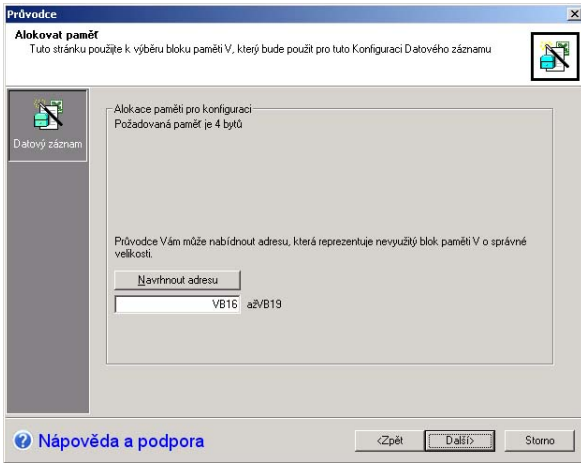
Obr. 14-4 Editace stávajících konfigurací datovýh záznamů

Přidělování paměti

"Průvodce datovým záznamem" vytvoří blok v paměti V vašeho PLC. Tento blok je adresa paměti, kde bude vytvořena položka datového záznamu, než bude zapsána do paměťového modulu. Specifikujete počáteční adresu paměti V, kam chcete umístit konfiguraci. K přidělení adresy nevyužitého paměťového bloku V správné velikosti můžete buď zvolit adresu paměti V, nebo ji nechat přidělit "Průvodcem datovým záznamem". Velikost bloku se liší podle konkrétních voleb, které jste provedli v "Průvodci datovým záznamem". Viz obrázek 14-5.

K přidělení paměti postupujte podle následujících kroků:

1. Abyste vybrali adresu paměti V, kde bude vytvořena položka datového záznamu, klikněte do oblasti "Navrhnout adresu" a zadejte adresu.
2. Chcete–li nechat "Průvodce datovým záznamem" vybrat nevyužitý paměťový blok správné velikosti, klikněte na tlačítko "Navrhnout adresu".
3. Klikněte na tlačítko "Další".



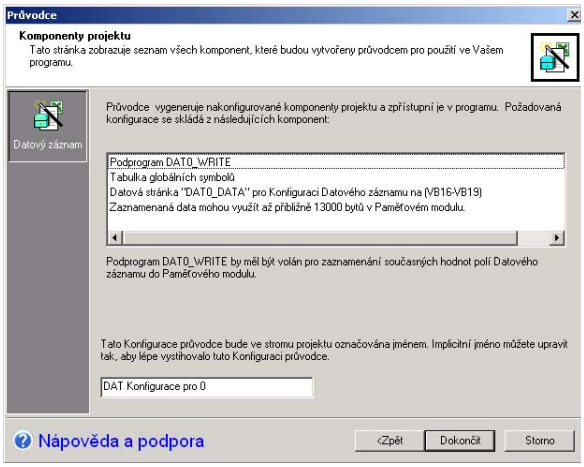
Obr. 14-5 Přidělování paměti

Komponenty projektu

V okně komponent projektu je seznam různých komponent, které budou přidány do vašeho projektu. Viz obrázek 14-6.

Klinutím na tlačítko "Dokončit" ukončíte "Průvodce datovým záznamem" a přidáte tyto komponenty.

Každé konfiguraci datového záznamu lze přidělit jedinečný název. Tento název bude zobrazen ve stromu projektu s každou konfigurací průvodce. Definice datového záznamu (DATx) bude připojena ke konci tohoto názvu.



Obr. 14-6 Komponenty projektu

Použití tabulky symbolů

Pro každou konfiguraci datového záznamu je vytvořena tabulka symbolů. Každá tabulka definuje konstantní hodnoty, které představují každý datový záznam. Tyto symboly lze použít jako parametry pro instrukce DATx_WRITE.

Každá tabulka také vytváří názvy symbolů pro každé pole datového záznamu. Tyto symboly můžete použít k přístupu k hodnotám datového záznamu v paměti V.



	Název	Adresa	Komentář
1	delka_doj	VB19	délka dojení krávy
2	tepl_kravy	VB18	denní teplota krávy
3	mnoz_mleka	VB17	množství mléka denně
4	id_kravy	VB16	identifikace krávy

Obr. 14-7 Tabulka symbolů

Download projektu, který obsahuje konfiguraci datového záznamu

Před použitím datového záznamu musíte provést download projektu, který obsahuje konfiguraci datového záznamu, do CPU S7-200. Jestliže nějaký projekt má konfiguraci datového záznamu, pak okno pro download má implicitně zaškrtnutou možnost konfigurací datových záznamů.

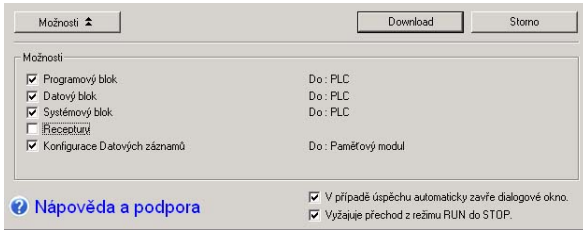


Tip

Při downloadu projektu, který obsahuje konfigurace datových záznamů, budou všechny položky datového záznamu uložené v paměťovém modulu ztraceny.

Download projektu, který obsahuje konfigurace datových záznamů, proveďte podle následujících kroků. Viz obrázek 14-8.

1. Vyberte **Soubor > Download**.
2. V dialogovém okně na záložce “Možnosti” zatrhněte políčko “Konfigurace datového záznamu”.
3. Klikněte na tlačítko “Download”.



Obr. 14-8 Download projektu s konfigurací datového záznamu

Použití Průzkumníka S7-200

Průzkumník S7-200 je aplikace používaná k načtení datového záznamu z paměťového modulu, a k následnému uložení datového záznamu do souboru s hodnotami oddělenými čárkou (CSV).

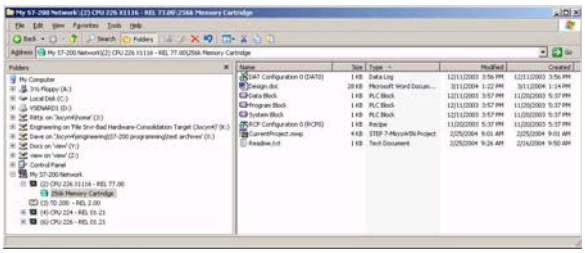
Při každém čtení datového záznamu se vytvoří nový soubor. Tento soubor je uložen v adresáři “Datový záznam”. Formát názvu souboru je následující: adresa PLC, název datového záznamu, datum, čas.

Můžete zvolit, zda se po úspěšném načtení datového záznamu automaticky spustí aplikace asociovaná s příponou CSV. Tuto volbu lze provést z menu datového záznamu po kliknutím pravým tlačítkem myši.

Adresář datového záznamu bude pod adresářem specifikovaným během instalace. Implicitní adresář při instalaci je c:\program files\siemens\Microsystems (pokud není nainstalován STEP 7). Implicitní instalace je c:\siemens\Microsystems (je – li nainstalován STEP 7).

Načtení datového záznamu proveďte následujícím způsobem:

1. Otevřete průzkumník Windows. Měla by se automaticky zobrazit složka “My S7-200 Network”.
2. Zvolte složku “My S7-200 Network”.
3. Zvolte správnou složku “My S7-200 Network”.
4. Zvolte složku paměťového modulu
5. Najděte správný soubor konfigurace datového záznamu. Tyto soubory budou mít název DAT Configuration x (DATx).
6. Klikněte pravým tlačítkem myši a z kontextového menu zvolte “Upload” (Nahrát).



Obr. 14-9 Použití průzkumníku S7-200

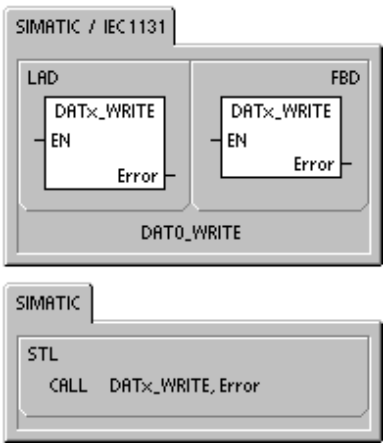
Instrukce vytvořená “Průvodcem datovým záznamem”

“Průvodce datovým záznamem” přidá do vašeho projektu jeden podprogram instrukce.

Podprogram DATx_WRITE

Podprogram DATx_WRITE se používá k zápisu aktuálních hodnot polí datových záznamů do paměťového modulu. DATxWRITE přidá jednu položku k zapsaným datům do paměťového modulu. Volání tohoto podprogramu se objeví následovně.

Jestliže tato instrukce nemá správný přístup do paměťového modulu, vrátí se chyba 132.



Tabulka 14-1 Parametry pro podprogram DATAx_WRITE

Vstupy/Výstupy	Typ dat	Operandy
Error	byte	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD



Tip

EEPROM používaná v paměťovém modulu podporuje omezený počet zápisů. Standardně je to jeden milion zapisovacích cyklů. Jakmile je tento limit dosažen, EEPROM nebude fungovat správně.

Zajistěte, aby instrukce DATx_WRITE nebyla aktivována v každém programovém cyklu. Aktivací této instrukce při každém cyklu se paměťový modul opotřebuje v relativně krátké době.

15 Adaptivní PID a Ovládací panel PID

Možnost adaptace PID byla zabudována do PLC S7-200 a do programovacího balíku STEP 7–Micro/WIN byl přidán Ovládací panel PID. Tyto funkce společně značně zvyšují užitnost a snadné používání funkce PID poskytované řadou Micro–PLC S7-200.

Adaptace může být spuštěna uživatelským programem z panelu obsluhy nebo z Ovládacího panelu PID. Smyčky PID lze adaptovat jednu po druhé nebo v případě potřeby lze adaptovat všech osm smyček současně. Adaptace PID vypočítává doporučené (téměř optimální) hodnoty pro nastavení zesílení, integrační časové konstanty (reset) a derivační časové konstanty (rate). Také vám umožňuje zvolit nastavení pro rychlou, střední, pomalou nebo velmi pomalou odezvu vaší smyčky.

Pomocí Ovládacího panelu PID můžete spustit proces adaptace, přerušit proces adaptace a monitorovat výsledky v grafické formě. Na ovládacím panelu se zobrazí všechny chybové stavy nebo výstrahy, které se mohou vytvořit. Také vám umožňuje použít hodnoty zesílení, reset a rate vypočítané adaptací.

Přehled kapitoly

Vysvětlení adaptace PID 372
Rozšířená tabulka smyčky 372
Požadavky 375
Automatická hystereze a automatická odchylka 375
Sekvence adaptace 376
Výjimečné stavy 376
Poznámky týkající se PV mimo rozsah (výsledný kód 3) 378
Ovládací panel PID 378

Vysvětlení adaptace PID

Úvod

Adaptační algoritmus používaný v S7-200 je založen na technice nazývané reléová zpětná vazba, navržené K. J. Ĺströmem a T. Hägglundem v roce 1984. V posledních dvaceti letech se reléová zpětná vazba používá v široké škále průmyslových odvětví.

Koncept reléové zpětné vazby je vytvořit malé, ale ustálené oscilace v jinak stabilním procesu. Na základě doby oscilací a změn amplitudy, sledovaných v proměnné procesu, se určuje mezní frekvence a mezní zesílení procesu. Potom pomocí hodnot mezního zesílení a mezní frekvence navrhuje adaptační algoritmus PID hodnotu nastavení pro zesílení, reset a rate.

Navržené hodnoty závisí na vaší volbě rychlosti odezvy smyčky pro váš proces. Můžete zvolit rychlou, střední, pomalou nebo velmi pomalou odezvu. V závislosti na vašem procesu může mít rychlá odezva překmit a odpovídala by nedostatečně tlumené odezvě systému. Střední rychlost odezvy může být na pokraji aperiodicity a odpovídala by kriticky tlumené odezvě systému. Pomalá rychlost nemusí mít žádný překmit a odpovídala by přetlumené odezvě systému. Velmi pomalá rychlost nesmí mít žádný překmit a odpovídala by silně přetlumené odezvě systému.

Kromě navrhování hodnot parametrů může adaptační algoritmus PID určit hodnoty pro hysterezi a maximální odchylku PV. Tyto parametry se používají ke snížení účinku šumu procesu při omezení amplitudy ustálených oscilací nastavených adaptačním algoritmem PID.

Adaptační algoritmus PID je schopen stanovit navrhované hodnoty nastavení pro přímo i inverzně působící smyčky P, PI, PD a PID.

Účelem adaptačního algoritmu PID je stanovit sadu parametrů, které poskytnou přiměřenou aproximaci optimálních hodnot pro vaši smyčku. Spuštění s navrženými hodnotami nastavení vám umožní provést jemné doladění a správně optimalizovat proces.

Rozšířená tabulka smyčky

Instrukce PID pro S7-200 se odkazuje na tabulku smyčky, která obsahuje parametry smyčky. Tato tabulka byla původně dlouhá 36 bytů. Po přidání adaptace PID byla tabulka rozšířena a nyní je dlouhá 80 bytů. Rozšířená tabulka smyčky je uvedena v tabulce 15-1 a 15-2.

Jestliže používáte Ovládací panel PID, veškerá interakce s tabulkou smyčky PID je za vás prováděna ovládacím panelem. Jestliže potřebujete zajistit možnost adaptace z panelu obsluhy, váš program musí zajistit interakci mezi obsluhou a tabulkou smyčky PID, aby spustil a monitoroval proces adaptace a potom použil doporučené hodnoty nastavení.

Tabulka 15-1 Tabulka smyčky

Offset	Pole	Formát	Typ	Popis
0	Regulovaná veličina (PV_n)	REAL	vstup	Obsahuje regulovanou veličinu, která musí být v rozsahu mezi 0,0 a 1,0.
4	Žádaná hodnotaí (SP_n)	REAL	vstup	Obsahuje požadovanou hodnotu, která musí být v rozsahu mezi 0,0 a 1,0.
8	Výstup (M_n)	REAL	vstup/ výstup	Obsahuje vypočítanou výstupní hodnotu, která musí být v rozsahu mezi 0,0 a 1,0.
12	Zesílení (K_C)	REAL	vstup	Obsahuje zesílení, které je proporcionální konstanta. Může to být kladné nebo záporné číslo.
16	Vzorkovací perioda (T_S)	REAL	vstup	Obsahuje periodu vzorkování v sekundách. Musí to být kladné číslo.
20	Integrační časová konstanta (T_I)	REAL	vstup	Obsahuje integrační časovou konstantu v minutách.
24	Derivační časová konstanta (T_D)	REAL	vstup	Obsahuje derivační časovou konstantu v minutách.
28	Integrační suma (MX)	REAL	vstup/ výstup	Obsahuje hodnotu integrační sumy v rozsahu mezi 0,0 a 1,0.
32	Předcházející regulovaná veličina (PV_{n-1})	REAL	vstup/ výstup	Obsahuje hodnotu regulované veličiny, uloženou od posledního provedení instrukce PID.
36	ID rozšířené tabulky PID	ASCII	Konstanta	'PIDA' (rozšířená tabulka PID, verze A): Konstanta ASCII
40	Řízení AT (ACNTL)	BYTE	vstup	Viz tabulku 15-2
41	Stav AT (ASTAT)	BYTE	výstup	Viz tabulku 15-2
42	Výsledek AT (ARES)	BYTE	vstup/ výstup	Viz tabulku 15-2
43	Konfigurace AT (ACNFG)	BYTE	vstup	Viz tabulku 15-2
44	Odchylka (DEV)	REAL	vstup	Normalizovaná hodnota maximální amplitudy oscilace PV (rozsah: 0,025 až 0,25).
48	Hystereze (HYS)	REAL	vstup	Normalizovaná hodnota hystereze PV používaná ke stanovení průchodů nulou (rozsah: 0,005 až 0,1). Jestliže je poměr DEV k HYS menší než 4, během adaptace se objeví výstraha.
52	Počáteční výstupní skok (STEP)	REAL	vstup	Normalizovaná velikost skokové změny výstupní hodnoty, používané k vyvolání oscilací v PV (rozsah: 0,05 až 0,4).
56	Watchdog (WDOG)	REAL	vstup	Maximální čas povolený mezi nulovými průsečíky ve vteřinách (rozsah: 60 až 7200).
60	Doporučené zesílení (AT_K_C)	REAL	výstup	Doporučené zesílení smyčky, jak je stanoveno procesem adaptace.
64	Doporučená integrační časová konstanta (AT_T_I)	REAL	výstup	Doporučená integrační časová konstanta, jak je stanovena procesem adaptace.
68	Doporučená derivační časová konstanta (AT_T_D)	REAL	výstup	Doporučená derivační časová konstanta, jak je stanovena procesem adaptace.
72	Okamžitá velikost skoku (ASTEP)	REAL	výstup	Normalizovaná velikost výstupního skoku, jak je stanovena procesem adaptace.
76	Okamžitá hystereze (AHYS)	REAL	výstup	Normalizovaná hodnota hystereze PV, jak je stanovena procesem adaptace.

Tabulka 15-2 Rozšířený popis řídicích a stavových polí

Pole	Popis
Řízení AT (ACNTL) Vstup – Byte	MSB 7 … LSB 0 0 0 0 0 0 0 0 EN EN – Nastavte na 1 pro spuštění adaptace; nastavte na 0 pro přerušení adaptace.
Stav AT (ASTAT) Výstup – Byte	MSB 7 … LSB 0 W0 W1 W2 0 AH 0 0 IP W0 – Výstraha: Nastavení odchylky není čtyřikrát větší než nastavení hystereze. W1 – Výstraha: Nekonzistentní odchylky procesu mohou způsobit nesprávné nastavení hodnoty výstupního kroku. W2 – Výstraha: Okamžitá průměrná odchylka není čtyřikrát větší než nastavení hystereze. AH – Probíhá automatický výpočet hystereze: 0 – operace neprobíhá 1 – operace probíhá IP – Probíhá adaptace: 0 – operace neprobíhá 1 – operace probíhá Po každém spuštění sekvence adaptace PLC vymaže bity výstrahy a nastaví bit ”operace probíhá”. Po dokončení adaptace PLC vymaže bit ”operace probíhá”.
Výsledek AT (ARES) Vstup/Výstup – Byte	MSB 7 … LSB 0 D | Výsledný kód D – bit ”Done”: 0 – adaptace neukončena 1 – adaptace ukončena Musí být nastaveno na 0 před spuštěním adaptace. Výsledný kód: 00 – ukončeno normálně (doporučené hodnoty adaptace k dispozici) 01 – přerušeno uživatelem 02 – přerušeno, vypršení doby watchdogu pro průchod nulou 03 – přerušeno, proces (PV) mimo rozsah 04 – přerušeno, mezní hodnota hystereze překročena 05 – přerušeno, zjištěna neplatná hodnota konfigurace 06 – přerušeno, zjištěna numerická chyba 07 – přerušeno, instrukce PID provedena s odpojeným akčním členem (smyčka v ručním režimu) 08 – přerušeno, adaptace povolena pouze pro smyčky P, PI, PD nebo PID 09 až 7F – rezervováno
Konfigurace AT (ACNFG) Vstup – Byte	MSB 7 … LSB 0 0 0 0 0 R1 R0 DS HS R1 R0 Dynamická odezva 0 0 Rychlá odezva 0 1 Střední odezva 1 0 Pomalá odezva 1 1 Velmi pomalá odezva DS – Nastavení odchylky: 0 – použít hodnotu odchylky z tabulky smyčky 1 – stanovit hodnotu odchylky automaticky HS – Nastavení hystereze: 0 – použít hodnotu hystereze z tabulky smyčky 1 – stanovit hodnotu hystereze automaticky

Požadavky

Smyčka, kterou chcete adaptovat, musí být v automatickém režimu. Výstup smyčky musí být řízen provedením instrukce PID. Adaptace se neprovede, jestliže je smyčka v ručním režimu.

Před spuštění operace adaptace musí být váš proces uveden do stabilního stavu, což znamená, že PV dosáhlo žádané hodnoty (nebo u smyčky typu P konstantního rozdílu mezi PV a žádanou hodnotou) a výstup se chaoticky nemění.

V ideálním případě musí být hodnota výstupu smyčky uprostřed rozsahu řízení při spuštění adaptace. Procedura adaptace vyvolá oscilaci v procesu prováděním malých skokových změn na výstupu smyčky. Jestliže je výstup smyčky u některého z extrémů rozsahu řízení, skokové změny prováděné v proceduře adaptace mohou způsobit, že výstupní hodnota se bude snažit překročit minimální nebo maximální limit rozsahu.

Pokud by se to mělo stát, může nastat generování chybového stavu adaptace, což bude mít jistě za následek stanovení doporučených hodnot, které nebudou ani blízko optimálním hodnotám.

Automatická hystereze a automatická odchylka

Parametr “Hystereze” specifikuje odchýlení (plus nebo minus) od žádané hodnoty, kterou může mít PV (proměnná procesu), aniž způsobí, že reléový regulátor změní výstup. Tato hodnota se používá k minimalizaci účinku šumu v signálu PV, aby se přesněji stanovila přirozená frekvence oscilace procesu.

Jestliže zvolíte, aby se hodnota hystereze stanovila automaticky, adaptační algoritmus PID spustí sekvenci stanovení hystereze. Tato sekvence zahrnuje vzorkování proměnné procesu po určitou dobu a provedení výpočtu standardní odchylky podle výsledků vzorku.

Aby se získal statisticky smysluplný vzorek, musí být získán soubor alespoň 100 vzorků. Pro smyčku se vzorkovací periodou 200 ms trvá získání 100 vzorků 20 vteřin. Pro smyčky s delší vzorkovací periodou to bude trvat déle. I když může být 100 vzorků pro smyčky se vzorkovací periodou menším než 200 ms získáno za méně než 20 vteřin, sekvence stanovení hystereze vždy získává vzorky po dobu nejméně 20 vteřin.

Jakmile byly získány všechny vzorky, vypočítá se standardní odchylka pro soubor vzorků. Hodnota hystereze je definována tak, aby byla dvojnásobkem standardní odchylky. Vypočítaná hodnota hystereze se zapíše do pole okamžité hystereze (AHYS) tabulky smyčky.



Tip

Když probíhá sekvence automatické hystereze, normální výpočet PID se neprovádí. Proto je nezbytné, aby před spuštěním sekvence adaptace byl proces ve stabilním stavu. Tím se získá lepší výsledek pro hodnotu hystereze a zajistí se, že proces nebude během sekvence automatického stanovení hystereze mimo kontrolu .

Parametr “Odchylka” specifikuje požadovaný rozkmit proměnné procesu kolem žádané hodnoty. Jestliže zvolíte automatické stanovení této hodnoty, požadovaná odchylka PV se vypočítá vynásobením hodnoty hystereze číslem 4,5. Výstup bude řízen proporcionálně, aby vyvolal tuto velikost oscilace v procesu během adaptace.

Sekvence adaptace

Sekvence adaptace začne po stanovení hodnot hystereze a odchylky. Proces adaptace začne, jakmile je použit počáteční výstupní skok na výstup smyčky.

Tato změna ve výstupní hodnotě by měla způsobit odpovídající změnu v hodnotě proměnné procesu. Když změna výstupu řídí PV mimo žádanou hodnotu až za hranici hystereze, adaptace zjistí událost průchodu nulou. Při každé události průchodu nulou adaptace řídí výstup opačným směrem.

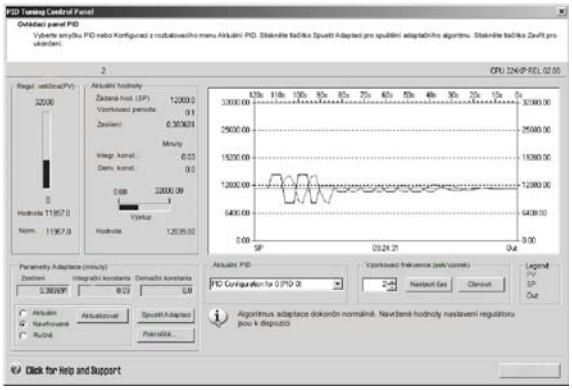
Adaptační mechanismus pokračuje ve vzorkování PV a čeká na další událost průchodu nulou. K dokončení sekvence se požaduje celkem dvanáct průchodů nulou. Velikost sledovaných hodnot PV špička-špička (špičková odchylka) a rychlost, při níž vznikají průchody nulou, přímo souvisí s dynamikou procesu.

V první fázi procesu adaptace je hodnota výstupního skoku jednou proporcionálně upravena, aby vyvolala následné rozkmitání PV a přiblížila se požadované velikosti odchylky. Po nastavení se nová velikost výstupního skoku zapíše do pole okamžité velikosti skoku (ASTEP) tabulky smyčky.

Sekvence adaptace bude ukončena chybou, jestliže doba mezi průchody nulou překročí interval watchdogu. Implicitní hodnota pro watchdog průchodů nulou je dvě hodiny.

Na obrázku 15-1 je zobrazeno chování výstupu a proměnné procesu během sekvence adaptace na přímo působící smyčce. Ovládací panel PID se použil ke spuštění a monitorování sekvence adaptace.

Všimněte si, jak adaptační algoritmus přepíná výstup, který způsobí, aby proces (jak je patrné z hodnoty PV) podstupoval malé oscilace. Frekvence a amplituda oscilací PV svědčí o zesílení a přirozené frekvenci procesu.



Obr. 15-1 Sekvence adaptace na přímo působící smyčce

Na základě informací shromážděných o frekvenci a zesílení procesu během adaptace se vypočítá mezní zesílení a mezní frekvence. Z těchto hodnot se vypočítají doporučené hodnoty pro zesílení (zesílení smyčky), reset (integrální časovou konstantu) a rate (derivační časovou konstantu).

Tip

Typ vaší smyčky určuje, které hodnoty adaptace vypočítá adaptační mechanismus. Například pro smyčku PI vypočítá adaptační mechanismus hodnoty zesílení a integrální časové konstanty, ale doporučená derivační časová konstanta bude 0,0 (žádná derivační akce).

Jakmile sekvence adaptace skončí, výstup smyčky se vrátí na počáteční hodnotu. Při příštím spuštění smyčky bude proveden normální výpočet PID.

Výjimečné stavy

Během adaptace mohou být generovány tři stavy výstrahy. Tyto výstrahy jsou oznámeny ve třech bitech pole ASTAT tabulky smyčky a jakmile jsou tyto bity nastaveny, zůstanou nastaveny až do příštího spuštění sekvence adaptace.

- Výstraha 0 vznikne, jestliže hodnota odchylky není alespoň 4x větší než hodnota hystereze. Tato kontrola se provádí, až je hodnota hystereze skutečně známa, což záleží na nastavení automatické hystereze.
- Výstraha 1 vznikne, jestliže je zjištěn více než 8násobný rozdíl mezi dvěma hodnotami chyby špičky během prvních 2,5 cyklů procedury adaptace.
- Výstraha 2 vznikne, jestliže změřená průměrná špičková odchylka není alespoň 4x větší než hodnota hystereze.

Kromě stavů výstrahy je možný vznik několika chybových stavů. V tabulce 15-3 je seznam chybových stavů spolu s popisem příčiny každé chyby.

Tabulka 15-3 Chybové podmínky během provádění adaptace

Výsledný kód (v ARES)	Podmínka
01 zrušeno uživatelem	EN bit vymazán, když probíhá adaptace
02 přerušeno z důvodu vypršení doby watchdogu pro průchod nulou	Uplynulý čas poloviny cyklu překračuje interval watchdogu pro průchod nulou
03 přerušeno kvůli procesu mimo rozsah	PV jde mimo rozsah: • během sekvence automatické hystereze nebo • dvakrát před čtvrtým průchodem nulou nebo • po čtvrtém průchodu nulou
04 přerušeno kvůli překročení maximální hodnoty hystereze	Uživatelem specifikovaná hodnota hystereze nebo automaticky stanovená hodnota hystereze > maximum
05 přerušeno z důvodu neplatné hodnoty konfigurace	Následující chyby kontroly rozsahu: • Počáteční hodnota výstupu smyčky je < 0,0 nebo > 1,0 • Uživatelsky specifikovaná hodnota odchylky je <= hodnota hystereze nebo je > maximum • Počáteční výstupní krok je <= 0,0 nebo je > maximum • Doba intervalu watchdogu pro průchod nulou je < minimum • Hodnota vzorkovací periody v tabulce smyčky je záporná
06 přerušeno z důvodu numerické chyby	Neplatné číslo pohyblivé desetinné čárky nebo dělení nulou
07 Instrukce PID byla provedena s odpojeným akčním členem (ruční režim).	Instrukce PID provedena s odpojeným akčním členem, zatímco probíhá nebo je vyžadována adaptace.
08 adaptace povolena pouze pro smyčky P, PI, PD nebo PID	Typ smyčky není P, PI, PD nebo PID.

Poznámky týkající se PV mimo rozsah (výsledný kód 3)

Adaptační algoritmus považuje proměnnou procesu za jsoucí v rozsahu, jestliže je jeho hodnota větší než 0,0 a menší než 1,0.

Jestliže se během sekvence automatické hystereze zjistí, že PV je mimo rozsah, pak je adaptace ihned přerušena s chybovým výsledkem procesu mimo rozsah.

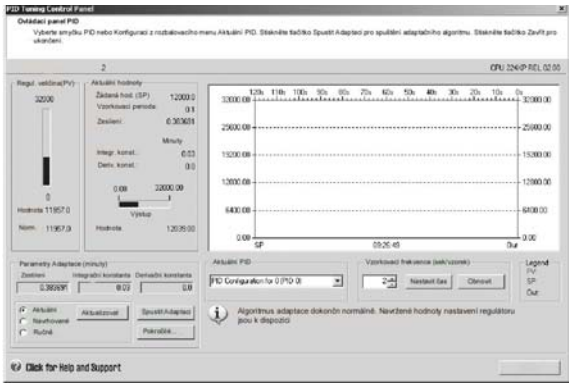
Jestliže se zjistí mezi počátečním bodem sekvence adaptace a čtvrtým nulovým průsečíkem, že PV je mimo rozsah, pak hodnota výstupního skoku je snížena na polovinu a sekvence adaptace začne opět od začátku. Jestliže vznikne druhá událost PV mimo rozsah po prvním nulovém průsečíku po restartu, pak je adaptace přerušena s chybovým výsledkem procesu mimo rozsah.

Jakákoli událost, kdy je PV mimo rozsah, která vznikne po čtvrtém nulovém průsečíku, má za následek okamžité přerušení adaptace a generování chybového výsledku mimo rozsah.

Ovládací panel PID

STEP 7–Micro/WIN obsahuje Ovládací panel PID, který vám umožňuje graficky monitorovat smyčky PID. Kromě toho vám ovládací panel umožňuje spustit sekvenci adaptace, přerušit sekvenci a použít doporučené hodnoty nastavení PID nebo vaše vlastní hodnoty nastavení PID.

Abyste mohli používat ovládací panel, musíte komunikovat s S7-200 PLC a v PLC musí existovat konfigurace smyčky PID vytvořená průvodcem. PLC musí být v režimu RUN, aby ovládací panel zobrazil operaci smyčky PID. Na obrázku 15-2 je implicitní okno pro ovládací panel.



Obr. 15-2 Ovládací panel PID

Na ovládacím panelu se zobrazuje adresa stanice (vzdálená adresa) cílového PLC v horní levé části obrazovky. V horní pravé části obrazovky se zobrazuje typ a číslo verze PLC. Pod polem se vzdálenou adresou je grafické sloupcové znázornění hodnoty regulované veličiny spolu s její normovanou i nenormovanou hodnotou. Hned vpravo od sloupcového grafu PV je oblast okamžitých hodnot.

V oblasti okamžitých hodnot jsou zobrazeny tyto hodnoty: žádaná hodnota, vzorkovací perioda, zesílení, integrační časová konstanta a derivační časová konstanta. Hodnota výstupu je zobrazena ve vodorovném sloupci spolu s číselnou hodnotou. Napravo od okamžitých hodnot je grafický displej.

Grafický displej zobrazuje barevně rozlišené grafy PV, SP a výstupu jako funkci času. PV a SP mají stejnou svislou stupnici, která je na levé straně grafu, zatímco svislá stupnice výstupu je na pravé straně grafu.

V dolní levé části obrazovky je oblast parametrů nastavení PID (minuty). Uvnitř této oblasti jsou zobrazeny hodnoty zesílení, integrační časové konstanty a derivační časové konstanty. Přepínače označují, zda jsou zobrazeny okamžité, doporučené nebo ručně nastavené hodnoty pro zesílení, integrační časovou konstantu a derivační časovou konstantu. Kliknutím na přepínač můžete zobrazit některý ze tří zdrojů pro tyto hodnoty. Chcete–li změnit parametry nastavení PID, klikněte na přepínač ručního režimu.

Tlačítko “Aktualizace PLC” můžete použít pro přenos zobrazených hodnot zesílení, integrační časové konstanty a derivační časové konstanty do PLC pro smyčku PID, která je monitorována. Tlačítko “Spustit adaptaci” můžete použít ke spuštění sekvence adaptace. Jakmile je sekvence adaptace spuštěna, tlačítko “Spustit adaptaci” se změní na “Zastavit adaptaci”.

Přímo pod grafickým displejem je oblast volby “aktuální PID” s rozbalovacím menu, které vám umožní vybrat smyčku PID, kterou chcete monitorovat ovládacím panelem.

V oblasti vzorkovací periody můžete zvolit vzorkovací periodu grafického displeje od 1 do 480 vteřin na vzorek. Vzorkovací periodu můžete editovat, pak použít tlačítko “Nastavit čas”, aby se změna použila. Časové měřítko grafu se automaticky přizpůsobí nejlepšímu zobrazení dat při nové periodě.

Graf můžete pozastavit stisknutím tlačítka “Pozastavit”. Stisknutím tlačítka “Obnovit” obnovíte vzorkování dat se zvolenou periodou. Vymazání grafu provedete volbou “Vymazat” kliknutím pravého tlačítka myši v rámci grafu.

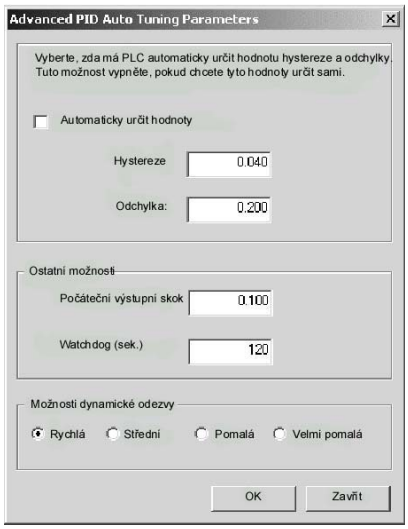
Vpravo od oblasti možnosti grafu je legenda, která identifikuje barvy použité pro grafické znázornění hodnot PV, SP a výstupu.

Přímo pod oblastí volby “Aktuální PID” je pole, které se používá k zobrazení informací týkajících se prováděné operace.

Tlačítko “Pokročilá...” v oblasti “Parametry adaptace” vám umožní další konfiguraci procesu adaptace. Okno pokročilých nastavení je zobrazeno dole na obrázku 15-3.

V okně “Pokročilá nastavení” můžete zaškrtnout políčko, které způsobí, že adaptační emchanismus automaticky určí hodnoty pro hysterezi a odchylku (implicitní nastavení), nebo můžete zadat hodnoty pro tato pole, které budou minimalizovat rušení vašeho procesu během procedury adaptace.

V oblasti “Další možnosti” můžete specifikovat počáteční velikost výstupního skoku a vypršení doby watchdogu pro průchod nulou.



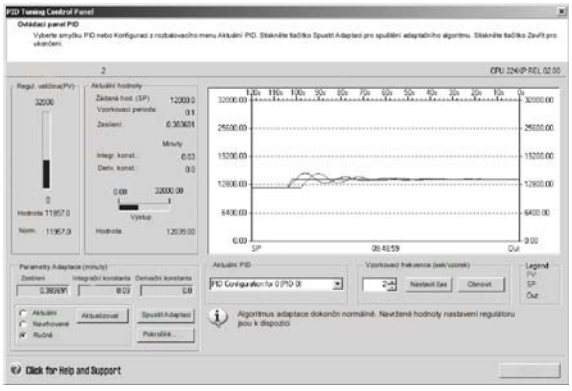
Obr. 15-3 Pokročilé parametry nastavení PID

V oblasti “Možnosti dynamické odezvy” klikněte na přepínač, který odpovídá typu odezvy smyčky, který chcete mít pro váš proces. Podle vašeho procesu může mít rychlá odezva překmit a odpovídala by nedostatečně tlumené odezvě systému. Střední rychlost odezvy může být na pokraji překmitu a odpovídala by kriticky tlumené odezvě systému. Pomalá rychlost nemusí mít žádný překmit a odpovídala by přetlumené odezvě systému. Velmi pomalá rychlost nesmí mít žádný překmit a odpovídala by silně přetlumené odezvě systému.

Jakmile jste provedli požadované volby, kliknutím na OK se vrátíte na hlavní obrazovku Ovládacího panelu PID.

Jakmile jste ukončili sekvenci adaptace a přenesli jste doporučené parametry adaptace do PLC, můžete použít ovládací panel k monitorování odezvy vaší smyčky na změnu skoku žádané hodnoty. Na obrázku 15-4 je odezva smyčky na změnu žádané hodnoty (12000 na 14000) s původními parametry nastavení (před spuštěním adaptace).

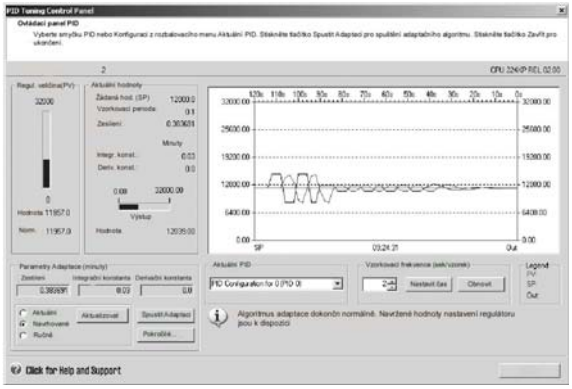
Všimněte si překmitu a dlouhého tlumeného dokmitávacího chování procesu při použití původních parametrů adaptace.



Obr. 15-4 Odezva na změnu žádané hodnoty

Na obrázku 15-5 je odezva smyčky na tutéž změnu žádané hodnoty (12000 na 14000) po použití hodnot stanovených procesem adaptace s použitím volby pro rychlou odezvu. Všimněte si, že pro tento proces není žádný překmit, ale jen malé dokmitávání. Jestliže si přejete eliminovat dokmitávání na úkor rychlosti odezvy, musíte zvolit střední nebo pomalou odezvu a znovu spustit proces adaptace.

Jakmile budete mít dobrý počáteční bod pro parametry adaptace vaší smyčky, můžete použít ovládací panel k vylepšení parametrů. Pak můžete monitorovat odezvu smyčky při změně žádané hodnoty. Tímto způsobem můžete jemně doladit váš proces na optimální odezvu pro svoji aplikaci.



Obr. 15-5 Odezva po procesu adaptace

Technické specifikace

A

Přehled kapitoly

Obecné technické specifikace 382
Specifikace CPU 385
Specifikace digitálních rozšiřovacích modulů 394
Specifikace analogových rozšiřovacích modulů 400
Specifikace termočlánkových a RTD rozšiřovacích modulů 411
Specifikace modulu EM 277 PROFIBUS–DP 423
Specifikace modemového modulu EM 241 435
Specifikace polohovacího modulu EM 253 437
Specifikace ethernetového modulu (CP 243–1) 443
Specifikace internetového modulu (CP 243–1 IT) 445
Specifikace modulu AS–Interface (CP 243–2) 448
Volitelné zásuvné moduly 450
Kabel pro rozšiřovací moduly 451
RS-232/PPI Multi-Master nebo USB/PPI Multi-Master kabel 452
Simulátory vstupu 456

Obecné technické specifikace

Normy, kterým výrobek odpovídá

Níže uvedené státní a mezinárodní normy byly použity pro stanovení příslušné specifikace vlastnostní a zkoušek pro skupinu výrobků S7-200. Tabulka A-1 definuje konkrétní zkušební organizace.

- ❑ Směrnice Evropského společenství (CE) pro nízké napětí 73/23/EEC
  EN 61131 – 2: Programovatelné automaty – Požadavky na zařízení
- ❑ Směrnice Evropského společenství (CE) pro EMC (elektromagnetickou kompatibilitu) 89/336/EEC

  Norma pro elektromagnetické emise
  EN 61000 – 6 – 3: obyvatelstvo, komerční a lehký průmysl
  EN 61000 – 6 – 4: průmyslové prostředí

  Normy elektromagnetické odolnosti
  EN 61000 – 6 – 2: průmyslové prostředí
- ❑ Underwriters Laboratories, Inc.: UL 508 (Průmyslová řídicí zařízení), Registrační číslo E75310
- ❑ Canadian Standards Association (Kanadské sdružení pro normy): CSA C22.2 číslo 142 (Zařízení pro řízení procesu)
- ❑ Factory Mutual Research: Třída číslo 3600, třída číslo 3611, FM třída I, oddělení 2, nebezpečná místa skupiny A, B, C a D, T4A a třída I, pásmo 2, IIC, T4.
- ❑ Směrnice Evropského společenství (ATEX) pro výbušná prostředí 94/9/ES
  EN 60079 – 0 Všeobecné požadavky
  EN 50020 Vnitřní bezpečnost ‘i’
  EN 60079 – 15 Typ ochrany ‘n’

  Osvědčení podle směrnice ATEX 94/9/ES bylo v době vydání této publikace neúplné. Nejnovější informace získáte u oblastního zástupce firmy Siemens.



Tip

Řada SIMATIC S7-200 splňuje normu CSA.

Logo cULus znamená, že S7-200 byl testován a certifikován laboratořemi Underwriters Laboratories (UL) podle norem UL 508 a CSA 22.2 č. 142.

Schválení pro lodní dopravu

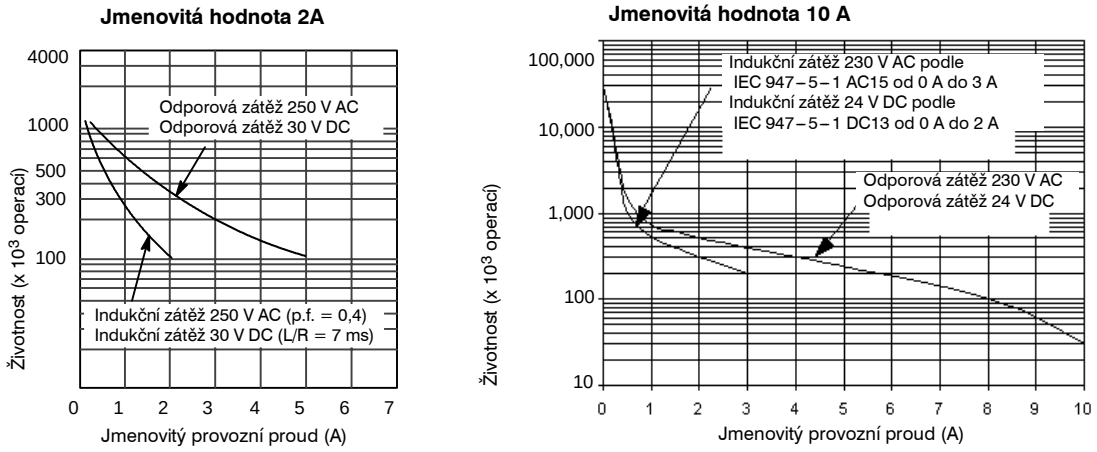
Výrobky S7-200 jsou periodicky předkládány ke schválení speciálním agenturám, v návaznosti na specifické trhy a použití. Tato tabulka uvádí agenturu a číslo jejího certifikátu, kterým byly výrobky S7-200 schváleny. Většina výrobků S7-200 uvedených v tomto manuálu byla homologována těmito speciálními úřady. Konzultujte svého místního zástupce společnosti Siemens, pokud potřebujete další informace týkající se nejnovějšího seznamu přesných schválení podle čísla dílu.

Agentura	Číslo certifikátu
Lloyds Register of Shipping (LRS)	99 / 20018(E1)
American Bureau of Shipping (ABS)	01 – HG20020 – PDA
Germanischer Lloyd (GL)	12 045 – 98 HH
Det Norske Veritas (DNV)	A – 8862
Bureau Veritas (BV)	09051 / B0BV
Nippon Kaiji Kyokai (NK)	A – 534
Polski Rejestr	TE/1246/883241/99

Elektrická životnost relé

Typické údaje o vlastnotech relé jsou uvedeny na obrázku A-1. Skutečné vlastnosti se mohou lišit v závislosti na konkrétním použití.

Externí ochranný obvod přizpůsobený zátěži prodlouží životnost kontaktů.



Obr. A-1 Elektrická životnost relé

Technické specifikace

Všechny CPU S7-200 a rozšiřovací moduly odpovídají technickým specifikacím uvedeným v tabulce A-1.

Poznámka

Po zapnutí napájení CPU S7-200 nebo kteréhokoliv digitálního rozšiřovacího modulu je vyslán ”1” signál log.1 digitálním výstupům po dobu cca 50 mikrosekund. S tím musíte počítat obzvláště tehdy, používáte–li zařízení, která reagují na krátké pulzy.

Tabulka A-1 Technická specifikace

Podmínky prostředí — Doprava a skladování	
EN 60068–2–2, zkouška Bb, suché teplo EN 60068–2–1, zkouška Ab, chlad	–40 °C až +70 °C
EN 60068–2–30, zkouška Db, vlhké teplo	25 °C až 55 °C, vlhkost 95 %
EN 60068–2–14, zkouška Na, teplotní šok	–40 °C až +70 °C, doba trvání 3 hodiny, 2 cykly
EN 60068–2–31 Převržení	100 mm, 4 pády, nebaleno
EN 60068–2–32 Volný pád	1 m, 5krát, baleno pro dopravu
Podmínky prostředí — Provoz	
Rozsah teplot okolí (Přívod vzduchu 25 mm pod přístrojem)	0 °C až 55 °C při horizontální montáži, 0 °C až 45 °C při vertikální montáži nekondenzující vlhost 95 %
Atmosferický tlak	1080 až 795 hPa (odpovídá nadmořské výšce –1000 až 2000 m)
Koncentrace znečišťujících látek	SO_2: < 0,5 ppm; H_2S: < 0,1 ppm; RH < 60% nekondenzující
EN 60068–2–14, zkouška Nb, změna teploty	5 °C až 55 °C, 3 °C/min
EN 60068–2–27 Mechanický ráz	15 G, pulz 11 ms, 6 rázů ve směru všech 3 os
EN 60068–2–6 Sinusové vibrace	Montáž na panelu: 0,30 mm od 10 do 57 Hz; 2 G od 57 do 150 Hz Montáž na DIN lištu: 0,15 mm od 10 do 57 Hz; 1 G od 57 do 150 Hz 10 vychýlení v každé ose, 1 oktáva/min
EN 60529, IP20 Mechanická ochrana	Ochrana před nebezpečným dotykovým napětím (prstem), jak bylo odzkoušeno standardními sondami. Vnější ochrana je nutná proti prachu, nečistotě, vodě a cizím předmětům o průměru < 12,5 mm.

Tabulka A-1 Technická specifikace, pokračování

Elektromagnetická kompatibilita — odolnost podle EN61000–6–2[1]	
EN 61000–4–2 Elektrostatický výboj	8 kV výboj vzduchem na všechny povrchy a komunikační port, 4 kV výboj dotykem na všechny exponované vodivé povrchy
EN 61000–4–3 Vyzařované elektromagnetické pole	10 V/m, 80–1000 MHz a 1,4 až 2,0 GHz, 80% AM při 1 kHz
EN 61000–4–4 Rychlé přechodové vysokofrekvenční pulzy	2 kV, 5 kHz s párovanou sítí k AC a DC napájení systému 2 kV, 5 kHz s párovanou svorkou ke vstupům a výstupům 1 kV, 5 kHz s párovanou svorkou ke komunikaci
EN 61000–4–5 Odolnost vůči rázům	Napájení: 2 kV asymetrické, 1 kV symetrické vstupy/výstupy 1 kV symetrické (obvody 24 V DC vyžadují externí ochranu proti rázu)
EN 61000–4–6 Poruchy z vedení	0,15 až 80 MHz, 10 V RMS, 80% amplitudová modulace při 1 kHz
EN 61000–4–11 Poklesy napětí, krátká přerušení a kolísání napětí	>95% redukce pro 8,3 ms, 83 ms, 833 ms a 4167 ms
VDE 0160 Neperiodické přepětí	Při lince 85 V AC, 90∀fázový úhel, platí špička 390 V, impulz 1,3 ms Při lince 180 V AC, 90∀fázový úhel, platí špička 750 V, impulz 1,3 ms
Elektromagnetická kompatibilita — Vedené a vyzařované emise podle EN 61000–6–3[2] a EN 61000–6–4	
EN 55011, třída A, skupina 1, vedené[1] 0,15 MHz až 0,5 MHz 0,5 MHz až 5 MHz 5 MHz až 30 MHz	 < 79 dB (σV) kvazi–špička; < 66 dB (σV) střední hodnota < 73 dB (σV) kvazi–špička; < 60 dB (σV) střední hodnota < 73 dB (σV) kvazi–špička; < 60 dB (σV) střední hodnota
EN 55011, třída A, skupina 1, vyzařované[1] 30 MHz až 230 MHz 230 MHz až 1 GHz	 40 dB (σV/m) kvazi–špička; měřeno v 10 m 47 dB (σV/m) kvazi–špička; měřeno v 10 m
EN 55011, třída B, skupina 1, vedené[2] 0,15 až 0,5 MHz 0,5 MHz až 5 MHz 5 MHz až 30 MHz	 < 66 dB (σV) klesá s log frekvencí na 56 dB (σV); < 56 dB (σV) průměrná hodnota klesá s log frekvencí na 46 dB (σV); < 56 dB (σV) kvazi–špička; < 46 dB (σV) střední hodnota < 60 dB (σV) kvazi–špička; < 50 dB (σV) střední hodnota
EN 55011, třída B, skupina 1, vyzařované[2] 30 MHz až 230 kHz 230 MHz až 1 GHz	 30 dB (σV/m) kvazi–špička; měřeno v 10 m 37 dB (σV/m) kvazi–špička; měřeno v 10 m
Vysokonapěťová izolační zkouška	
obvody 24 V/5 V obvody 115/230 V vůči zemi obvody 115/230 V vůči obvodům 115/230 V obvody 230 V vůči obvodům 24 V/5 V obvody 115 V vůči obvodům 24 V/5 V	500 V AC (hranice optické izolace) 1 500 V AC 1 500 V AC 1 500 V AC 1 500 V AC

1 Přístroj musí být instalován na uzemněném kovovém rámu; uzemnění S7-200 musí být provedeno přímo na instalační kov. Kabely jsou vedeny po kovových podpěrách.

2 Přístroj musí být instalován v uzemněném kovovém krytu. Napájení vedení AC musí být vybaveno filtrem EPCOS B84115–E–A30 nebo ekvivalentním, délka vodiče od filtru k automatu S7-200 může být maximálně 25 cm. Kabeláž napájení 24 V DC a napájení snímače musí být stíněná.

Specifikace CPU

Tabulka A-2 Objednací čísla CPU

Objednací číslo	Model CPU	Napájení (jmenovité)	Digitální vstupy	Digitální výstupy	Komunik. porty	Analogové vstupy	Analogové výstupy	Odnímatelná svorkovnice
6ES7 211–0AA23–0XB0	CPU 221	24 VDC	6 x 24 V DC	4 x 24 V DC	1	Ne	Ne	Ne
6ES7 211–0BA23–0XB0	CPU 221	120 až 240 VAC	6 x 24 V DC	4 x Relé	1	Ne	Ne	Ne
6ES7 212–1AB23–0XB0	CPU 222	24 VDC	8 x 24 V DC	6 x 24 V DC	1	Ne	Ne	Ne
6ES7 212–1BB23–0XB0	CPU 222	120 až 240 VAC	8 x 24 V DC	6 x Relé	1	Ne	Ne	Ne
6ES7 214–1AD23–0XB0	CPU 224	24 VDC	14 x 24 V DC	10 x 24 V DC	1	Ne	Ne	Ano.
6ES7 214–1BD23–0XB0	CPU 224	120 až 240 VAC	14 x 24 V DC	10 x Relé	1	Ne	Ne	Ano.
6ES7 214–1AD22–0XB0	CPU 224XP	24 VDC	14 x 24 V DC	10 x 24 V DC	2	2	1	Ano.
6ES7 214–1BD22–0XB0	CPU 224XP	120 až 240 VAC	14 x 24 V DC	10 x Relé	2	2	1	Ano.
6ES7 216–1AD22–0XB0	CPU 226	24 VDC	24 x 24 V DC	16 x 24 V DC	2	Ne	Ne	Ano.
6ES7 216–1BD22–0XB0	CPU 226	120 až 240 VAC	24 x 24 V DC	16 x Relé	2	Ne	Ne	Ano.

Tabulka A-3 Obecná specifikace CPU

Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	K dispozici	
					+5 VDC	+24 VDC[1]
6ES7 211–0AA23–0XB0	CPU 221 DC/DC/DC 6 vstupů/4 výstupů	90 x 80 x 62	270 g	3 W	0 mA	180 mA
6ES7 211–0BA23–0XB0	CPU 221 AC/DC/relé 6 vstupů/4 relé	90 x 80 x 62	310 g	6 W	0 mA	180 mA
6ES7 212–1AB23–0XB0	CPU 222 DC/DC/DC 8 vstupů/6 výstupů	90 x 80 x 62	270 g	5 W	340 mA	180 mA
6ES7 212–1BB23–0XB0	CPU 222 AC/DC/relé 8 vstupů/6 relé	90 x 80 x 62	310 g	7 W	340 mA	180 mA
6ES7 214–1AD23–0XB0	CPU 224 DC/DC/DC 14 vstupů/10 výstupů	120,5 x 80 x 62	360 g	7 W	660 mA	280 mA
6ES7 214–1BD23–0XB0	CPU 224 AC/DC/relé 14 vstupů/ 10 relé	120,5 x 80 x 62	410 g	10 W	660 mA	280 mA
6ES7 214–1AD22–0XB0	CPU 224XP DC/DC/DC 14 vstupů/10 výstupů	140 x 80 x 62	390 g	8 W	660 mA	280 mA
6ES7 214–1BD22–0XB0	CPU 224XP AC/DC/relé 14 vstupů/10 relé	140 x 80 x 62	440 g	11 W	660 mA	280 mA
6ES7 216–1AD22–0XB0	CPU 226 DC/DC/DC 24 vstupů/16 výstupů	196 x 80 x 62	550 g	11 W	1000 mA	400 mA
6ES7 216–1BD22–0XB0	CPU 226 AC/DC/relé 24 vstupů/16 relé	196 x 80 x 62	660 g	17 W	1000 mA	400 mA

[1] Je to napájení snímačů 24 VDC, které je k dispozici po zohlednění požadavků na napájení interních relé a komunikačního portu 24 VDC.

Tabulka A-4 Specifikace CPU

	CPU 221	CPU 222	CPU 224	CPU 224XP	CPU 226
Paměť					
Velikost uživatelského programu s editací v režimu RUN bez editace v režimu RUN	4096 bytů 4096 bytů		8192 bytů 12288 bytů	12288 bytů 16384 bytů	16384 bytů 24576 bytů
Uživatelská data	2048 bytů		8192 bytů	10240 bytů	10240 bytů
Záloha (velkokapacitní kondenzátor) (volitelně baterie)	běžně 50 hodin (8 hodin při 40°C) běžně 200 dnů		běžně 100 hodin (70 hodin při 40°C) běžně 200 dnů	běžně 100 hodin (70 hodin min.při 40°C) běžně 200 dnů	
Vstupy a výstupy					
Digitální vstupy/výstupy	6 vstupů/ 4 výstupy	8 vstupů/6 výstupů	14 vstupů/10 výstupů	14 vstupů/ 10 výstupů	24 vstupů/16 výstupů
Analogové vstupy/výstupy	Nemá			2 vstupy/1 výstup	nemá
Velikost obrazu digitálních I/O	256 (128 vstupů/128 výstupů)				
Velikost obrazu analogových I/O	Nemá	32 (16 vstupů/ 16 výstupů)	64 (32 vstupů/32 výstupů)		
Max. počet rozšiřovacích modulů	Nemá	2 modulů[1]	7 modulů[1]		
Max. počet inteligentních modulů	Nemá	2 modulů[1]	7 modulů[1]		
Vstupy pro zachycení pulzů	6	8	14		24
Vysokorychlostní čítače Jednofázové Dvoufázové	celkem 4 čítače 4 při 30 kHz 2 při 20 kHz		celkem 6 čítačů 6 při 30 kHz 4 při 20 kHz	celkem 6 čítačů 4 při 30 kHz 2 při 200 kHz 3 při 20 kHz 1 při 100 kHz	celkem 6 čítačů 6 při 30 kHz 4 při 20 kHz
Pulzní výstupy	2 při 20 kHz (pouze DC výstupy)			2 při 100 kHz (pouze výstupy DC)	2 při 20 kHz (pouze výstupy DC)
Obecná					
Časovače	256 časovačů celkem; 4 časovače (1 ms); 16 časovačů (10 ms); 236 časovačů (100 ms)				
Čítače	256 (zálohovaných velkokapacitním kondenzátorem nebo baterií)				
Bity vnitřní paměti Ukládány při výpadku napájení	256 (zálohovaných velkokapacitním kondenzátorem nebo baterií) 112 (uložených do EEPROM)				
Časovaná přerušení	2 s rozlišením 1 ms				
Přerušení od náběžné/sestupné hrany	4 od náběžné a 4 od sestupné				
Analogové potenciometry	1 s rozlišením 8 bitů		2 s rozlišením 8 bitů		
Rychlost Booleovských instrukcí	0,22 µs na instrukci				
Hodiny reálného času	Zásuvný modul		Integrované		
Volitelné zásuvné moduly	Paměť, baterie a hodiny reálného času		Paměťový a bateriový		
Zabudovaná komunikace					
Porty (omezený výkon)	1 port RS–485			2 porty RS–485	
Přenosové rychlosti PPI, DP/T	9,6; 19,2; 187,5 kbaud				
Přenosové rychlosti Freeport	1,2 kbaud až 115,2 kbaud				
Max. délka kabelu na segment	S napěťově odděleným repeaterem: 1000 m až do 187,5 kbaud, 1200 m až do 38,4 kbaud Bez napěťově odděleného repeateru: 50 m				
Max. počet stanic	32 na segment, 126 na síť				
Max. počet zařízení typu master	32				
Síť s rovnocennými uzly (režim PPI Master)	Ano (NETR/NETW)				
Připojení MPI	celkem 4, 2 vyhrazeny (1 pro PG a 1 pro OP)				

1 Musíte spočítat energetickou bilanci, abyste stanovili, kolik energie (nebo proudu) může CPU S7-200 poskytnout dané konfiguraci. Pokud je překročena energetická bilance CPU, možná nebudete moci připojit maximální počet modulů. Požadavky na napájení CPU a rozšiřovacích modulů najdete v příloze A a výpočet energetické bilance v příloze B.

Tabulka A-5 Specifikace napájení CPU

	DC		AC	
Příkon				
Vstupní napětí	20,4 až 28,8 V DC		85 až 264 V AC (47 až 63 Hz)	
Vstupní proud	CPU pouze při 24 V DC	Max. zatížení při 24 V DC	Pouze CPU	Max. zatížení
CPU 221	80 m A	450 mA	30/15 mA při 120/240 V AC	120/60 mA při 120/240 V AC
CPU 222	85 mA	500 mA	40/20 mA při 120/240 V AC	140/70 mA při 120/240 V AC
CPU 224	110 mA	700 mA	60/30 mA při 120/240 V AC	200/100 mA při 120/240 V AC
CPU 224XP	120 mA	900 mA	70/35 mA při 120/240 V AC	220/100 mA při 120/240 V AC
CPU 226	150 mA	1050 mA	80/40 mA při 120/240 V AC	320/160 mA při 120/240 V AC
Nárazový proud	12 A při 28,8 V DC		20 A při 264 V AC	
Napěťové oddělení (okolí od logiky)	Není odděleno		1500 V AC	
Doba uchování (ztráta napájení)	10 ms při 24 V DC		20/80 ms při 120/240 V AC	
Pojistka (nevyměňuje se)	3 A, 250 V pomalá		2 A, 250 V pomalá	
Napájení snímačů 24 V DC				
Napětí senzoru (omezený výkon)	L+ minus 5 V		20,4 až 28,8 V DC	
Proudové omezení	1,5 A špička, tepelné omezení nedestruktivní (jmenovité zatížení viz tabulku A-3).			
Zvlnění / rušení	Odvozeno od vstupního napájení		Méně než 1 V špička-špička	
Napěťové oddělení (snímač od logiky)	Není odděleno			

Tabulka A-6 Specifikace digitálních vstupů CPU

Obecná	Vstup 24 VDC (CPU 221, CPU 222, CPU 224, CPU 226)		Vstup 24 VDC (CPU 224XP)
Typ	Aktivní/pasivní (pasivní IEC typ 1)		Aktivní/pasivní (pasivní IEC typ 1, s výjimkou I0.3 až I0.5)
Jmenovité napětí	Typické 24 V DC při 4 mA		Typické 24 V DC při 4 mA
Max. trvalé dovolené napětí	30 VDC		
Rázové napětí	35 V DC po 0,5 s		
Logická 1 (min.)	15 V DC při 2,5 mA		15 VDC při 2,5 mA (I0.0 až I0.2 a I0.6 až I1.5) 4 VDC při 8 mA (I0.3 až I0.5)
Logická 0 (max.)	5 V DC při 1 mA		5 VDC při 1 mA (I0.0 až I0.2 a I0.6 až I1.5) 1 VDC při 1 mA (I0.3 až I0.5)
Zpoždění vstupu	Volitelné (0,2 až 12,8 ms)		
Připojení dvouvodičového snímače přiblížení. Povolený svodový proud (max.)	1 mA		
Napěťové oddělení (okolí od logiky) Optické (galvanické) Oddělení po skupinách	Ano 500 V AC po dobu 1 minuty Viz schéma zapojení		
Rychlost vstupních dat vysokorychlostního čítače (HSC)			
Vstupy HSC	Úroveň logická 1	Jedna fáze	Dvě fáze
Všechny HSC	15 až 30 VDC	20 kHz	10 kHz
Všechny HSC	15 až 26 VDC	30 kHz	20 kHz
HC4, HC5 pouze na CPU 224XP	} 4 VDC	200 kHz	100 kHz
Současně sepnuté vstupy	Všechny		Všechny Pouze CPU 224XP AC/DC/relé: Všechny při 55 °C s DC vstupy při 26 VDC max. Všechny při 50 °C s DC vstupy při 30 VDC max.
Délka kabelu (max.) Stíněný Nestíněný	500 m normální vstupy, 50 m vstupy HSC[1] 300 m normální vstupy		

[1] Pro vstupy HSC se doporučuje stíněná kroucená dvojlinka.

Tabulka A-7 Specifikace digitálních výstupů CPU

Obecná	Výstup 24 VDC (CPU 221, CPU 222, CPU 224, CPU 226)	Výstup 24 VDC (CPU 224XP)	Reléový výstup
Typ	Polovodičový MOSFET[1] (aktivní)		Kontakt (suchý)
Jmenovité napětí	24 VDC	24 VDC	24 V DC nebo 250 V AC
Rozsah napětí	20,4 až 28,8 V DC	5 až 28,8 VDC (Q0.0 až Q0.4) 20,4 až 28,8 VDC (Q0.5 až Q1.1)	5 až 30 V DC nebo 5 až 250 V AC
Nárazový proud (max.)	8 A po dobu 100 ms		5 A po dobu 4 s při 10% pracovním cyklu
Logická 1 (min.)	20 V DC při maximálním proudu	L+ minus 0,.4 V při max. proudu	–
Logická 0 (max.)	0,1 V DC při zatížení 10 K T		–
Jmenovitý proud na výstup (max.)	0,75 A		2,0 A
Jmenovitý proud společný (max.)	6 A	3,75 A	10 A
Svodový proud (max.)	10 σA		–
Zátěž (max.)	5 W		30 W DC; 200 W AC[3, 4]
Indukční napětí na svorkách	L+ minus 48 V DC, rozptyl 1 W		–
Odpor v sepnutém stavu (kontakt)	0,3 T běžné (73= T max.)		0,2 T (maximální u nového)
Napěťové oddělení Optické (galvanické, okolí od logiky) Logika od kontaktu Odpor (logika od kontaktu) Oddělení po skupinách	 500 V AC po dobu 1 minuty – – Viz schéma zapojení		 – 1500 V AC po dobu 1 minuty 100 MT Viz schéma zapojení
Zpoždění (max.) Vyp. na zap. (σs) Zap. na vyp. (σs) Sepnutí	 2 σs (Q0.0, Q0.1), 15 σs (všechny ostatní) 10 σs (Q0.0, Q0.1), 130 σs (všechny ostatní) –	 0,5 σs (Q0.0, Q0.1), 15 σs (všechny ostatní) 1,5 σs (Q0.0, Q0.1), 130 σs (všechny ostatní) –	 – – 10 ms
Frekvence impulzů (max.)	20 kHz[2] (Q0.0 a Q0.1)	100 kHz[2] (Q0.0 a Q0.1)	1 Hz
Mechanická životnost v cyklech	–	–	10.000.000 (bez zatížení)
Žlvotnost kontaktů	–	–	100.000 (jmenovité zatížení)
Současně zapnuté výstupy	Všechny při 55 \C (horizontální), všechny při 45 \C (vertikální)		
Paralelní spojení dvou výstupů	Ano, pouze výstupy stejné skupiny		Ne
Délka kabelu (max.) Stíněný Nestíněný	 500 m 150 m		

1 Po zapnutí napájení CPU S7-200 nebo kteréhokoliv digitálního rozšiřovacího modulu je vyslán "1" signál log.1 digitálním výstupům pod dobu cca 50 mikrosekund. S tím musíte počítat obzvláště tehdy, používáte–li zařízení, která reagují na krátké pulzy.

2 Podle typu přijímače pulzů a kabelu může přídavný vnější zatěžovací odpor (alespoň 10 % jmenovitého proudu) zlepšit kvalitu impulzového signálu a odolnost proti rušení.

3 Životnost relé se žárovkovou zátěží bude snížena o 75 %, pokud nebudou přijata opatření ke snížení nárazů při zapnutí pod jmenovitý nárazový proud výstupu.-

4 Výkonová žárovková zatížitelnost je pro jmenovité napětí. Snižte proporcionálně výkonovou zatížitelnost pro spínací napětí (např. 120 VAC – 100 W).

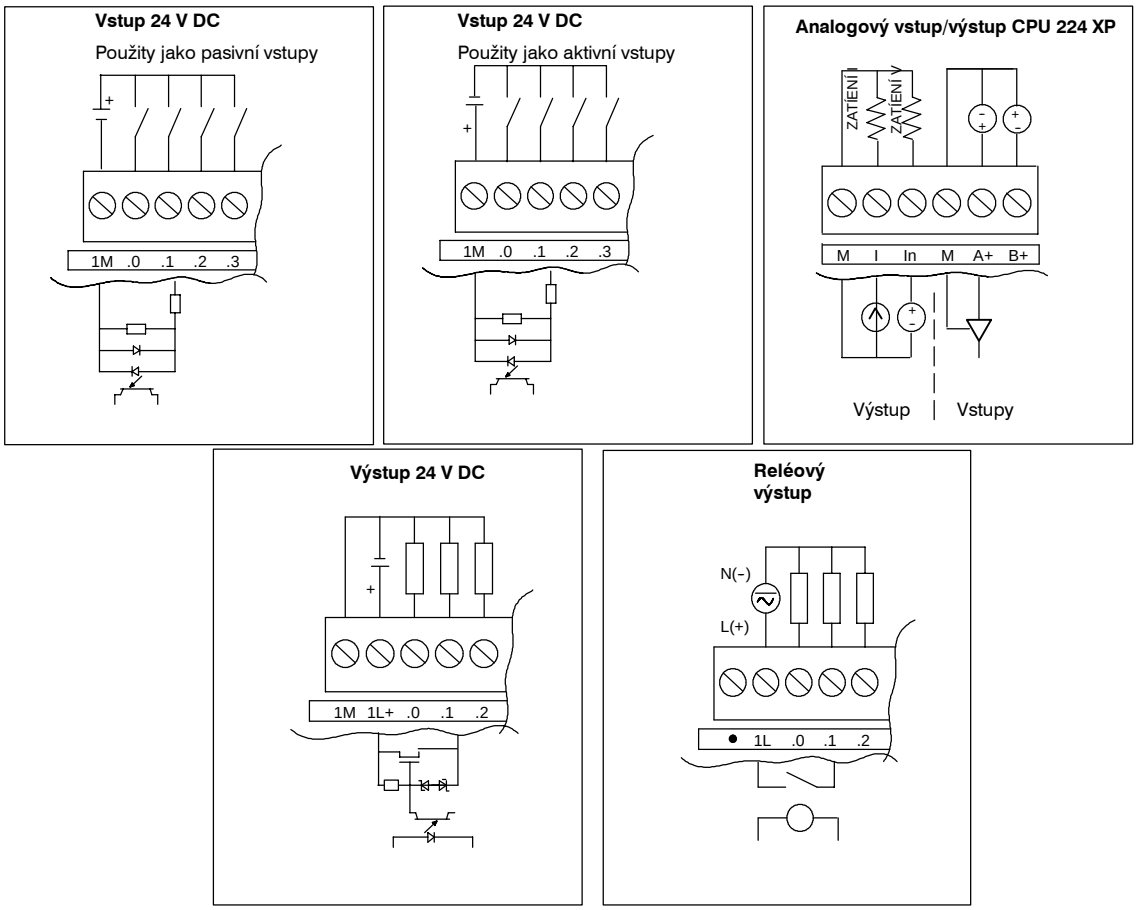
Tabulka A-8 Specifikace analogových vstupů CPU 224XP

Obecná	Analogový vstup (CPU 224XP)
Počet vstupů	2 vstupy
Typ analogového vstupu	Jednoduchý
Rozsah napětí	∂ 10 V
Formát datového wordu, rozsah celé stupnice	–32 000 až +32 000
Impedance DC vstupu	}100 KT
Maximální vstupní napětí	30 VDC
Odstranění	11 bitů plus 1 znaménkový bit
Hodnota LSB	4,88 mV
Napěťové oddělení	None (žádný)
Přesnost Nejhorší případ, 0 ∀až 55 ∀C Běžně 25 ∀C	 ∂ 2,5 % plné stupnice ∂ 1,0 % plné stupnice
Opakovatelnost	∂ 0,05 % plné stupnice
Doba konverze analogového na digitální	125 ms
Typ konverze	Sigma Delta
Skoková odezva	250 ms max.
Potlačení šumu	běžně –20 dB při 50 Hz

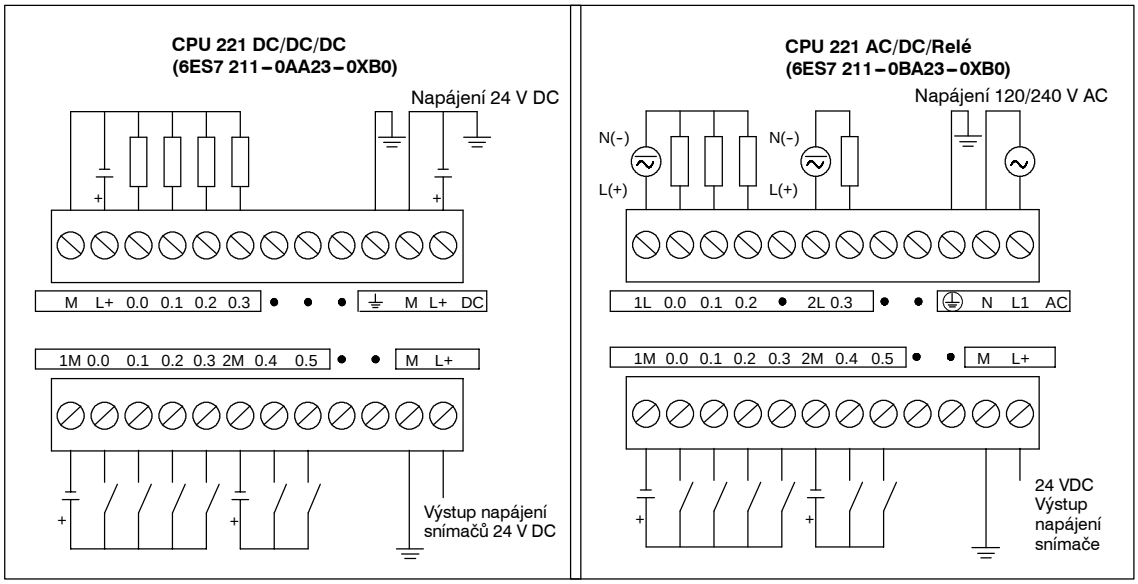
Tabulka A-9 Specifikace analogových výstupů CPU 224XP

Obecná	Analogový výstup (CPU 224XP)
Počet výstupů	1 výstup
Rozsah signálu Napětí Proud	 0 až 10 V (omezený výkon) 0 až 20 mV (omezený výkon)
Formát datového word, plný rozsah	0 až +32767
Formát datového word, plná stupnice	0 až +32000
Rozlišení, plný rozsah	12 bitů
Hodnota LSB Napětí Proud	 2,44 mV 4,88 σA
Napěťové oddělení	nemá
Přesnost Nejhorší případ, 0 ∀až 55 ∀C Napěťový výstup Proudový výstup Běžně 25 ∀C Napěťový výstup Proudový výstup	 ∂ 2 % plné stupnice ∂ 3 % plné stupnice ∂ 1 % plné stupnice ∂ 1 % plné stupnice
Doba ustálení Napěťový výstup Proudový výstup	 { 50 σS { 100 σS
Řízení maximálního výstupu Napěťový výstup Proudový výstup	 ∅5000 T minimum Ω500 T maximum

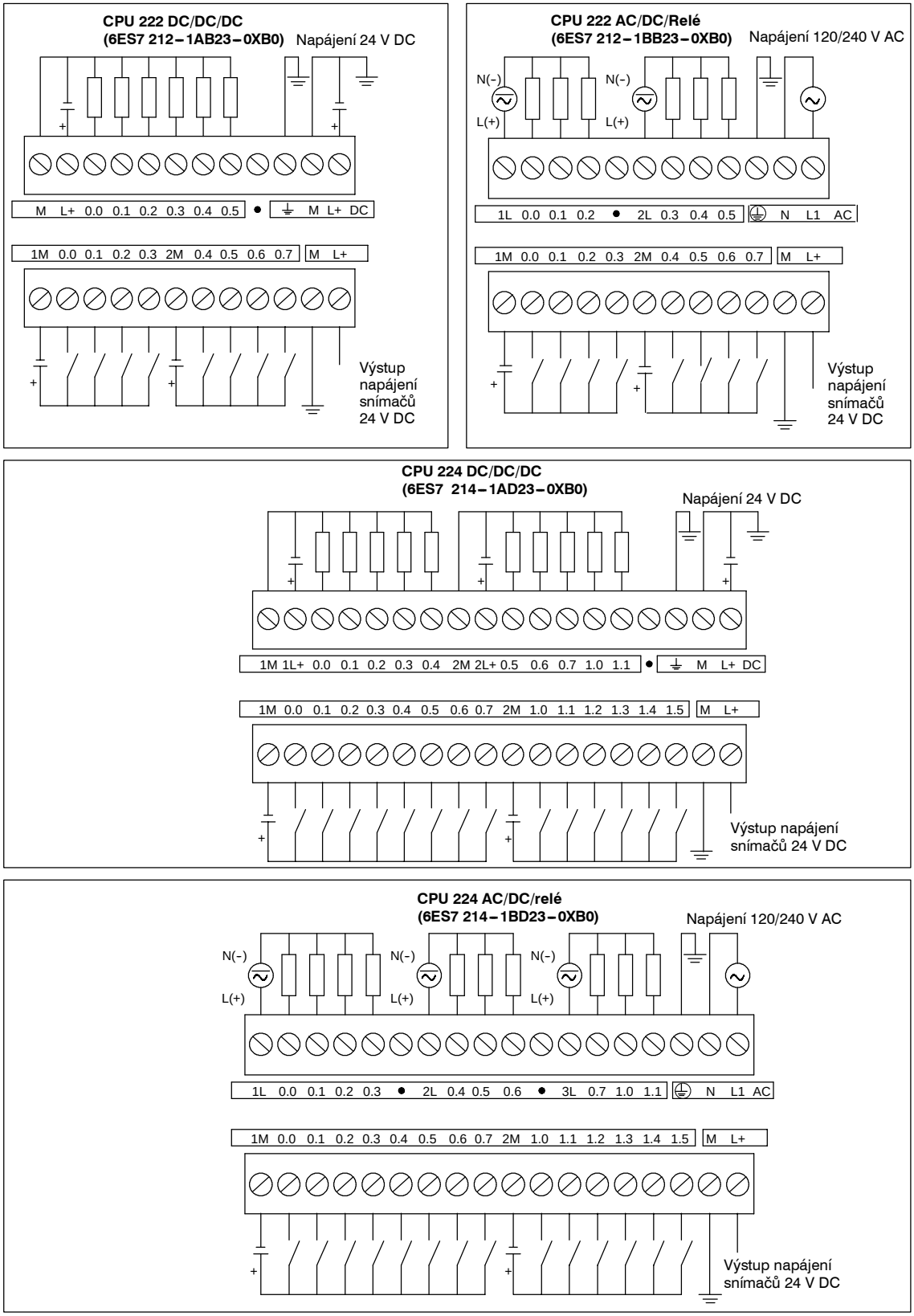
Schéma zapojení



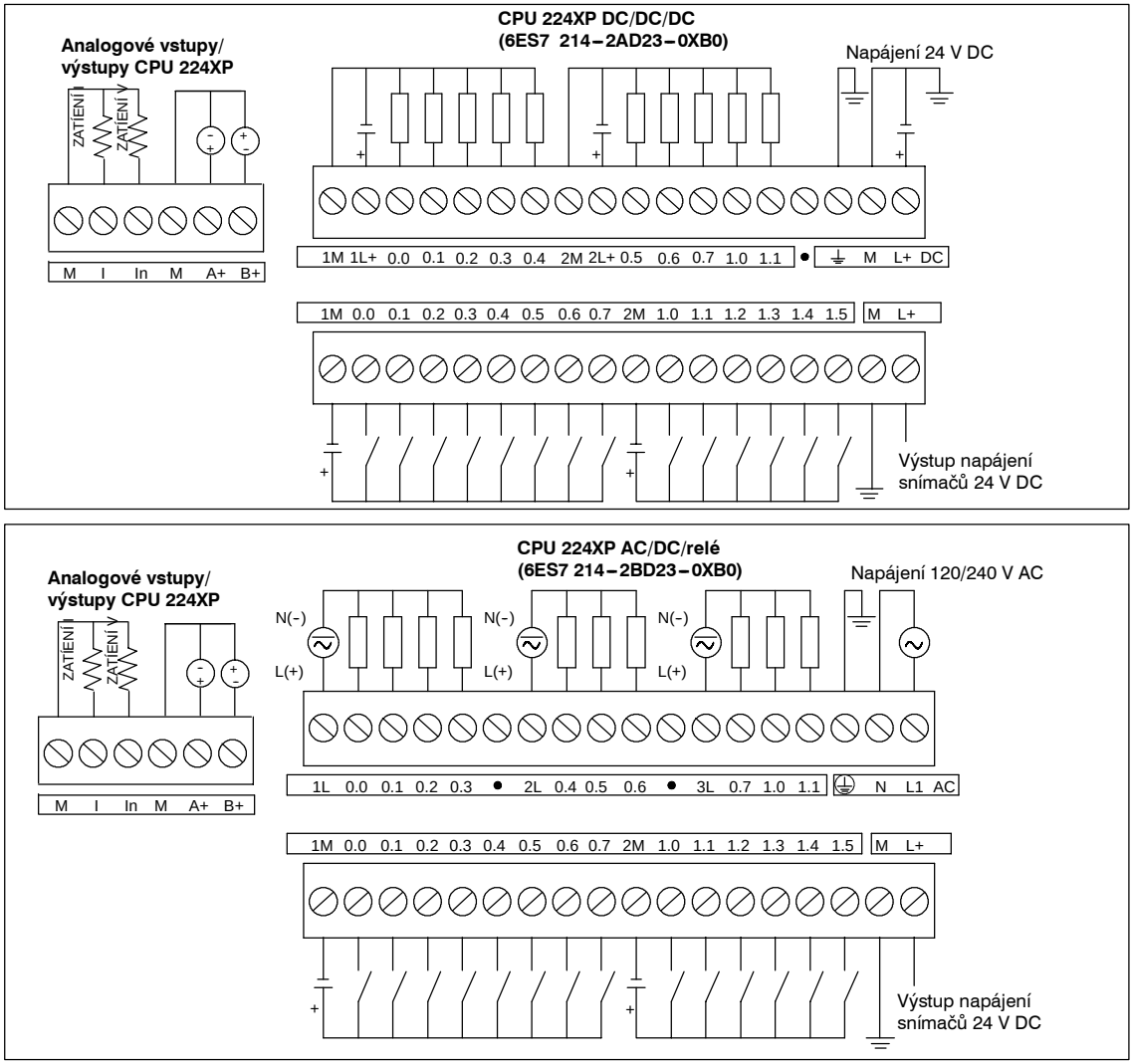
Obr. A-2 Vstupy a výstupy CPU



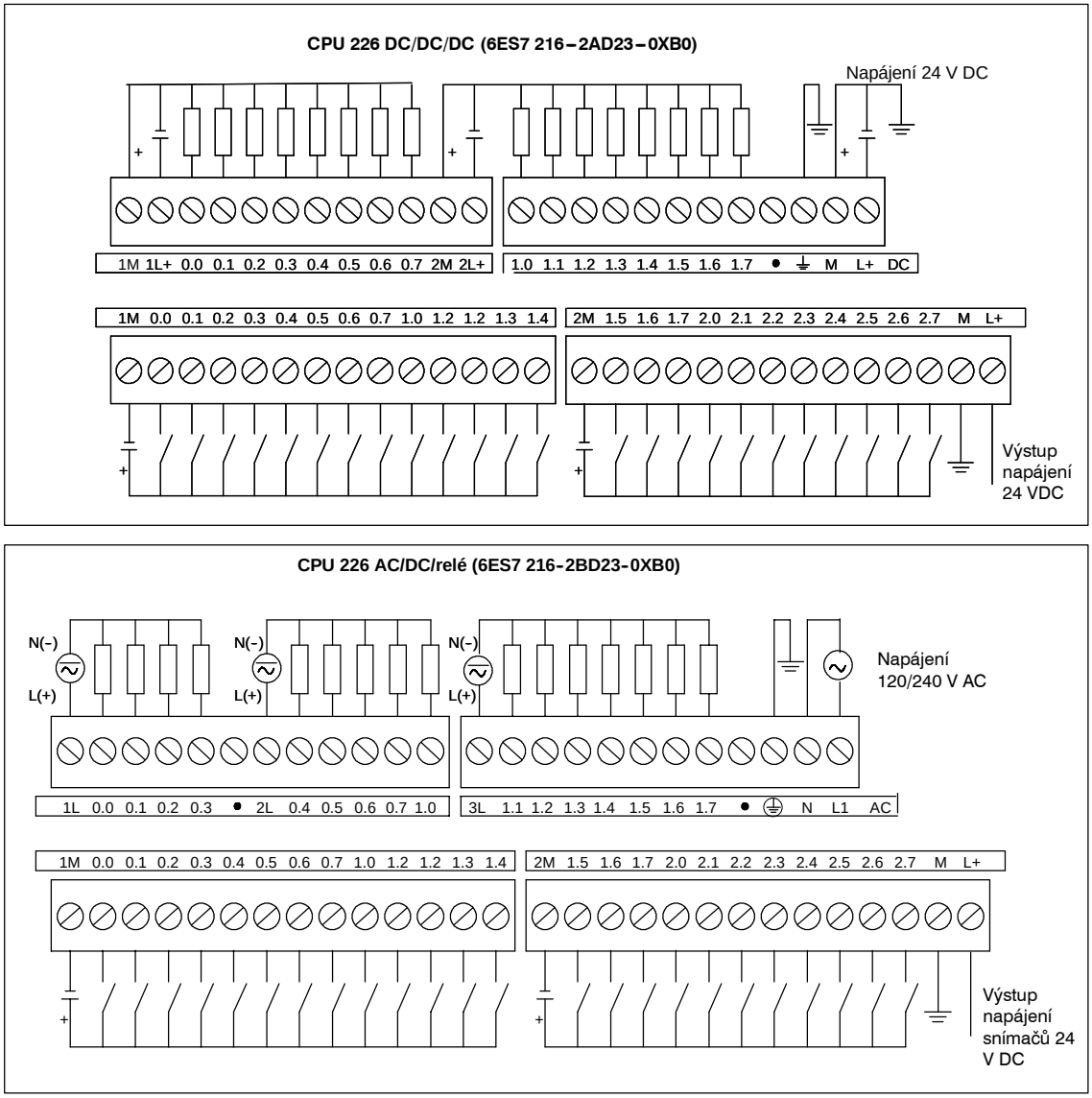
Obr. A-3 Schémata zapojení CPU 221



Obr. A-4 Schémata zapojení CPU 222 a CPU 224

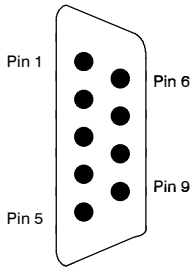


Obr. A-5 Schéma zapojení CPU 224XP



Obr. A-6 Schéma zapojení CPU 226

Tabulka A-10 Rozmístění vývodů v komunikačním portu automatu S7-200 (omezený výkon)

Konektor	Číslo vývodu	PROFIBUS Signal	Port 0/Port 1
	1	Stínění	Uzemnění rozvaděče
	2	24 V zpětný	Logický společný
	3	RS-485 signál B	RS-485 signál B
	4	Výzva k vysílání	RTS (TTL)
	5	5 V zpětný	Logický společný
	6	+5 V	+5 V, 100 T sériový odpor
	7	+24 V	+24 V
	8	RS-485 signál A	RS-485 signál A
	9	Nepoužito	10bitová volba protokolu (vstup)
	Pouzdro konektoru	Stínění	Uzemnění rozvaděče

Specifikace digitálních rozšiřovacích modulů

Tabulka A-11 Objednací čísla digitálních rozšiřovacích modulů

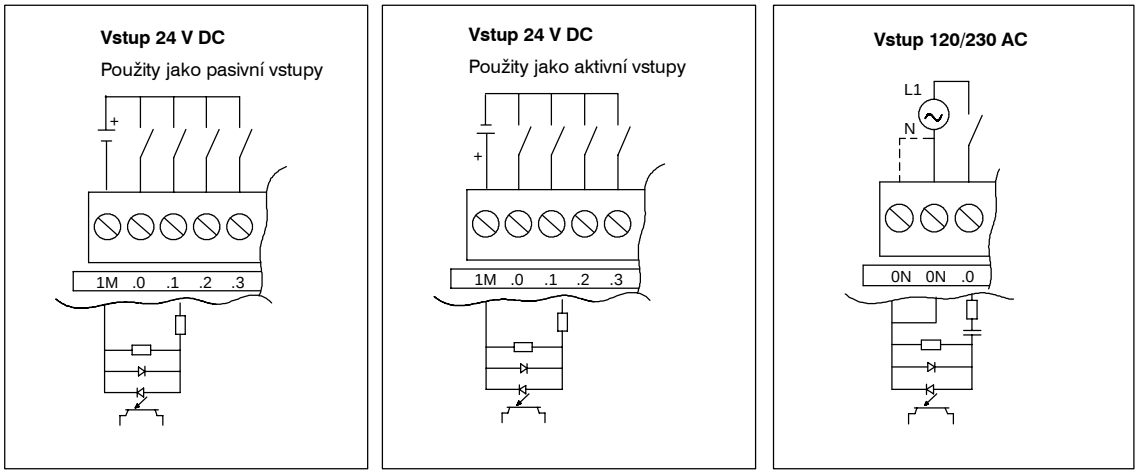
Objednací číslo	Rozšiřovací modul	Digitální vstupy	Digitální výstupy	Odnímatelná svorkovnice
6ES7 221 – 1BF22 – 0XA0	EM 221 digitální vstup 8 x 24 V DC	8 x 24 V DC	–	Ano
6ES7 221 – 1EF22 – 0XA0	EM 221 digitální vstup 8 x 120/230 V AC	8 x 120/230 VAC	–	Ano
6ES7 221 – 1BH22 – 0XA0	EM 221 digitální vstup 16 x 24 V DC	16 x 24 V DC	–	Ano
6ES7 222 – 1BD22 – 0XA0	EM 222 digitální výstup 4 x 24 V DC – 5 A	–	4 x 24 V DC – 5 A	Ano
6ES7 222 – 1HD22 – 0XA0	EM 222 digitální výstup 4 x relé – 10 A	–	4 x Relé – 10 A	Ano
6ES7 222 – 1BF22 – 0XA0	EM 222 digitální výstup 8 x 24 V DC	–	8 x 24 V DC – 0,75 A	Ano
6ES7 222 – 1HF22 – 0XA0	EM 222 digitální výstup 8 x relé	–	8 x Relé – 10 A	Ano
6ES7 222 – 1EF22 – 0XA0	EM 222 digitální výstup 8 x 120/230 V AC	–	8 x 120/230 VAC	Ano
6ES7 223 – 1BF22 – 0XA0	EM 223 24 V DC digitální kombinace 4 vstupů/4 výstupů	4 x 24 V DC	4 x 24 V DC – 0,75 A	Ano
6ES7 223 – 1HF22 – 0XA0	EM 223 24 V DC digitální kombinace 4 vstupů/4 reléových výstupů	4 x 24 V DC	4 x Relé – 2 A	Ano
6ES7 223 – 1BH22 – 0AX0	EM 223 24 V DC digitální kombinace 8 vstupů/8 výstupů	8 x 24 V DC	8 x 24 V DC – 0,75 A	Ano
6ES7 223 – 1PH22 – 0XA0	EM 223 24 V DC digitální kombinace 8 vstupů/8 reléových výstupů	8 x 24 V DC	8 x Relé – 2 A	Ano
6ES7 223 – 1BL22 – 0XA0	EM 223 24 V DC digitální kombinace 16 vstupů/16 výstupů	16 x 24 V DC	16 x 24 V DC – 0,75 A	Ano
6ES7 223 – 1PL22 – 0XA0	EM 223 24 V DC digitální kombinace 16 vstupů/16 reléových výstupů	16 x 24 V DC	16 x Relé – 2 A	Ano

Tabulka A-12 Obecná specifikace digitálních rozšiřovacích modulů

Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	Požadavky V DC	
					+5 V DC	+24 V DC
6ES7 221 – 1BF22 – 0XA0	EM 221 DI 8 x 24 V DC	46 x 80 x 62	150 g	2 W	30 mA	Zapnuto: 4 mA/vstup,
6ES7 221 – 1EF22 – 0XA0	EM 221 DI 8 x 120/230 VAC	71,2 x 80 x 62	160 g	3 W	30 mA	–
6ES7 221 – 1BH22 – 0XA0	EM 221 DI 16 x 24 V DC	71,2 x 80 x 62	160 g	3 W	70 mA	Zapnuto: 4 mA/vstup,
6ES7 222 – 1BD22 – 0XA0	EM 222 DO 4 x 24 V DC – 5 A	46 x 80 x 62	120 g	3 W	40 mA	–
6ES7 222 – 1HD22 – 0XA0	EM 222 DO 4 x relé – 10 A	46 x 80 x 62	150 g	4 W	30 mA	Zapnuto: 20 mA/výstup
6ES7 222 – 1BF22 – 0XA0	EM 222 DO 8 x 24 V DC	46 x 80 x 62	150 g	2 W	50 mA	–
6ES7 222 – 1HF22 – 0XA0	EM 222 DO 8 x relé	46 x 80 x 62	170 g	2 W	40 mA	Zapnuto: 9 mA/výstup
6ES7 222 – 1EF22 – 0XA0	EM 222 DO 8 x 120/230 VAC	71,2 x 80 x 62	165 g	4 W	110 mA	–
6ES7 223 – 1BF22 – 0XA0	EM 223 24 V DC 4 vstupů/ 4 výstupů	46 x 80 x 62	160 g	2 W	40 mA	Zapnuto: 4 mA/vstup,
6ES7 223 – 1HF22 – 0XA0	EM 223 24 V DC 4 vstupů/4 relé	46 x 80 x 62	170 g	2 W	40 mA	Zapnuto: 9 mA/výstup, 4 mA/vstup
6ES7 223 – 1BH22 – 0AX0	EM 223 24 V DC 8 vstupů/8 výstupů	71,2 x 80 x 62	200 g	3 W	80 m A	–
6ES7 223 – 1PH22 – 0XA0	EM 223 24 V DC 8 vstupů/8 relé	71,2 x 80 x 62	300 g	3 W	80 m A	Zapnuto: 9 mA/výstup, 4 mA/vstup
6ES7 223 – 1BL22 – 0XA0	EM 223 24 V DC 16 vstupů/16 výstupů	137,3 x 80 x 62	360 g	6 W	160 mA	–
6ES7 223 – 1PL22 – 0XA0	EM 223 24 V DC 16 vstupů/16 relé	137,3 x 80 x 62	400 g	6 W	150 mA	Zapnuto: 9 mA/výstup, 4 mA/vstup

Tabulka A-13 Specifikace vstupů digitálních rozšiřovacích modulů

Obecná	Vstup 24 V DC	Vstup 120/230 V AC (47 až 63 Hz)
Typ	Aktivní/pasivní (pasivní IEC typ 1)	IEC typ I
Jmenovité napětí	24 V DC při 4 mA	120 V AC při 6 mA nebo 230 V AC při 9 mA jmenovitých
Maximální trvalé dovolené napětí	30 VDC	264 V AC
Rázové napětí (max.)	35 V DC po 0,5 s	–
Logická 1 (min.)	15 V DC při 2,5 mA	79 V AC při 2,5 mA
Logická 0 (max.)	5 V DC při 1 mA	20 V AC nebo 1 mA AC
Zpoždění na vstupu (max.)	4,5 ms	15 ms
Připojení dvouvodičového snímače přiblížení. Povolený svodový proud (max.)	 1 mA	 1 mA AC
Napěťové oddělení Optické (galvanické, okolí od logiky) Oddělení po skupinách	 500 V AC po dobu 1 minuty Viz schéma zapojení	 1500 V AC po dobu 1 minuty 1 bod
Současně sepnuté vstupy	Všechny při 55∀C (horizontální), všechny zapnuty při 45∀C (vertikální)	
Délka kabelu (max.) Stíněný Nestíněný	 500 m 300 m	 500 m 300 m



Obr. A-7 Vstupy digitálních rozšiřovacích modulů automatu S7-200

Tabulka A-14 Specifikace výstupů digitálních rozšiřovacích modulů

Obecná	Výstup 24 V DC		Reléový výstup		Výstup 120/230 V AC
	0,75 A	5 A	2 A	10 A	
Typ	Polovodičový-MOSFET[1] (aktivní)		Kontakt (suchý)		Triak, "zero–cross" spínací [2]
Jmenovité napětí	24 VDC		24 V DC nebo 250 V AC		120/230 V AC
Rozsah napětí	20,4 až 28,8 V DC		5 až 30 V DC nebo 5 až 250 V AC	12 až 30 V DC nebo 12 až 250 V AC	40 až 264 VAC (47 až 63 Hz)
Rozsah napětí cívky 24 V DC	–		20,4 až 28,8 V DC		–
Nárazový proud (max.)	8 A po dobu 100 ms	30 A	5 A po dobu 4 s při 10% pracovním cyklu	15 A po dobu 4 s při 10% pracovním cyklu	efektivní hodnota 5 A po dobu 2 AC cyklů
Logická 1 (min.)	20 VDC		–		L1 (–0,9 V efektivní hodnota)
Logická 0 (max.)	0,1 V DC při zatížení 10 KT	0,2 V DC při zatížení 5 KT	–		–
Jmenovitý proud na výstup (max.)	0,75 A	5 A	2,00 A	10 A odporová; 2 A DC indukční; 3 A AC indukční	0,5 A AC[3]
Jmenovitý proud společný (max.)	6 A	5 A	8 A	10 A	0,5 A AC
Svodový proud (max.)	10 σA	30 σA	–		1,1 mA efektivní při 132 V AC a 1,8 mA efektivní při 264 V AC
Zátěž (max.)	5 W	50 W	30 W DC/ 200 W AC[6, 7]	100 W DC/ 1000 W AC	60 W
Indukční napětí na svorkách	L+ minus 48 V	L+ minus 47 V[4]	–		–
Odpor v sepnutém stavu (kontakt)	0,3 T běžný (0,6 T max.)	0,05 T max.	0,2 T maximální u nového	0,1 T max. u nového	410 T max. při zátěžovém proudu menším než 0,05 A
Napěťové oddělení					
Optické (galvanické, okolí od logiky)	500 V AC po dobu 1 minuty		–		1500 V AC po dobu 1 minuty
Cívka od logiky	–		Nemá		–
Cívka od kontaktu	–		1500 V AC po dobu 1 minuty		–
Odpor (cívka vůči kontaktu)	–		100 MT min. v novém stavu		–
Oddělení po skupinách	Viz schéma zapojení		Viz schéma zapojení		1 výstup
Zpoždění vypnutí, zapnutí/zapnutí, vypnutí (max.)	50 σs /200 σs	500 σs	–	–	0,2 ms + 1/2 AC cyklu
Sepnutí (max.)	–	–	10 ms	15 ms	–
Spínací kmitočet (max.)	–		1 Hz		10 Hz
Mechanická životnost v cyklech	–		10 000 000 (bez zatížení)	30 000 000 (bez zatížení)	–
Životnost kontaktů	–		100 000 (jmenovité zatížení)	30 000 (jmenovité zatížení)	–
Současně sepnuté výstupy	Všechny při 55 ∀C (horizontální), všechny při 45 ∀C (vertikální)			Všechny při 55 ∀C (horizontální) s max. proudem modulu 20 A Všechny při 45 ∀C (vertikální s max. proudem modulu 20 A[5] Všechny při 40 ∀C (horizontální) a 10 A na výstup	Všechny při 55∀C (horizontální), všechny při 45∀C (vertikální)
Paralelní spojení dvou výstupů	Ano, pouze výstupy stejné skupiny		Ne		Ne
Délka kabelu (max.)					
Stíněný	500 m		500 m		500 m
Nestíněný	150 m		150 m		150 m

1 Po zapnutí napájení CPU S7-200 nebo kteréhokoliv digitálního rozšiřovacího modulu je vyslán "1" signál log.1 digitálním výstupům pod dobu cca 50 mikrosekund. S tím musíte počítat obzvláště tehdy, používáte–li zařízení, která reagují na krátké pulzy.

2 Po připojení napájení na AC rozšiřovací modul, vysílá signál "1" na AC výstupy po dobu přibližně 1/2 AC periody. S tím je třeba počítat.

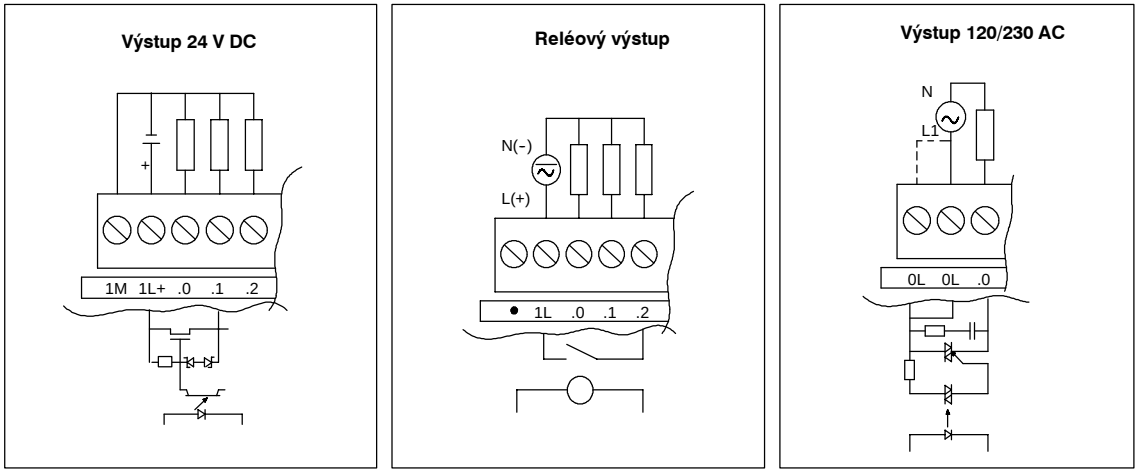
3 Zátěžový proud musí mít celou periodu AC, nesmí být pouze půlperioda, kvůli obvodům "zero–cross". Minimální zátěžový proud je 0,05 A AC. Při zatěžovacím proudu mezi 5 mA a 50 mA AC je možné proud regulovat, ale dojde k dalšímu poklesu napětí kvůli sériovému odporu 410 ohmů.

4 Jestliže se výstup přehřeje kvůli nadměrným indukčním vypínacím proudům nebo mimořádným podmínkám, může se vypnout nebo poškodit. Výstup se může přehřát nebo poškodit, je–li vystaven více než 0,7 J energie při vypnutí indukční zátěže. Aby se vyloučila potřeba takového omezení, je možné paralelně se zátěží přidat přepěťovou ochranu tak, jak je to popsáno v kapitole 3. Tyto komponenty musí být správně dimenzovány pro danou aplikaci.

5 EM 222 DO 4 x relé má jiné zařazení FM než ostatní S7-200. Tento modul má klasifikaci T4 místo T4A pro FM třídu I, skupiny oddělení A, B, C a D pro nebezpečná místa.

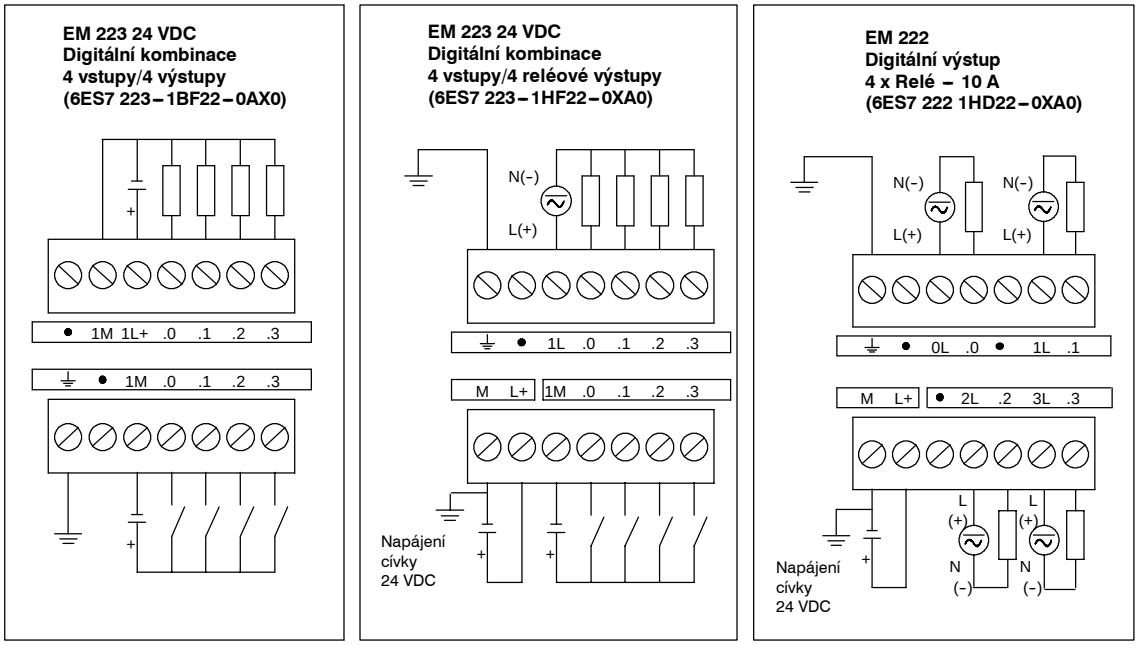
6 Životnost relé se žárovkovou zátěží bude snížena o 75 %, pokud nebudou přijata opatření ke snížení nárazů při zapnutí pod jmenovitý nárazový proud výstupu.

7 Výkonová žárovková zatížitelnost je pro jmenovité napětí. Snižte proporcionálně výkonovou zatížitelnost pro spínací napětí (např. 120 VAC – 100 W).

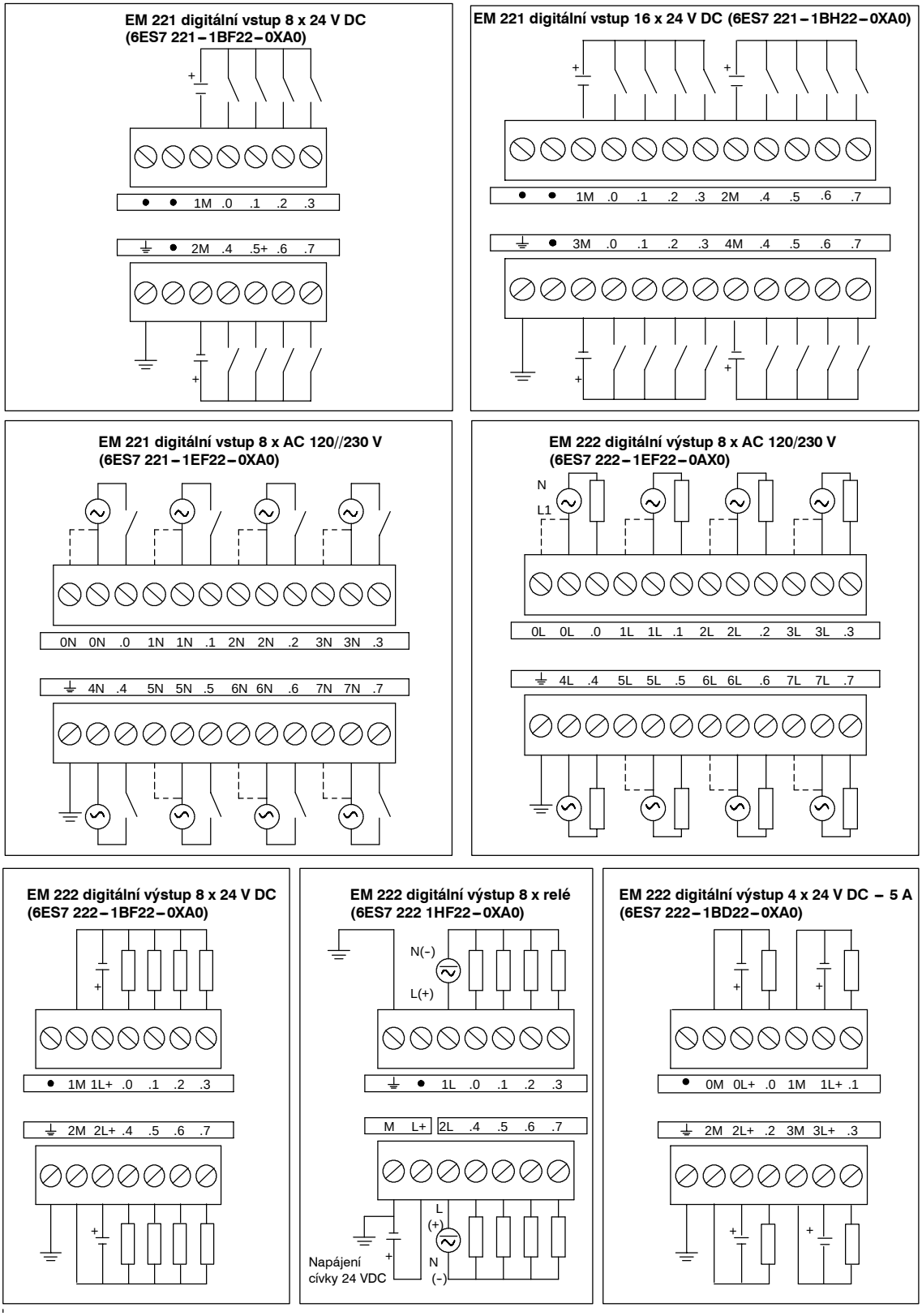


Obr. A-8 Výstupy digitálních rozšiřovacích modulů automatu S7-200

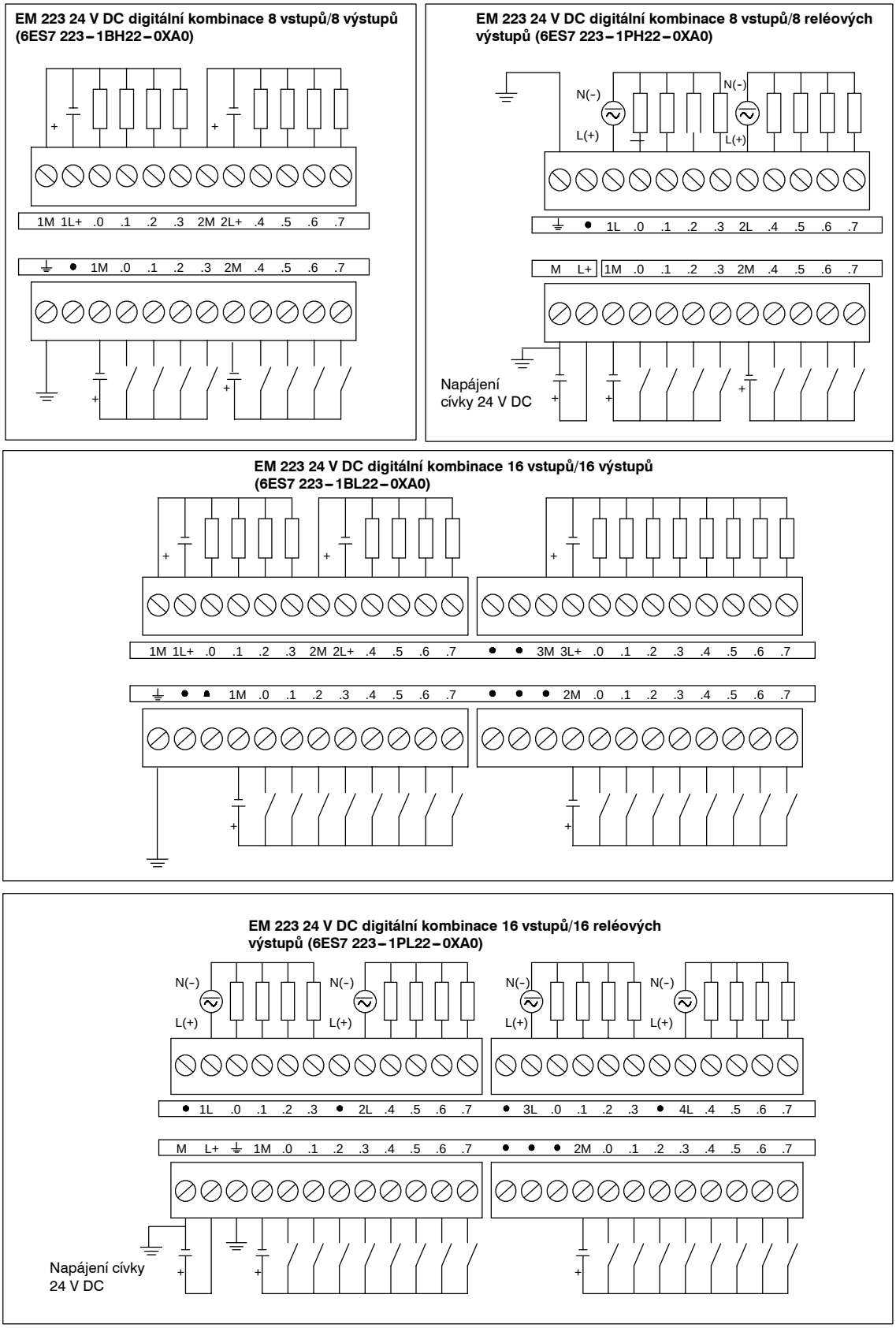
Schéma zapojení



Obr. A-9 Schéma zapojení pro rozšiřovací moduly EM 222 a EM 223



Obr. A-10 Schéma zapojení pro rozšiřovací moduly EM 221 a EM 222



Obr. A-11 Schéma zapojení pro rozšiřovací moduly EM 223

Specifikace analogových rozšiřovacích modulů

Tabulka A-15 Objednací čísla analogových rozšiřovacích modulů

Objednací číslo	Rozšiřovací modul	Vstupy EM	Výstupy EM	Odnímatelná svorkovnice
6ES7 231–0HC22–0XA0	EM 231 analogový vstup, 4 vstupy	4	–	Ne
6ES7 232–0HB22–0XA0	EM 232 analogový výstup, 2 výstupy	–	2	Ne
6ES7 235–0KD22–0XA0	EM 235 analogová kombinace, 4 vstupy/1 výstup	4	1[1]	Ne

1 CPU rezervuje pro tento modul 2 analogové výstupy.

Tabulka A-16 Obecná specifikace analogových rozšiřovacích modulů

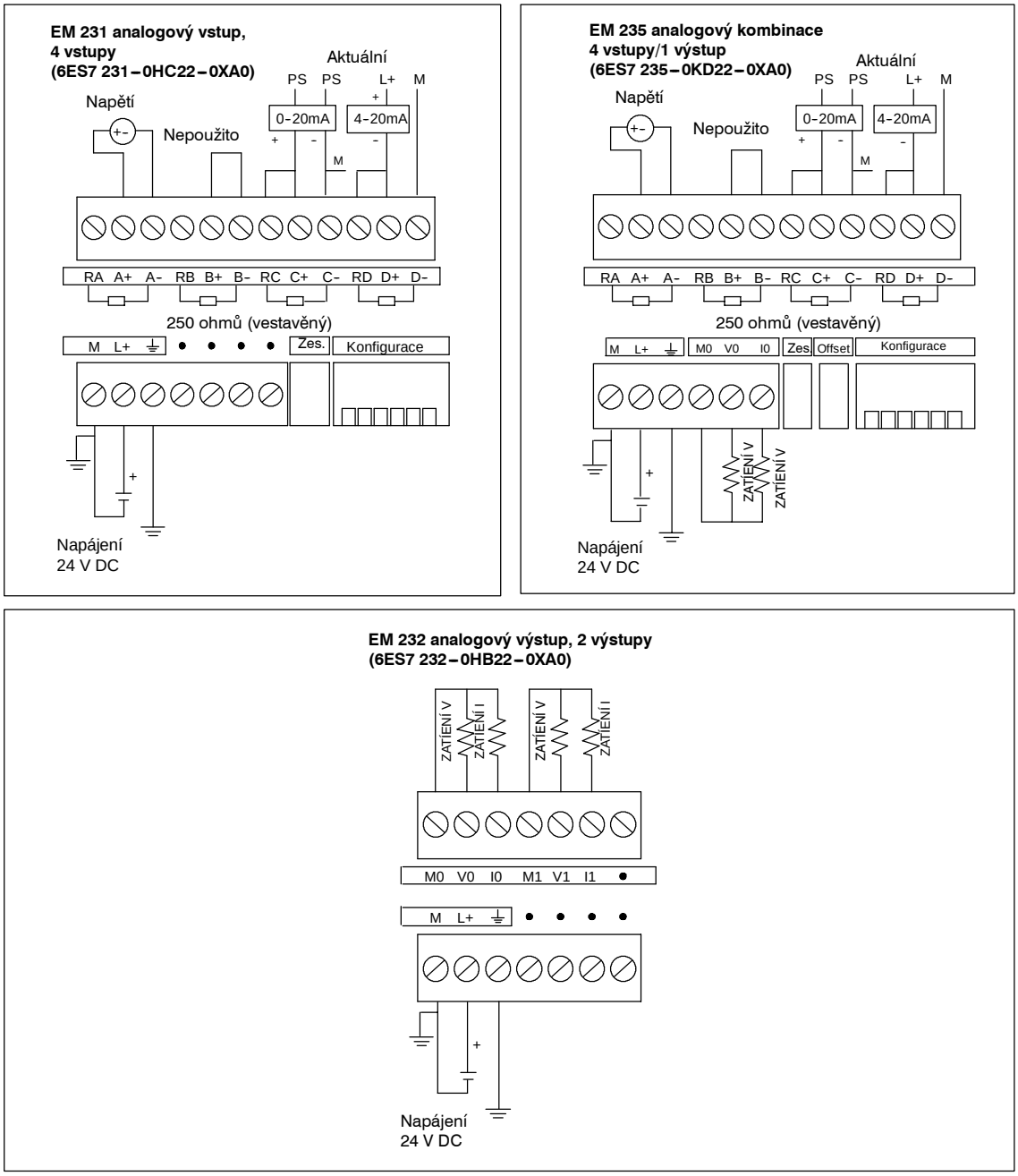
Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	Požadavky V DC +5 V DC	Požadavky V DC +24 V DC
6ES7 231–0HC22–0XA0	EM 231 analogový vstup, 4 vstupy	71,2 x 80 x 62	183 g	2 W	20 mA	60 mA
6ES7 232–0HB22–0XA0	EM 232 analogový výstup, 2 výstupy	46 x 80 x 62	148 g	2 W	20 mA	70 mA (s oběma výstupy na 20 mA)
6ES7 235–0KD22–0XA0	EM 235 analogová kombinace 4 vstupy/1 výstup	71,2 x 80 x 62	186 g	2 W	30 mA	60 mA (s výstupem na 20 mA)

Tabulka A-17 Specifikace vstupů analogových rozšiřovacích modulů

Obecná	6ES7 231-0HC22-0XA0	6ES7 235-0KD22-0XA0
Formát datového word	(Viz obrázekA-14)	(Viz obrázekA-14)
Bipolární, rozsah celé stupnice	–32000 až +32000	–32000 až +32000
Unipolární, rozsah celé stupnice	0 až 32000	0 až 32000
Impedance DC vstupu	Ø10 MT napěťový vstup 250 T proudový vstup	Ø10 MT napěťový vstup 250 T proudový vstup
Útlum vstupního filtru	–3 dB při 3,1 kHz	–3 dB při 3,1 kHz
Maximální vstupní napětí	30 VDC	30 VDC
Maximální vstupní proud	32 mA	32 mA
Odstranění Bipolární Unipolární	 11 bitů plus 1 znaménkový bit 12 bitů	
Napěťové oddělení (okolí od logiky)	Nemá	Nemá
Typ vstupu	Diferenční	Diferenční
Vstupní rozsahy Napětí Proud	 Volitelné, dostupné rozsahy viz tabulka A-20 0 až 20 mA	 Volitelné, dostupné rozsahy viz tabulka A-21 0 až 20 mA
Vstupní rozlišení	Viz tabulka A-20	Viz tabulka A-21
Doba konverze analogového na digitální	<250 os	<250 os
Skoková odezva analogového vstupu	1,5 ms na 95 %	1,5 ms na 95 %
Potlačení souhlasného signálu	40 dB, DC na 60 Hz	40 dB, DC na 60 Hz
Souhlasné napětí	Signálové napětí plus souhlasné napětí musí být Ω12 V	Signálové napětí plus souhlasné napětí musí být Ω12 V
Rozsah napětí napájení 24 V DC	20,4 až 28,8 V DC (třída 2, omezený výkon nebo napájení snímačů z PLC)	

Tabulka A-18 Specifikace výstupů analogových rozšiřovacích modulů

Obecná	6ES7 232−0HB22−0XA0	6ES7 235−0KD22−0XA0
Napěťové oddělení (okolí od logiky)	Nemá	Nemá
Rozsah signálu		
Napěťový výstup	∂ 10 V	∂ 10 V
Proudový výstup	0 až 20 mA	0 až 20 mA
Rozlišení při plné výchylce		
Napětí	12 bitů plus 1 znaménkový bit	11 bitů plus 1 znaménkový bit
Proud	11 bitů	11 bitů
Formát datového word		
Napětí	−32000 až +32000	−32000 až +32000
Proud	0 až +32000	0 až +32000
Přesnost		
Nejhorší případ, 0 ∀až 55 ∀C		
Napěťový výstup	∂ 2 % plné stupnice	∂ 2 % plné stupnice
Proudový výstup	∂ 2 % plné stupnice	∂ 2 % plné stupnice
Běžně, 25 ∀C		
Napěťový výstup	∂ 0,5 % plné stupnice	∂ 0,5 % plné stupnice
Proudový výstup	∂ 0,5 % plné stupnice	∂ 0,5 % plné stupnice
Doba nastavení		
Napěťový výstup	100 σs	100 σs
Proudový výstup	2 ms	2 ms
Maximální buzení		
Napěťový výstup	5000 T minimálně	5000 T minimálně
Proudový výstup	500 T maximální	500 T maximální
Rozsah napětí napájení 24 V DC	20,4 až 28,8 V DC (třída 2, omezený výkon nebo napájení snímačů z PLC)	



Obr. A-12 Schémata zapojení analogových rozšiřovacích modulů

Signálky analogových modulů

Signálky analogových modulů jsou uvedeny v tabulce A-19.

Tabulka A-19 Signálky analogových modulů

LED	ZAPNUTO	VYPNUTO
Napájení 24 VDC v pořádku	Bez poruch	Chybí napájení 24 V DC



Tip

Stav uživatelského napájení také hlásí speciální paměťové bity (SM). Další informace najdete v příloze D, SMB8 až SMB21 ID vstupního/výstupního modulu a chybový registr.

Kalibrace vstupů

Nastavení kalibrace ovlivňuje zesílení přístrojového zesilovače, který se nachází za analogovým multiplexerem (viz blokové schéma vstupů pro EM 231 na obrázku A-15 a EM 235 na obrázku A-16). Proto ovlivňuje kalibrace všechny uživatelské vstupní kanály. Dokonce i po kalibraci způsobí odchylky hodnot komponent každého vstupního obvodu, který se nachází před analogovým multiplexorem, drobné rozdíly v odečtech kanálů připojených na stejný vstupní signál.

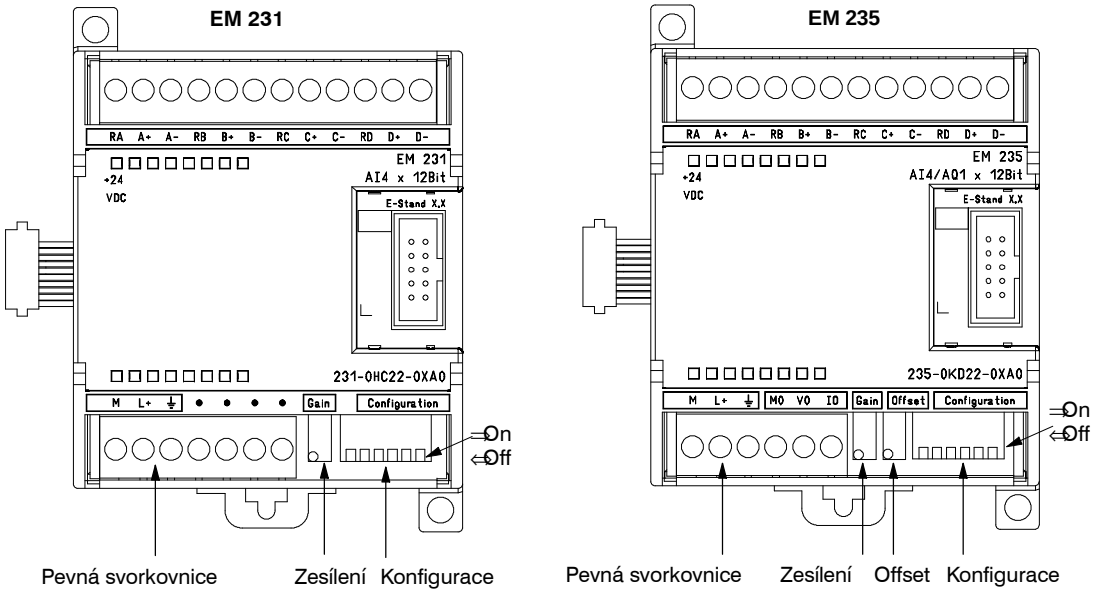
Aby hodnoty odpovídaly specifikaci, musíte aktivovat analogové vstupní filtry pro všechny vstupy modulu. Vyberte 64 nebo více vzorků a vypočítejte průměrnou hodnotu.

Chcete – li kalibrovat vstup, proveďte následující kroky.

1. Vypněte napájení modulu. Vyberte požadovaný vstupní rozsah.
2. Zapněte napájení CPU a modulu. Nechejte modul stabilizovat po dobu 15 minut.
3. Pomocí vysílače, zdroje napětí nebo zdroje proudu připojte nulový signál na jeden ze vstupních konektorů.
4. Přečtěte hodnotu poslanou CPU příslušným vstupním kanálem.
5. Otáčejte potenciometrem OFFSET (posunutí), dokud se odečet nerovná nule nebo požadované hodnotě číselného údaje.
6. Na jeden ze vstupních konektorů připojte signál s maximální hodnotou. Přečtěte hodotu poslanou CPU.
7. Otáčejte potenciometrem GAIN (zesílení), dokud není odečet 32000 nebo požadovaná hodnota.
8. Kalibraci OFFSET a GAIN opakujte podle potřeby.

Umístění kalibračních potenciometrů u EM 231 a EM 235

Obrázek A-13 ukazuje kalibrační potenciometr a konfigurační DIP přepínače umístěné vpravo od dolní svorkovnice modulu.



Obr. A-13 Umístění kalibračního potenciometru a konfiguračního DIP přepínače u EM 231 a EM 235

Konfigurace EM 231

Tabulka A-20 ukazuje, jak konfigurovat modul EM 231pomocí konfiguračních DIP přepínačů. Přepínače 1, 2 a 3 volí rozsah analogového vstupu. Všechny vstupy se nastavují na stejný rozsah analogového vstupu. V této tabulce znamená ON sepnutý, OFF rozepnutý. Nastavení přepínačů je možné číst pouze při zapnutém napájení.

Tabulka A-20 EM 231 Tabulka konfiguračních přepínačů pro volbu rozsahu analogového vstupu

Unipolární			Plný rozsah vstupu	Odstranění
SW1	SW2	SW3		
ON	OFF	ON	0 až 10 V	2,5 mV
	ON	OFF	0 až 5 V	1,25 mV
			0 až 20 mA	5 σA
Bipolární			**Plný rozsah vstupu**	**Odstranění**
SW1	**SW2**	**SW3**		
OFF	OFF	ON	∂ 5 V	2,5 mV
	ON	OFF	∂ 2,5 V	1,25 mV

Konfigurace EM 235

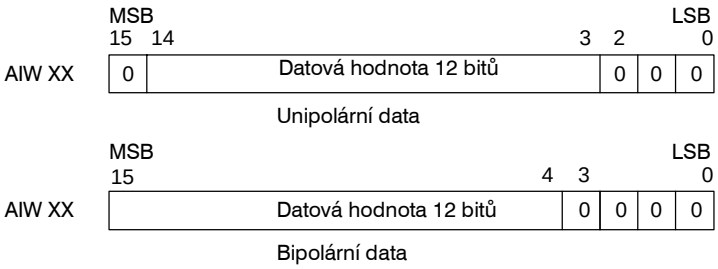
Tabulka A-21 ukazuje, jak nakonfigurovat modul EM 235 pomocí konfiguračních DIP přepínačů. Přepínače 1 až 6 volí rozsah a rozlišení analogového vstupu. Všechny vstupy se nastavují na stejný rozsah a formát analogového vstupu. Tabulka A-21 ukazuje, jak volit typ signálu unipolární/ bipolární (přepínač 6), zesílení (přepínače 4 a 5) a útlum (přepínače 1, 2 a 3). V těchto tabulkách znamená ON sepnutý, OFF znamená rozepnutý. Nastavení přepínačů je možné číst pouze při zapnutém napájení.

Tabulka A-21 Tabulka konfiguračních přepínačů modulu EM 235 pro volbu typu signálu, zesílení a útlumu

Unipolární						Plný rozsah vstupu	Odstranění
SW1	**SW2**	**SW3**	**SW4**	**SW5**	**SW6**		
ON	OFF	OFF	ON	OFF	ON	0 až 50 mV	12.5 μV
OFF	ON	OFF	ON	OFF	ON	0 až 100 mV	25 μV
ON	OFF	OFF	OFF	ON	ON	0 až 500 mV	125 μV
OFF	ON	OFF	OFF	ON	ON	0 až 1 V	250 μV
ON	OFF	OFF	OFF	OFF	ON	0 až 5 V	1,25 mV
ON	OFF	OFF	OFF	OFF	ON	0 až 20 mA	5 μA
OFF	ON	OFF	OFF	OFF	ON	0 až 10 V	2,5 mV
Bipolární						**Plný rozsah vstupu**	**Odstranění**
SW1	**SW2**	**SW3**	**SW4**	**SW5**	**SW6**		
ON	OFF	OFF	ON	OFF	OFF	± 25 mV	12.5 μV
OFF	ON	OFF	ON	OFF	OFF	± 50 mV	25 μV
OFF	OFF	ON	ON	OFF	OFF	± 100 mV	50 μV
ON	OFF	OFF	OFF	ON	OFF	± 250 mV	125 μV
OFF	ON	OFF	OFF	ON	OFF	± 500 mV	250 μV
OFF	OFF	ON	OFF	ON	OFF	± 1 V	500 μV
ON	OFF	OFF	OFF	OFF	OFF	± 2,5 V	1,25 mV
OFF	ON	OFF	OFF	OFF	OFF	± 5 V	2,5 mV
OFF	OFF	ON	OFF	OFF	OFF	± 10 V	5 mV

Formát vstupního datového word pro EM 231 a EM 235

Obrázek A-14 ukazuje, kde je 12bitová hodnota dat umístěna v analogovém vstupním word CPU.



Obr. A-14 Formát vstupního datového word pro EM 231 a EM 235

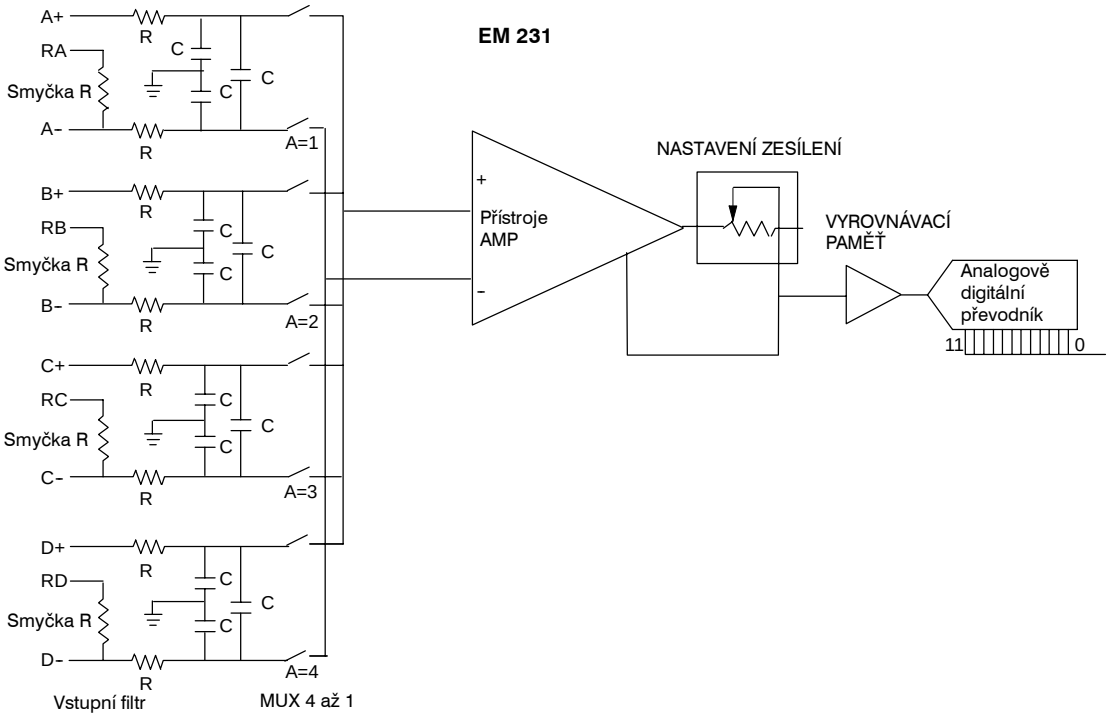
Tip

12 bitů načtených z analogově číslicového převodníku je v datovém formátu word zarovnaného vlevo. MSB je znaménkový bit: nula znamená kladnou hodnotu datového word.

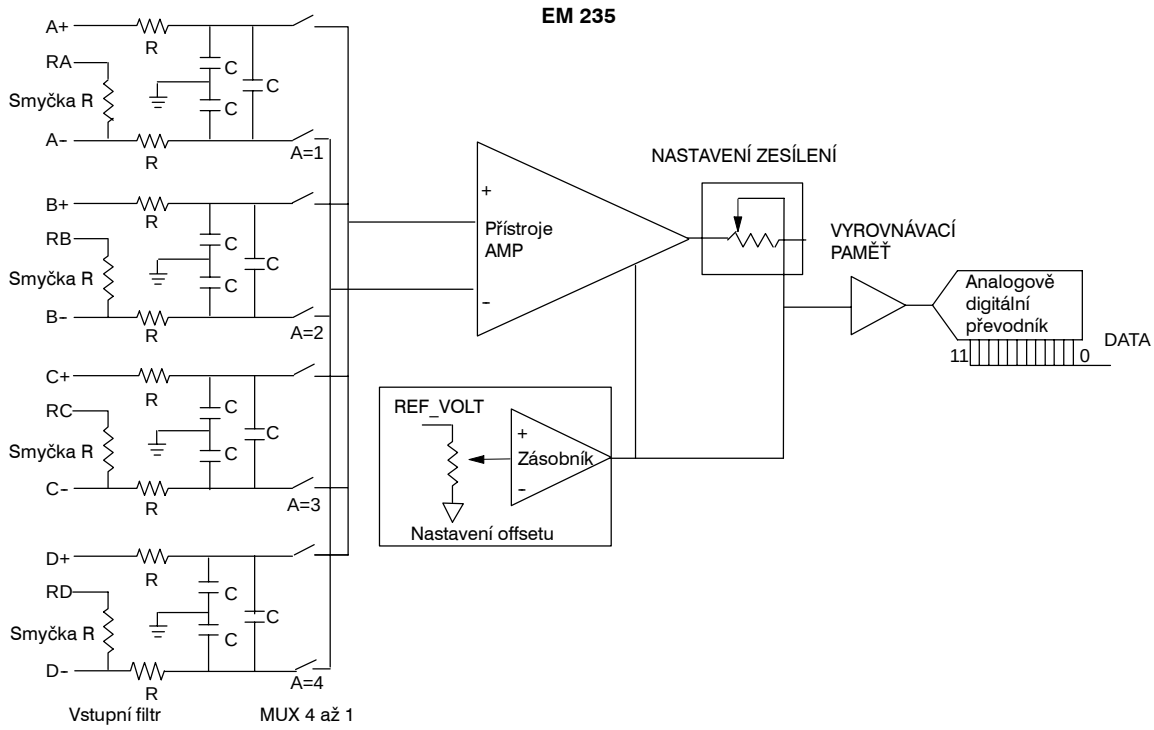
V unipolárním formátu mají tři za sebou jdoucí nuly za následek změnu datového word o osm pro každou jednotkovou změnu v hodnotě analogově číslicového převodníku.

V bipolárním formátu mají čtyři za sebou jdoucí nuly za následek změnu datového word o šestnáct pro každou jednotkovou změnu v hodnotě analogově číslicového převodníku.

Blokové schéma vstupů EM 231 a EM 235



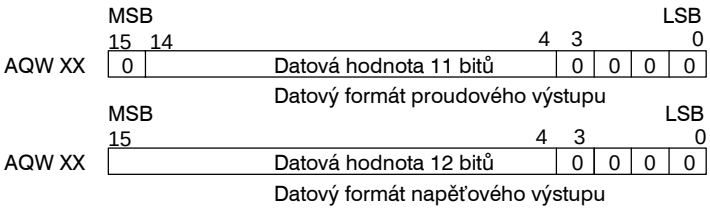
Obr. A-15 Blokové schéma vstupů EM 231



Obr. A-16 Blokové schéma vstupů EM 235

Formát výstupního datového word pro EM 232 a EM 235

Obrázek A-17 ukazuje, kde je 12bitová hodnota dat umístěna v analogovém výstupním word CPU.

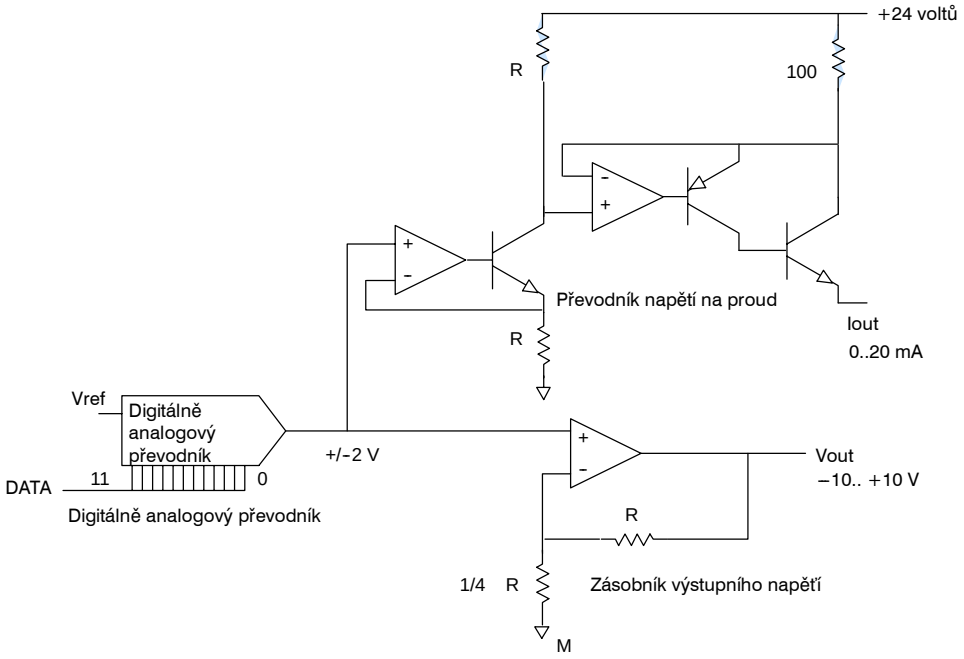


Obr. A-17 Formát výstupního datového word pro EM 232 a EM 235

Tip

12 bitů načtených z číslicově analogového převodníku je v datovém formátu word zarovnáno vlevo. MSB je znaménkový bit: nula znamená kladnou hodnotu datového word. Čtyři za sebou jdoucí nuly se před nahráním do registrů DAC vypustí. Tyto bity nemají žádný vliv na hodnotu výstupního signálu.

Blokové schéma výstupů EM 232 a EM 235



Obr. A-18 Blokové schéma výstupů EM 232 a EM 235

Návod pro instalaci

Abyste zajistili přesnost a opakovatelnost, postupujte podle následujícího návodu:

- ❏ Zajistěte, aby napájení snímačů 24 VDC bylo bez šumu a stabilizované.
- ❏ Pro připojení snímače použijte co možná nejkratší kabely.
- ❏ Pro kabeláž snímače použijte stíněnou kroucenou dvojlinku.
- ❏ Stínění se zakončuje pouze na straně snímače.
- ❏ Pro nepoužité kanály zkratujte vstupy podle obrázku A-18.
- ❏ Vyhýbejte se ohýbání vodičů do ostrých úhlů.
- ❏ Vodiče veďte kabelovými kanály.
- ❏ Vyhýbejte se paralelnímu vedení signálních vodičů se silovými vodiči. Pokud se dva vodiče musí křížit, umístěte je kolmo na sebe.
- ❏ Oddělením vstupních signálů nebo zavedením analogového referenčního společného vodiče 24 V zajistěte, aby byly vstupní signály ve specifikovaném rozsahu souhlasného napětí.

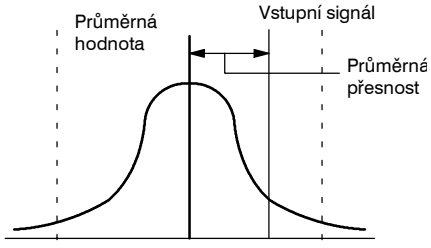
Tip
Nedoporučuje se používat rozšiřovací moduly EM 231 a EM 235 s termočlánky.

Vysvětlení analogového vstupního modulu: Přesnost a opakovatelnost

Moduly EM 231 a EM 235 jsou levné a rychlé 12bitové analogové vstupní moduly. Dokáží převést analogový vstupní signál na odpovídající digitální hodnotu za 149 σsec. Vstupní analogový signál se převádí pokaždé, když uživatelský program přistupuje k analogovému vstupu. Tyto doby převodu se musí přičíst k základní době provádění instrukce použité pro přístup k analogovému vstupu.

Moduly EM 231 a EM 235 poskytují nezpracovanou číselnou hodnotu (bez linearizace nebo filtrování), která odpovídá analogovému napětí nebo proudu na vstupních svorkách modulu. Protože jsou moduly vysokorychlostní, mohou sledovat rychlé změny analogového vstupního signálu (včetně interního a externího šumu).

Rozdíly mezi dvěma hodnotami vzorků zapříčiněné šumem v případě konstantního nebo pomalu se měnícího analogového vstupního signálu je možné minimalizovat vypočítáním průměru z určitého počtu vzorků. Uvědomte si, že zvýšení počtu vzorků použitých při výpočtu průměrné hodnoty má za následek úměrně pomalejší dobu odezvy na změny vstupního signálu.



Obr. A-19 Definice přesnosti

Obrázek A-19 ukazuje 99 % hranice opakovatelnosti, střední neboli průměrnou hodnotu jednotlivých vzorků a průměrnou přesnost v grafické formě.

Specifikace opakovatelnosti popisuje odchylky mezi jednotlivými vzorky modulu pro vstupní signál, který se nemění. Specifikace opakovatelnosti definuje hranice, mezi kterými se pohybuje 99 % všech vzorků. Opakovatelnost je na tomto obrázku znázorněna křivkou ve tvaru zvonu.

Specifikace průměrné přesnosti popisuje průměrnou hodnotu chyby (rozdílu mezi průměrnou hodnotou jednotlivých odečtů a přesnou hodnotou skutečného analogového vstupního signálu).

Tabulka A-22 uvádí specifikaci opakovatelnosti a střední přesnost ve vztahu ke každému z konfigurovatelných rozsahů.

Definice vlastností

- ❏ Přesnost: odchylka od předpokládané hodnoty v daném bodě.
- ❏ Rozlišení: výsledek změny LSB, jak se odráží na výstupu.

Tabulka A-22 Specifikace EM 231 a EM 235

Plný rozsah vstupu	Opakovatelnost[1]		Průměrná přesnost[1,2,3,4]	
	% plného rozsahu	Počet	% plného rozsahu	Počet
Specifikace EM 231				
0 až 5 V	∂ 0,075 %	∂ 24	∂ 0,1 %	∂ 32
0 až 20 mA				
0 až 10 V				
∂ 2,5 V		∂ 48	∂ 0,05 %	
∂ 5 V				
Specifikace EM 235				
0 až 50 mV	∂ 0,075 %	∂ 24	∂ 0,25 %	∂ 80
0 až 100 mV			∂ 0,2 %	∂ 64
0 až 500 mV			∂ 0,05 %	∂ 16
0 až 1 V				
0 až 5 V				
0 až 20 mA				
0 až 10 V				
∂ 25 mV	∂ 0,075 %	∂ 48	∂ 0,25 %	∂ 160
∂ 50 mV			∂ 0,2 %	∂ 128
∂ 100 mV			∂ 0,1 %	∂ 64
∂ 250 mV			∂ 0,05 %	∂ 32
∂ 500 mV				
∂ 1 V				
∂ 2,5 V				
∂ 5 V				
∂ 10 V				

1 Měření byla provedena po kalibraci zvoleného vstupního rozsahu.
2 Chyba posunutí u signálu blízkého nulovému analogovému vstupu není korigována a není zahrnuta do specifikace přesnosti.
3 Při přechodu z kanálu na kanál dochází k chybě převodu způsobené konečnou dobou ustálení analogového multiplexoru. Maximální chyba při přechodu je 0,1 % rozdílu mezi kanály.
4 Průměrná přesnost zahrnuje účinky nelinearity a driftu při 0 až 55 ˚C.

Specifikace termočlánkových a RTD rozšiřovacích modulů

Tabulka A-23 Objednací čísla termočlánkových a RTD modulů

Objednací číslo	Rozšiřovací modul	Vstupy EM	Výstupy EM	Odnímatelná svorkovnice
6ES7 231 – 7PD22 – 0XA0	EM 231 analogový vstup pro termočlánky, 4 vstupy	4 termočlánky	–	Ne
6ES7 231 – 7PB22 – 0XA0	EM 231 analogový vstup RTD, 2 vstupy	2 RTD	–	Ne

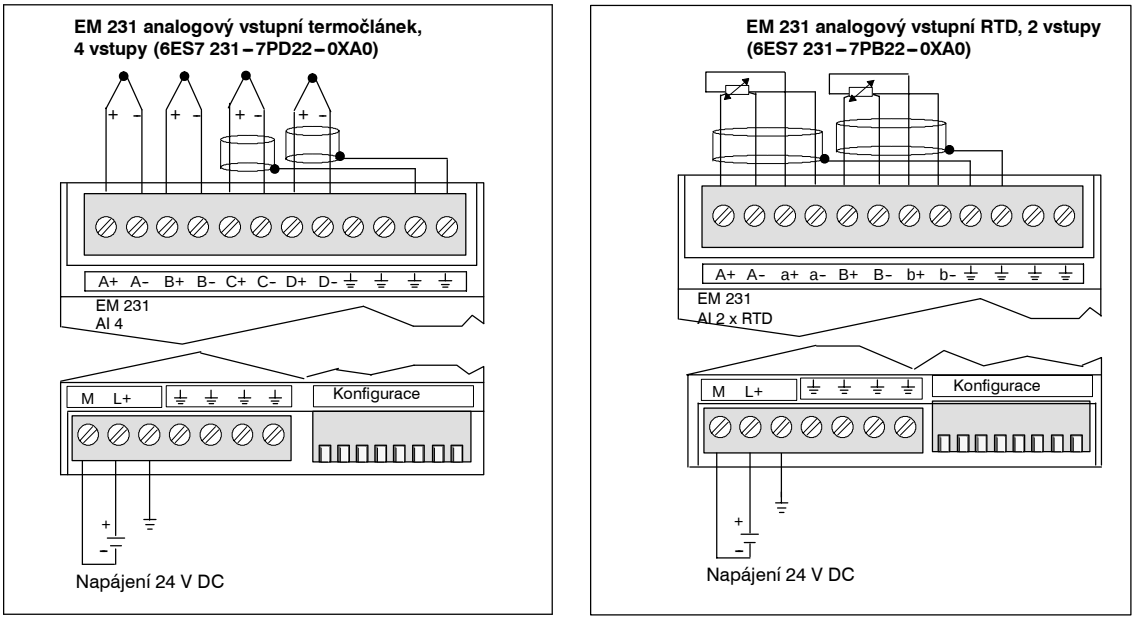
Tabulka A-24 Obecná specifikace termočlánkových a RTD modulů

Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	Požadavky V DC +5 V DC	+24 V DC
6ES7 231 – 7PD22 – 0XA0	EM 231 analogový vstup pro termočlánky, 4 vstupy	71,2 x 80 x 62	210 g	1,8 W	87mA	60 mA
6ES7 231 – 7PB22 – 0XA0	EM 231 analogový vstup RTD, 2 vstupy	71,2 x 80 x 62	210 g	1,8 W	87 mA	60 mA

Tabulka A-25 Specifikace termočlánkových a RTD modulů

Obecná	6ES7 231 – 7PD22 – 0XA0 Termočlánek	6ES7 231 – 7PB22 – 0XA0 RTD
Napěťové oddělení		
Okolí od logiky	500 V AC	500 V AC
Okolí od 24 V DC	500 V AC	500 V AC
24 V DC od logiky	500 V AC	500 V AC
Rozsah vstupního souhlasného napětí (vstupní kanál ke vstupnímu kanálu)	120 V AC	0
Potlačení souhlasného signálu	> 120 dB při 120 V AC	> 120 dB při 120 V AC
Typ vstupu	Plovoucí termočlánek	RTD vztažený k uzemnění modulu
Vstupní rozsahy[1]	Typy termočlánků (vyberte jeden na modul) S, T, R, E, N, K, J Rozsah napětí : +/– 80 mV	Typy RTD (vyberte jeden na modul): platina (Pt), měď (Cu), nikl (Ni) nebo odpor Dostupné typy RTD viz tabulka A-30.
Vstupní rozlišení		
Teplota	0.1 °C / 0,1 °F	0.1 °C / 0,1 °F
Napětí	15 bitů plus znaménko	–
Odpor	–	15 bitů plus znaménko
Princip měření	Sigma-delta	Sigma-delta
Doba aktualizace modulu: Všechny kanály	405 ms	405 ms (700 ms pro Pt10000)
Délka vodičů	100 metrů k senzoru max.	100 metrů k senzoru max5
Odpor smyčky vodičů	100 Ω max.	20 Ω, 2,7 Ω pro Cu max.
Potlačení rušení	85 dB při 50 Hz/60 Hz/ 400 Hz	85 dB při 50 Hz/60 Hz/ 400 Hz
Formát datového word	Napětí: –27648 až +27648	Odpor: 0 až +27648
Maximální ztráty snímače	–	1 mW
Vstupní impedance	∅1 MΩ	∅10 MΩ
Maximální vstupní napětí	30 VDC	30 V DC (snímání), 5 V DC (napájení)
Útlum vstupního filtru	–3 dB při 21 kHz	–3 dB při 3,6 kHz
Základní chyba	0,1 % plného rozsahu (napětí)	0,1 % plného rozsahu (odpor)
Opakovatelnost	0,05 % plného rozsahu	0,05 % plného rozsahu
Chyba studeného spoje	∂1,5 °C	–
Rozsah napětí napájení 24 V DC	20,4 až 28,8 V DC	20,4 až 28,8 V DC

[1] Volba vstupního rozsahu (teplota, napětí na odporu) platí pro všechny kanály modulu.



Obr. A-20 Identifikace připojovacích svorek termočlánkového modulu EM 231 a RTD modulu EM 231

Kompatibilita

Modul RTD a termočlánkový modul jsou určeny pro práci s CPU 222, CPU 224, CPU 224XP a CPU 226.

Tip

Modul RTD a termočlánkový modul podávají maximální výkon, jsou – li instalovány v prostředí se stálou teplotou.

Například termočlánkový modul EM 231 má speciální obvod pro kompenzaci studeného spoje, který měří teplotu na svorkách modulu a provádí nutné změny, aby kompenzoval teplotní rozdíly mezi referenční teplotou a teplotou modulu. Pokud se rychle mění teplota okolí v prostoru, kde je instalován termočlánkový modul EM 231, zanášejí se do měření další chyby.

Aby se dosáhlo maximální přesnosti a opakovatelnosti, doporučuje Siemens instalovat RTD moduly a termočlánkové moduly programovatelného automatu S7-200 na místech se stabilní teplotou okolí.

Odolnost proti rušení

Pro dosažení nejlepší odolnosti proti rušení použijte stíněné kabely. Pokud není termočlánkový vstupní kanál použit, zkratujte nepoužité kanálové vstupy nebo je zapojte paralelně s jiným kanálem.

Termočlánový modul EM 231

Termočlánkový modul EM 231 je praktické, napěťově oddělené rozhraní řady S7-200 pro sedm typů termočlánků: J, K, E, N, S, T a R. Umožňuje připojení automatu S7-200 na nízkoúrovňové analogové signály v rozsahu ∂80 mV. Všechny termočlánky připojené k modulu musí být stejného typu.

Základní informace o termočláncích

Termočlánky se vytvoří, kdykoliv jsou elektricky vzájemně spojeny dva různé kovy. Vzniká napětí, které je přímo úměrné teplotě spoje. Jde o malé napětí; jeden mikrovolt může představovat mnoho stupňů. Měření napětí termočlánku, kompenzace nadbytečných spojů a následující linearizace výsledků tvoří základ měření teploty pomocí termočlánků.

Když připojujete termočlánek k termočlánkovému modulu EM 231, jsou dva vodiče z různých kovů připojeny k signálovému konektoru modulu. Místo, kde jsou dva různé vodiče vzájemně spojeny, tvoří termočlánek.

Dva další termočlánky jsou vytvořeny tam, kde jsou dva různé vodiče připojeny k signálovému konektoru. Teplota konektoru dává vzniknout napětí, které se přičítá k napětí termočlánku. Pokud toto napětí není korigováno, bude se hlášená teplota lišit od měřené teploty.

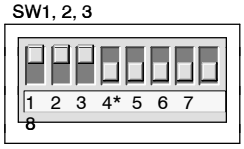
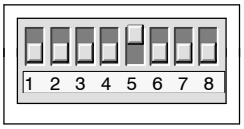
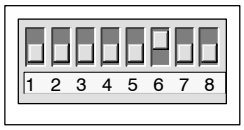
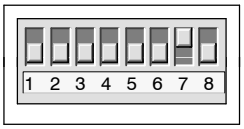
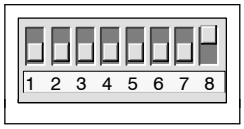
Kompenzace studených spojů se používá pro kompenzaci termočlánku vzniklého v místě konektoru. Tabulky pro termočlánky jsou založeny na referenční teplotě spoje, obvykle je to 0 °C. Kompenzace studeného spoje kompenzuje konektor na 0 °C. Kompenzace studeného spoje redukuje napětí přidané konektorovými termočlánky. Měří se vnitřní teplota modulu, která je převedena na hodnotu, která se přičte k převedené hodnotě snímače. Korigovaná převedená hodnota snímače je pak linearizována s použitím tabulek pro termočlánky.

Konfigurace termočlánkového modulu EM 231

Konfigurační DIP přepínače umístěné na spodní straně modulu umožňují vyběr typu termočlánku, detekce přerušeného vodiče, teplotní stupnice a kompenzace studeného spoje. Aby nastavení DIP přepínače nabyla platnosti, musíte vypnout a znovu zapnout programovatelný automat nebo odpojit napájení 24V.

DIP přepínač č. 4 je rezervován pro použití v budoucnosti. Nastavte DIP přepínače 4 na polohu 0 (dole neboli vypnuto). Tabulka A-26 ukazuje nastavení ostatních DIP přepínačů.

Tabulka A-26 Nastavení DIP přepínačů termočlánkového modulu

Přepínače 1,2,3	Typ termočlánku	Nastavení	Popis
SW1, 2, 3  Konfigurace ⇒1 – zapnuto ⇐0 – vypnuto * Nastavte DIP přepínač č. 4 do polohy 0 (dole).	J (Implicitní nastavení)	000	Přepínači 1 až 3 se volí typ termočlánku (nebo provoz s mV) pro všechny kanály modulu. Například pro termočlánek typu E SW1 = 0, SW2 = 1, SW3 = 1.
	K	001	
	T	010	
	E	011	
	R	100	
	S	101	
	N	110	
	+/–80 mV	111	
Přepínač č. 5	**Směr při detekci přerušeného vodiče**	**Nastavení**	**Popis**
SW5  Konfigurace ⇒1 – zapnuto ⇐0 – vypnuto	Nahoru po stupnici (+3276,7 stupňů)	0	0 indikuje kladnou hodnotu na přerušeném vodiči 1 indikuje zápornou hodnotu na přerušeném vodiči
	Dolů po stupnici (–3276,8 stupňů)	1	
Přepínač č. 6	**Povolení detekce přerušeného vodiče**	**Nastavení**	**Popis**
SW6  Konfigurace ⇒1 – zapnuto ⇐0 – vypnuto	Povoleno	0	Detekce přerušeného vodiče se provádí přivedením proudu 25 mA na vstupní svorky. Přepínač pro povolení detekce přerušeného vodiče povoluje nebo blokuje proudový zdroj. Kontrola rozsahu přerušeného vodiče se provádí vždy, i když je proudový zdroj zablokován. Termočlánkový modul EM 231 detekuje přerušený vodič, jestliže vstupní signál překročí přibližně ∂200 mV. Při zjištění přerušeného vodiče je odečet modulu nastaven na přednastavenou hodnotu.
	Blokovat	1	
Přepínač č. 7	**Teplotní stupnice**	**Nastavení**	**Popis**
SW7  Konfigurace ⇒1 – zapnuto ⇐0 – vypnuto	Celsius (°C)	0	Termočlánkový modul EM 231 může udávat teplotu ve stupních Celsia nebo Fahrenheita. Převod stupňů Celsia na stupně Fahrenheita se provádí v modulu.
	Fahrenheit (°F)	1	
Přepínač č. 8	**Studený spoj**	**Nastavení**	**Popis**
SW8  Konfigurace ⇒1 – zapnuto ⇐0 – vypnuto	Kompenzace studeného spoje povolena	0	Jestliže používáte termočlánky, musí být povolena kompenzace studeného spoje. Pokud není povolena kompenzace studeného spoje, budou převody z modulu chybné vzhledem k napětí, které vzniká při připojení vodiče termočlánku ke konektoru modulu. Tato kompenzace je automaticky blokována, jestliže zvolíte rozsah ∂80 mV.
	Kompenzace studeného spoje blokována	1	

Tip

- Zdroj proudu pro detekci přerušeného vodiče by mohl rušit signály některých nízkoúrovňových zdrojů, například termočlánkové simulátory.
- Vstupní napětí vyšší než asi ∂200 mV spustí detekci přerušeného vodiče, i když je proudový zdroj přerušeného vodiče blokován.

Tip

- Pokud dochází ke změnám teploty okolí, může být chyba modulu vyšší, než je uvedeno ve specifikaci.
- Překročení rozsahu teplot okolí specifikovaného pro modul může způsobit chybu kompenzace studeného spoje modulu.

Použití termočlánku: Stavové indikátory

Termočlánkový modul EM 231 dodává programovatelnému automatu hodnoty word, které představují teploty nebo chyby. Stavové bity indikují chybu rozsahu a poruchu napájení modulu. LED diody ukazují stav modulu. Uživatelský program by měl mít logiku, která by detekovala chyby a reagovala tak, jak to dané použití vyžaduje. Tabulka A-27 ukazuje stavové indikátory termočlánkového modulu EM 231.

Tabulka A-27 Stavové indikátory termočlánkového modulu EM 231

Chyba	Data v kanálu	SF LED Červená	24 V LED Zelená	Stavový bit rozsahu[1]	Chyba uživatelského napájení 24 V DC[2]
Bez chyby	Data z převodu	NESVÍTÍ	SVÍTÍ	0	0
Chybí 24 V	32766	NESVÍTÍ	NESVÍTÍ	0	1
Aktivována detekce přerušeného vodiče a proudový zdroj	–32768/32767	BLIKÁ	SVÍTÍ	1	0
Vstup mimo rozsah	–32768/32767	BLIKÁ	SVÍTÍ	1	0
Diagnostická chyba[3]	0000	SVÍTÍ	NESVÍTÍ	0	poznámka [3]

1 Stavový bit rozsahu je bit 3 v bytu chybového registru modulu (SMB9 pro modul 1, SMB11 pro modul 2, atd.)
2 Stavový bit chyby uživatelského napájení je bit 2 v bytu chybového registru modulu (SMB 9, SMB 11, atd., viz přílohu D)
3 Diagnostické chyby způsobují chybu konfigurace modulu. Stavový bit chyby napájení, může, ale nemusí být nastaven před chybou konfigurace modulu.

Tip

Datový formát kanálu je dvojkový doplněk 16bitových word. Teplota je uváděna v jednotkách 0,1 stupně. Je–li například naměřená teplota 100,2 stupoňů, modul uvádí 1002. Napěťová data jsou vynásobena koeficientem 27648. Například –60,0 mV je hlášeno jako –20736 (=–60 mV/80 mV * 27648).

Všechny čtyři kanály jsou aktualizovány každých 405 milisekund, pokud programovatelný automat načetl data. Pokud automat data nenačte během doby jedné aktualizace, uvádí modul stará data až do příští aktualizace modulu. Aby data v kanálu zůstávala aktuální, doporučuje se, aby program automatu načítal data nejméně tak často, jako je rychlost aktualizace modulu.

Tip

Při používání termočlánkového modulu EM 231 byste měli zablokovat analogovou filtraci programovatelného automatu. Analogová filtrace může zabránit včasnému rozpoznání chyby.

Tabulka A-28 Rozsahy teploty (°C) a přesnost pro různé typy termočlánků

Dat. word (1 číslice = 0,1°C)		Typ J	Typ K	Typ T	Typ E	Typ R, S	Typ N	±80mV	
Desítkové	Šetnáctkové								
32767	7FFF	>1200,0 °C	>1372,0 °C	>400,0 °C	>1000,0 °C	>1768,0 °C	>1300,0 °C	>94,071 mV	OF
⇒	⇒							⇒	⇒
32511	7EFF							94,071 mV	OR
:	:								
27649	6C01							80,0029 mV	
27648	6C00					⇒		80mV	NR
:	:								
17680	4510		⇒			1768,0 °C			
:	:								
13720	3598		1372,0 °C nad rozsahem				⇒		
:	:								
13000	32C8	⇒	1300,0 °C				1300,0 °C		
:	:								
12000	2EE0	1200,0 °C			⇒				
:	:								
10000	2710			⇒	1000,0 °C				
:	:								
4000	0FA0			400,0 °C		400.0°C			
:	:								
1	0001	0,1 °C	0,1 °C	0,1 °C	0,1 °C	0,1 °C	0,1 °C	0,0029 mV	
0	0000	0, 0 °C	0,0 °C	0,0 °C	0,0 °C	0,0 °C	0,0 °C	0,0 mV	
−1	FFFF	−0,1 °C	−0,1 °C	−0,1 °C	−0,1 °C	−0,1 °C	−0,1 °C	−0,0029 mV	
:	:					pod rozsahem			
−500	FE0C					−50,0 °C			
−1500	FA24	−150,0 °C				↓			
:	:								
−2000	F830	pod rozsahem	−200,0 °C						
:	:								
−2100	F7CC	−210,0 °C							
:	:								
−2400	F6A0				−240,0 °C				
:	:		pod rozsahem		pod rozsahem				
−2550	F60A			−255,0 °C					
:	:			pod rozsahem					
−2700	F574	↓	−270,0 °C	−270,0 °C	−270,0 °C		−270,0 °C		NR
:	:								
−27648	9400		↓	↓	↓		↓	−80 mV	
−27649	93FF							−80,0029 mV	
:	:								
−32512	8100							−94,071 mV	UR
↓	↓							↓	↓
−32768	8000	<−210,0 °C	<−270,0°C	<−270,0°C	<−270,0°C	<−50,0 °C	<−270,0°C	<−94,071 mV	UF
Přesnost v celém intervalu		∂0,1 %	∂0,3 %	∂0,6 %	∂0,3 %	∂0,6 %	∂0,4 %	∂0,1 %	
Přesnost (normální rozsah bez kompenzace studeného spoje)		∂1,5 °C	∂1,7 °C	∂1,4 °C	∂1,3 °C	∂3,7 °C	∂1,6 °C	∂0,10 %	
Chyba studeného spoje		∂1,5 °C	∂1,5 °C	∂1,5 °C	∂1,5 °C	∂1,5 °C	∂1,5 °C	Není k dispozici	

*OF = Přetečení; OR = Nad rozsahem; NR = Normální rozsah; UR = Pod rozsahem; UF = Podtečení

⇒znamená, že všechny analogové hodnoty větší než tato a pod prahem přerušeného vodiče mají za výsledek datovou hodnotu přetečení, 32767 (0x7FFF).
↓ znamená, že všechny analogové hodnoty menší než tato a vyšší než práh přerušeného vodiče mají za výsledek datovou hodnotu podtečení, −32768 (0x8000).

Tabulka A-29 Rozsahy teplot (°F) pro různé typy termočlánků

Datové word (1 číslice = 0,1 °F)		Typ J	Typ K	Typ T	Typ E	Typ R, S	Typ N	±80 mV	
Desítkové	Šestnáctkové								
32767	7FFF	>2192,0 °F	>2502,0 °F	>752,0 °F	>1832,0 °F	>3214,0 °F	>2372,0 °F	>94,071mV	OF
⇒	⇒					⇒		⇒	⇒
32511	7EFF							94,071 mV	
32140	7D90					3214,0 °F			OR
27649	6C01							80,0029 mV	
27648	6C00		⇒			2764,8 °F		80mV	
:	:								
25020	61B8		2502,0 °F				⇒		
:	:		nad rozsahem						NR
23720	5CA8	⇒	2372,0 °F				2372,0 °F		
:	:								
21920	55A0	2192,0 °F			⇒				
:	:								
18320	4790			⇒	1832,0 °F				
:	:								
7520	1D60			752,0 °F		752,0 °F			
:	:								
320	0140					pod rozsahem	32,0 °F		
:	:								
1	0001	0,1 °F	0,1 °F	0,1 °F	0,1 °F	0,1 °F	0,1 °F	0,0029 mV	
0	0000	0,0 °F	0,0 °F	0,0 °F	0,0 °F	0,0 °F	0,0 °F	0,0 mV	
−1	FFFF	−0,1 °F	−0,1 °F	−0,1 °F	−0,1 °F	−0,1 °F	−0,1 °F	−0,0029 mV	
:	:								
−580	FDBC					−58,0 °F			
:	:								
−2380	F6B4	−238,0 °F							
:	:								
−3280	F330	pod rozsahem	−328,0 °F				pod rozsahem		
:	:								
−3460	F27C	−346,0 °F				↓			
:	:		pod rozsahem						
−4000	F060				−400,0 °F				
:	:				pod rozsahem				
−4270	EF52			−427,0 °F					
:	:			pod rozsahem					
−4540	EE44	↓	−454,0 °F	−454,0 °F	−454,0 °F		−454,0 °F		NR
:	:								
−27648	9400		↓	↓	↓		↓	−80 mV	
−27649	93FF							−80,0029 mV	
:	:								
−32512	8100							−94,071 mV	OR
↓	↓							↓	↓
−3268	8000	<−346,0 °F	<−454,0 °F	<−454,0 °F	<−454,0 °F	<−58,0 °F	<−454,0 °F	<−94,07 mV	UF

*OF = Přetečení; OR = Nad rozsahem; NR = Normální rozsah; UR = Pod rozsahem; UF = Podtečení
⇒znamená, že všechny analogové hodnoty větší než tato a nižší než práh přerušeného vodiče mají za výsledek datovou hodnotu přetečení, 32767 (0x7FFF).
↓ znamená, že všechny analogové hodnoty menší než tato a vyšší než práh přerušeného vodiče mají za výsledek datovou hodnotu podtečení, −32768 (0x8000).

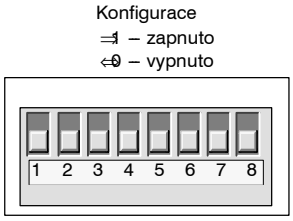
RTD modul EM 231

RTD modul EM 231 je praktické rozhraní řady S7-200 pro několik různých RTD (odporových teploměrů měření teploty). Zároveň umožňuje automatu S7-200 měřit tři různé rozsahy odporu. Obě RTD připojené k modulu musí být stejného typu.

Konfigurace RTD modulu EM 231

DIP přepínače umožňují zvolit typ RTD, konfiguraci zapojení, teplotní stupnici a směr při poruše čidla. DIP přepínače jsou umístěny na spodní straně modulu a znázorněny na obrázku A-21. Aby nastavení DIP přepínače nabyla platnosti, musíte vypnout a znovu zapnout programovatelný automat nebo odpojit napájení 24V.

Zvolte typ RTD nastavením DIP přepínačů č. 1, 2, 3, 4 a 5 tak, aby odpovídaly danému RTD, jak je to uvedeno v tabulce A-30. Nastavení ostatních DIP přepínačů viz tabulka A-31.



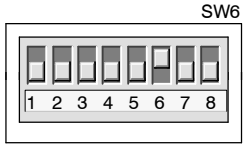
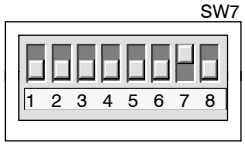
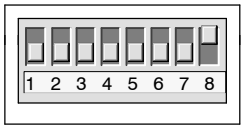
Obr. A-21 DIP přepínače pro RTD modul EM 231

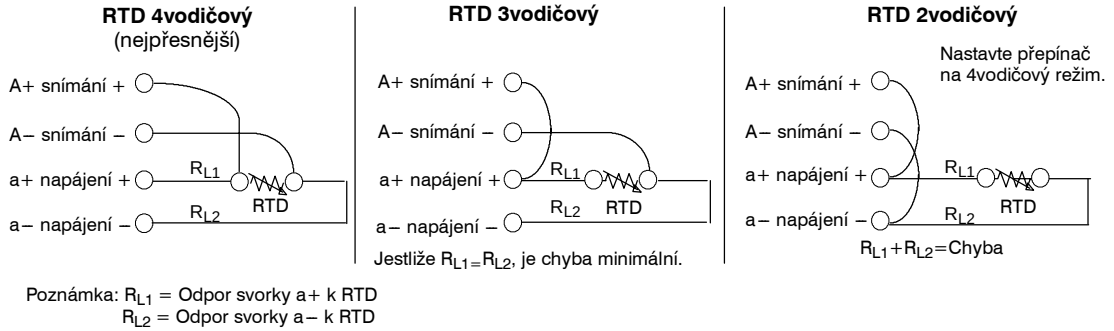
Tabulka A-30 Volba typu RTD: DIP přepínače 1 až 5

Typ RTD a alfa[1]	SW1	SW2	SW3	SW4	SW5	Typ RTD a alfa[1]	SW1	SW2	SW3	SW4	SW5
100T Pt 0,003850 (Implicitní)	0	0	0	0	0	100T Pt 0,003902	1	0	0	0	0
200T Pt 0,003850	0	0	0	0	1	200T Pt 0,003902	1	0	0	0	1
500T Pt 0,003850	0	0	0	1	0	500T Pt 0,003902	1	0	0	1	0
1000T Pt 0,003850	0	0	0	1	1	1000T Pt 0,003902	1	0	0	1	1
100T Pt 0,003920	0	0	1	0	0	REZERVA	1	0	1	0	0
200T Pt 0,003920	0	0	1	0	1	100T Ni 0,00672	1	0	1	0	1
500T Pt 0,003920	0	0	1	1	0	120T Ni 0,00672	1	0	1	1	0
1000T Pt 0,003920	0	0	1	1	1	1000T Ni 0,00672	1	0	1	1	1
100T Pt 0,00385055	0	1	0	0	0	100T Ni 73006178	1	1	0	0	0
200T Pt 0,00385055	0	1	0	0	1	120T Ni 0,006178	1	1	0	0	1
500T Pt 0,00385055	0	1	0	1	0	1000T Ni 0,006178	1	1	0	1	0
1000T Pt 0,00385055	0	1	0	1	1	10000T Pt 0,003850	1	1	0	1	1
100T Pt 0,003916	0	1	1	0	0	10T Cu 0,004270	1	1	1	0	0
200T Pt 0,003916	0	1	1	0	1	150T plný rozsah, odpor	1	1	1	0	1
500T Pt 0,003916	0	1	1	1	0	300T plný rozsah, odpor	1	1	1	1	0
1000T Pt 0,003916	0	1	1	1	1	600T plný rozsah, odpor	1	1	1	1	1

[1] Všechny RTD představují 0 °C při udaném odporu, kromě Cu 10 Ohm. Cu 10 Ohm je 25 °C při 10 Ohm a 0 °C při 9,035 Ohm.

Tabulka A-31 Nastavení DIP přepínačů pro RTD

Přepínač č. 6	Detekce přerušeného vodiče/ Mimo rozsah	Nastavení	Popis
	Nahoru po stupnici (+3276,7 stupňů)	0	Indikuje kladnou hodnotu na přerušeném vodiči nebo hodnotu mimo rozsah.
	Dolů po stupnici (–3276,8 stupňů)	1	Indikuje zápornou hodnotu na přerušeném vodiči nebo hodnotu mimo rozsah.
Přepínač č. 7	**Teplotní stupnice**	**Nastavení**	**Popis**
	Celsius (°C)	0	RTD modul EM 231 může udávat teplotu ve stupních Celsia nebo Fahrenheita. Převod stupňů Celsia na stupně Fahrenheita se provádí v modulu.
	Fahrenheit (°F)	1	
Přepínač č. 8	**Schéma zapojení**	**Nastavení**	**Popis**
	3vodičový	0	RTD modul můžete připojit ke snímači třemi způsoby (ukázáno na obrázku). (Nejpřesnější je 4vodičový.) Nejméně přesný je 2vodičový, který lze doporučit pouze tehdy, pokud je ve vaši aplikaci možné ignorovat chyby způsobené zapojením.
	2vodičový nebo 4vodičový	1	



Poznámka: R_{L1} = Odpor svorky a+ k RTD
R_{L2} = Odpor svorky a– k RTD

Obr. A-22 Připojení RTD ke snímači 4, 3 a 2 vodiči

Stavové indikátory RTD modulu EM 231

RTD modul dodává programovatelnému automatu hodnoty word, které představují teploty nebo chyby. Stavové bity indikují chybu rozsahu a poruchu napájení modulu. LED diody ukazují stav modulu. Uživatelský program by měl mít logiku, která by detekovala chyby a reagovala tak, jak to dané použití vyžaduje. Tabulka A-32 uvádí stavové indikátory RTD modulu EM 231.



Tip

Datový formát kanálu je dvojkový doplněk 16bitových word. Teplota je uváděna v jednotkách 0,1 stupně. (Jestliže je například naměřená teplota 100,2 stupňů, modul udává 1002.) Hodnoty odporů jsou vynásobeny koeficiente 27648. Například 75 % plného rozsahu modul uvádí jako 20736.

(225T / 300T * 27648 = 20736)

Tabulka A-32 Stavové indikátory RTD modulu EM 231

Chyba	Data v kanálu	SF LED Červená	24 V LED Zelená	Stavový bit rozsahu[1]	Chyba uživatelského napájení 24 V DC[2]
Bez chyby	Data z převodu	NESVÍTÍ	SVÍTÍ	0	0
Chybí 24 V	32766	NESVÍTÍ	NESVÍTÍ	0	1
SW zjistil přerušený vodič	–32768/32767	BLIKÁ	SVÍTÍ	1	0
Vstup mimo rozsah	–32768/32767	BLIKÁ	SVÍTÍ	1	0
Diagnostická chyba[3]	0000	SVÍTÍ	NESVÍTÍ	0	poznámka [3]

1 Stavový bit rozsahu je bit 3 v bytu chybového registru modulu (SMB9 pro modul 1, SMB11 pro modul 2, atd.)
2 Stavový bit chyby uživatelského napájení je bit 2 v bytu chybového registru modulu (například SMB 9, SMB 11, viz přílohu D)
3 Diagnostické chyby způsobují chybu konfigurace modulu. Stavový bit chyby napájení, může, ale nemusí být nastaven před chybou konfigurace modulu.

Data v kanálech se aktualizují každých 405 milisekund, pokud programovatelný automat načetl data. Pokud automat data nenačte během doby jedné aktualizace, uvádí modul stará data až do příští aktualizace modulu. Aby data v kanálu zůstávala aktuální, doporučuje se, aby program automatu načítal data nejméně tak často, jako je rychlost aktualizace modulu.



Tip

Při používání RTD modulu určitě zablokujte analogovou filtraci programovatelného automatu. Analogová filtrace může zabránit včasnému rozpoznání chyby.

Detekce přerušeného vodiče se provádí interním softwarem RTD modulu. Vstupy mimo rozsah a detekované stavy přerušených vodičů jsou signalizovány nastavením stavového bitu rozsahu v SMB a nastavením dat kanálu na stupnici pro nastavení přepínačů nahoru nebo dolů. Detekce přerušeného vodiče trvá minimálně tři programové cykly modulu a může trvat i déle, podle toho, který vodič je přerušený/které vodiče jsou přerušené. Přerušené vodiče napájení+ nebo napájení– snímače jsou detekovány v minimální době. Přerušené vodiče snímání+ a/nebo snímání– z čidla mohou být detekovány za 5 i více sekund. Přerušené vodiče snímání mohou náhodně dávat platná data, s občasnou detekcí přerušeného vodiče, obzvláště v prostředí s elektrickým šumem. Elektrický šum může také prodloužit dobu potřebnou pro detekci přerušeného vodiče. Po ohlášení platných dat se doporučuje zablokovat indikaci přerušeného vodiče a hodnoty mimo rozsah v uživatelském programu.



Tip

Jestliže máte nepoužitý kanál, můžete tento kanál zapojit s odporem namísto RTD, aby detekce přerušeného vodiče nezpůsobila blikání SF LED. Odpor musí mít jmenovitou hodnotu RTD. Např. použijte 100 Ohm pro PT100 RTD .

Rozsahy RTD modulu EM 231

Rozsahy teplot a přesnost pro každý typ RTD modulu EM 231 jsou uvedeny v tabulkách A-33 a A-34.

Tabulka A-33 Rozsahy teploty (°C) a přesnost pro různé typy RTD

Systémové word (1 číslice = 0,1 °C)		Pt10000	Pt100, Pt200, Pt500, Pt1000	Ni100, Ni120, Ni1000[1]	Cu10	0 – 150 T	0 – 300 T	0 – 600 T	
Desítkové	Šetnáctkové								
32767	7FF.								
32766	7FFE					⇒	⇒	⇒	
32511	7EFF					176.383T	352.767T	705.534T	
29649	6C01					150.005T	300.011T	600.022T	
27648	6C00					150.000T	300.000T	600.000T	
25000	61A8								⇒
18000	4650								OR
15000	3A98								
13000	32C8	⇒	⇒						
10000	2710	1000.0°C	1000.0°C						
8500	2134		850.0°C						
6000	1770	600.0°C			⇒				
3120	0C30			⇒	312.0°C				
2950	0B86			295.0°C					
2600	0A28				260.0°C				
2500	09C4			250.0°C					
1	0001	0.1°C	0.1°C	0.1°C	0.1°C	0.005T	0.011T	0.022T	
0	0000	0.0°C	0.0°C	0.0°C	0.0°C	0.000T	0.000T	0.000T	
–1	FFFF	–0.1°C	–0.1°	–0.1°C	–0.1°C	(nejsou možné záporné hodnoty)			
						⇔	⇔	⇔	
–600	FDA8			–60.0°C					NR
–1050	FBE6			–105.0°C					
				⇔					
–2000	F830	–200.0°C	–200.0°		–200.0°C				
–2400	F6A0				–240.0°C				
–2430	F682	–243.0°C	–243.0°C		⇔				
		⇔	⇔						
–5000	EC78								
–6000	E890								UR
–10500	D6FC								⇔
–12000	D120								
–20000	4E20								
–32767	8001								
–32768	8000								
Přesnost v celém intervalu		∂0,4 %	∂0,1 %	∂0,2 %	∂0,5 %	∂0,1 %	∂0,1 %	∂0,1 %	
Přesnost (jmenovitý rozsah)		∂4 °C	∂1 °C	∂0,6 °C	∂2,8 °C	∂0,15 T	∂0,3 T	∂0,6 T	
*OF = Přetečení; OR = Nad rozsahem; NR = Jmenovitý rozsah; UR = Pod rozsahem; UF = Podtečení									
⇒nebo ⇔indikuje, že všechny analogové hodnoty větší než mezní hlásí zvolenou hodnotu mimo rozsah, 32767 (0x7FF.) nebo –32768 (0x8000).									

[1] Spodní limit pro normální rozsah 1000 T Ni s alfou 0,006178 je 0 °C bez podrozsahu. 1000 T Ni s alfou 0,00672 je uveden v této tabulce.

Tabulka A-34 Rozsahy teplot (°F) pro různé typy RTD

Systémové word (1 číslice = 0,1 °F)		PT1000	PT100, Pt200, Pt500, Pt1000	Ni100, Ni120, Ni1000[1]	Cu 10	
Desítkové	Šestnáctkové					
32767	7FF.					
32766	7PHAGE					
						⇒
						Nad rozsahem
		⇒	⇒			
18320	4790	1832,0 °F	1832,0 °F			
15620	3D04		1562,0 °F			
11120	2B70	1112,0 °F				
					⇒	
5936	1730			⇒	593,6 °F	
5630	15FE			563,0 °F		
5000	1388				500,0 °F	
4820	12D4			482,0 °F		
						Normální rozsah
1	0001	0,1 °F	0,1 °F	0,1 °F	0,1 °F	
0	0000	0,0 °F	0,0 °F	0,0 °F	0,0 °F	
–1	FFFF	–0,1 °F	–0,1 °F	–0,1 °F	–0,1 °F	
–760	FD08			–76,0 °F		
–1570	F9DE			–157,0 °F		
				⇔		
–3280	F330	–328,0 °F	–328,0 °F		–328,0 °F	
–4000	F060				–400,0 °F	
–4054	F02A	–405,4 °F	–405,4 °F		⇔	
		⇔	⇔			
–5000	EC78					
–6000	E890					Pod rozsahem
–10500	D6FC					⇔
–32767	8001					
–32768	8000					

⇒nebo ⇔ndikuje, že všechny analogové hodnoty větší než mezní hlásí zvolenou hodnotu mimo rozsah, 32767 (0x7FFF) nebo –32768 (0x8000).

[1] Spodní limit pro normální rozsah 1000 Ω Ni s alfou 0,006178 je 32 °F bez podrozsahu. 1000 Ω Ni s alfou 0,00672 je uveden v této tabulce.

Specifikace modulu EM 277 PROFIBUS–DP

Tabulka A-35 Objednací číslo modulu EM 277 PROFIBUS–DP

Objednací číslo	Rozšiřovací modul	Vstupy EM	Výstupy EM	Odnímatelná svorkovnice
6ES7 277–0AA22–0XA0	EM 277 PROFIBUS–DP	–	–	Ne

Tabulka A-36 Obecná specifikace modulu EM 277 PROFIBUS–DP

Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	Požadavky V DC +5 V DC	+24 V DC
6ES7 277–0AA22–0XA0	EM 277 PROFIBUS–DP	71 x 80 x 62	175 g	2,5 W	150mA	Viz níže

Tabulka A-37 Specifikace modulu EM 277 PROFIBUS–DP

Obecná	6ES7 277–0AA22–0XA0
Počet portů (omezený výkon)	1
Elektrické rozhraní	RS–485
Přenosové rychlosti PROFIBUS–DP/MPI (nastaveno automaticky)	9,6; 19,2; 45,45; 93,75; 187,5 a 500 kbaud; 1; 1,5; 3; 6 a 12 Mbaud
Protokoly	PROFIBUS–DP slave a MPI slave
Délka kabelu	
Až do 93,75 kbaud	1 200 m
187,5 kbaud	1 000 m
500 kbaud	400 m
1 až 1,5 Mbaud	200 m
3 až 12 Mbaud	100 m
Možnosti sítě	
Nastavení adresy stanice	0 až 99 (nastavuje se otočnými přepínači)
Maximální počet stanic na segment	32
Maximální počet stanic v síti	126 až 99 stanic EM 277
Připojení MPI	celkem 6, 2 vyhrazeny (1 pro PG a 1 pro OP)
Požadavky na zdroj 24 V DC	
Rozsah napětí	20,4 až 28,8 V DC (třída 2, omezený výkon nebo napájení snímačů z PLC)
Maximální proud Pouze modul s aktivním portem Přidáno 90 mA zátěže na 5V portu Přidáno 120 mA zátěže na 24V portu	 30 mA 60 mA 180 mA
Zvlnění šumu (<10 MHz)	<1 V špička–špička (maximálně)
Napěťové oddělení (pole od logiky)[1]	500 V AC po dobu 1 minuty
Napájení 5 V DC na komunikačním portu	
Maximální proud na port	90 mA
Napěťové oddělení (24 V DC od logiky)	500 V AC po dobu 1 minuty
Napájení 24 V DC na komunikačním portu	
Rozsah napětí	20,4 až 28,8 V DC
Maximální proud na port	120 mA
Proudové omezení	0,7 až 2,4 A
Napěťové oddělení	Není odděleno, stejný obvod jako vstupní 24 V DC

[1] Logika modulu není napájena ze zdroje 24 V DC. 24 V DC napájí komunikační port.

CPU S7-200, které podporují inteligentní moduly

Modul EM 277 PROFIBUS–DP slave je inteligentní rozšiřovací modul určený pro spolupráci s CPU S7-200 uvedenými v tabulce A-38.

Tabulka A-38 Kompatibilita modulu EM 277 PROFIBUS–DP s CPU S7-200

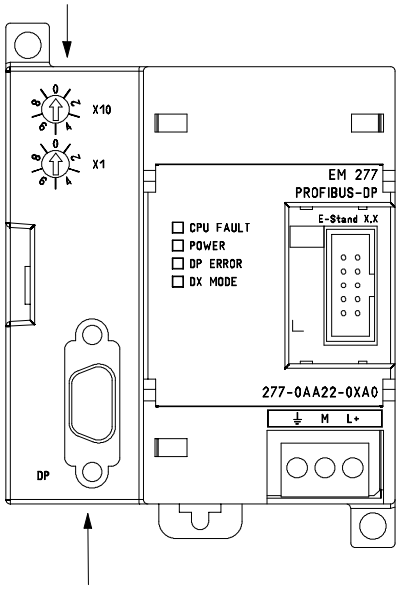
CPU	Popis
Model CPU 222 verze 1.10 nebo vyšší	CPU 222 DC/DC/DC a CPU 222 AC/DC/relé
Model CPU 224 verze 1.10 nebo vyšší	CPU 224 DC/DC/DC a CPU 224 AC/DC/relé
Model CPU 224XP verze 2.0 nebo vyšší	CPU 224XP DC/DC/DC a CPU 224XP AC/DC/relé
Model CPU 226 verze 1.00 nebo vyšší	CPU 226 DC/DC/DC a CPU 226 AC/DC/relé

Přepínače adres a stavové LED

Přepínače adres a LED diody jsou umístěny na přední straně modulu a znázorněny na obrázku A-23, kde je zobrazeno i rozmístění vývodů konektoru portu DP slave. Stavové LED jsou popsány v tabulce A-42.

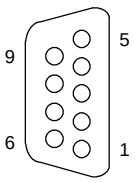
Pohled na EM 277 PROFIBUS–DP zepředu

Přepínače adres:
x10 = nastavuje nejvyšší platnou číslici adresy
x1 = nastavuje nejnižší platnou číslici adresy



Konektor portu DP Slave

9pinový konektor (zásuvka)



9kolíkový subminiaturní konektor typu D – rozmístění vývodů

Vývod č.	Popis
1	Ochranné uzemnění rozvaděče propojené s rámem konektoru
2	Střední vodič 24 V (stejné jako M na svorkovnici)
3	Napěťově oddělený signál B (RxD/TxD+)
4	Napěťově oddělená výzva k vysílání (úroveň TTL)
5	Napěťově oddělený střední vodič +5 V
6	Napěťově oddělené +5 V (maximálně 90 mA)
7	+24 V (maximálně 120 mA, s ochrannou diodou se závěrným napětím)
8	Napěťově oddělený signál A (RxD/TxD–)
9	Nezapojeno

Poznámka: Napěťově odděleno znamená izolaci 500 V od digitální logiky a 24V napájecího vstupu.

Obr. A-23 EM 277 PROFIBUS–DP

Komunikace se standardními distribuovanými periferními zařízeními (DP)

PROFIBUS – DP (neboli standard DP) je protokol pro dálkovou I/O komunikaci definovaný evropskou normou EN 50170. Zařízení, která dodržují tuto normu, jsou kompatibilní, i když jsou vyráběna různými firmami. DP znamená distribuovaná periferní zařízení, tj. dálkové vstupy a výstupy. PROFIBUS je zkráceno z Process Field Bus.

Modul EM 277 PROFIBUS – DP má implementovaný standardní protokol DP tak, jak je definován pro zařízení typu slave v následujících normách pro komunikační protokoly:

- EN 50 170 (PROFIBUS) popisuje přístup po sběrnici, protokol přenosu a udává vlastnosti přenosového média.
- EN 50 170 (Standard DP) popisuje vysokorychlostní cyklickou výměnu dat mezi zařízeními DP master a DP slave. Tato norma definuje postupy pro konfiguraci a přiřazení parametrů, vysvětluje fungování cyklické výměny dat s distribuovanými vstupy a výstupy a obsahuje seznam podporovaných diagnostických funkcí.

DP master je nakonfigurován tak, aby poznal adresy a typy zařízení slave a jakoukoliv informaci o přiřazení parametru, které zařízení slave vyžadují. Master obsahuje i informace o tom, kam umístit data, která načte ze zařízení slave (vstupy), a odkud získat data, která zapisuje na slave (výstupy). DP master zakládá síť a pak incializuje svá zařízení DP slave. Master zapisuje informace o poioazení parametru a konfiguraci vstupu a výstupu na zaoízení slave. Potom master přečte diagnostiku ze zařízení DP slave, aby ověřil, že toto zařízení přijalo parametry a konfiguraci vstupu a výstupu. Následně master zahájí výměnu I/O dat se zařízením slave. Při každé transakci se zařízením slave zapisuje výstupy a čte vstupy. Režim výměny dat pokračuje po neomezenou dobu. Zařízení slave mohou uvědomit master, když dojde k výjimečnému stavu, a master pak čte diagnostické informace ze zařízení slave.

Po tom, co DP master zapsal parametry a konfiguraci vstupu a výstupu na DP slave a slave akceptoval parametry a konfiguraci od masteru, který vlastní toto zařízení slave. Slave přijímá požadavky na zapisování pouze od svého vlastního masteru. Ostatní zařízení mastery v dané síti mohou číst vstupy a výstupy tohoto zařízení slave, ale nemohou do něj zapisovat.

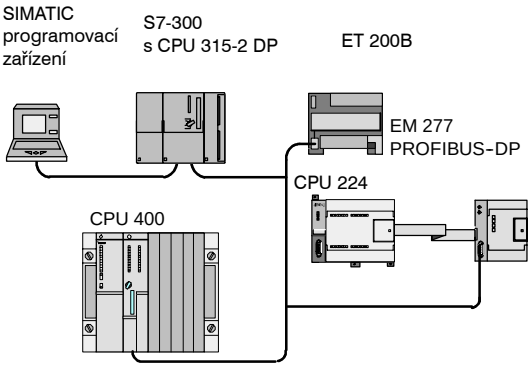
Použití EM 277 pro připojení S7-200 jako zařízení DP slave

CPU S7-200 může být připojena k síti PROFIBUS – DP prostřednictvím rozšiřovacího modulu EM 277 PROFIBUS – DP slave. Modul EM 277 je připojen k CPU S7-200 přes sériovou sběrnici. Síť PROFIBUS je k modulu EM 277 PROFIBUS – DP připojena přes jeho DP komunikační port. Tento port pracuje při kterékoliv přenosové rychlosti PROFIBUS v rozsahu od 9600 baudů do 12 Mbaudů. Podporované přenosové rychlosti najdete ve specifikaci modulu EM 277 PROFIBUS – DP.

Jako zařízení DP slave přijímá modul EM 277 od masteru několik různých konfigurací vstupu a výstupu, což umožňuje přizpůsobit množství přenášených dat tak, aby to odpovídalo požadavkům daného použití. Na rozdíl od mnoha DP zařízení nepřenáší modul EM 277 pouze I/O data. Vstupy, hodnoty čítače, hodnoty časovače nebo jiné vypočítané hodnoty je možné přenést do zařízení master tak, že se data nejprve přesunou do variabilní paměti CPU S7-200. Obdobně jsou data ze zařízení master ukládána do variabilní paměti CPU S7-200 a mohou se přesouvat do jiných datových oblastí.

DP port modulu EM 277 PROFIBUS – DP může být připojen k zařízení DP master v síti a přitom komunikovat jako MPI slave s jinými zařízeními typu master, jako jsou programovatelné zařízení SIMATIC nebo CPU S7-300/S7-400. Obrázek A-24 znázorňuje síť PROFIBUS s CPU 224 a modulem EM 277 PROFIBUS – DP.

- CPU 315 – 2 je DP master a byl nakonfigurován programovacím zařízením SIMATIC s programovacím sofwarem STEP 7.
- CPU 224 je DP slave a vlastní ho CPU 315 – 2. Modul ET 200 je také zařízení slave, které vlastní CPU 315 – 2.
- CPU S7-400 je připojen k síti PROFIBUS a čte data z CPU 224 pomocí instrukcí XGET v uživatelském programu CPU S7-400.

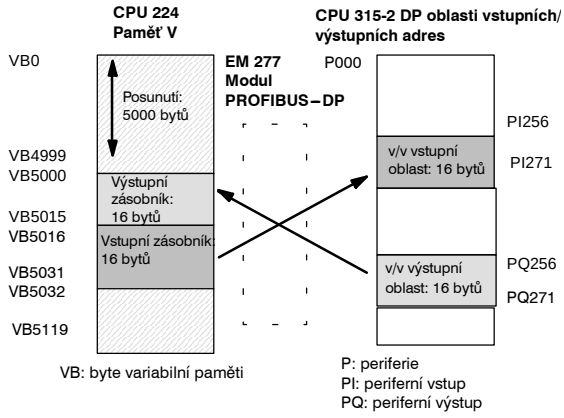


Obr. A-24 Modul EM 277 PROFIBUS – DP a CPU 224 v síti PROFIBUS

Konfigurace

Chcete – li používat EM 277 PROFIBUS – DP jako DP slave, musíte nastavit adresu stanice DP portu tak, aby odpovídala adrese v konfiguraci zařízení master. Adresa stanice se nastavuje otočnými přepínači na modulu EM 277. Po provedení změny přepínačem je třeba CPU vypnout a opět zapnout, aby nová adresa zařízení slave nabyla platnosti.

Master si vyměňuje data s každým svým slave tak, že posílá informace ze své výstupní oblasti do výstupního zásobníku zařízení slave, nazývaného ”Receive mailbox” (Schránka pro příjem). Zařízení slave odpovídá na zprávu vrácením vstupního zásobníku, nazývaného ”Send mailbox” (Schránka pro odesílání), který master uloží ve vstupní oblasti.



Obr. A-25 Paměť V a oblast adres vstupu a výstupu

Obrázek A-25 ukazuje příklad paměti V a oblasti adres vstupů a výstupů zařízení PROFIBUS – DP master.

EM 277 PROFIBUS – DP může být nakonfigurován DP master tak, aby přijímal výstupní data od zařízení master a vracel mu data vstupní. Zástobníky pro výstupní a vstupní data jsou umístěny ve variabilní paměti (V paměť) CPU S7-200. Když konfigurujete zařízení DP master, definujete místo bytu v paměti V, kde má začínat zásobník pro výstupní data, jako součást informace o přiřazení parametrů pro EM 277. Definujete i konfiguraci vstupů a výstupů jako množství výstupních dat, které se zapisuje do CPU S7-200, a jako množství vstupních dat, které se vrací z CPU S7-200. EM 277 určuje velikost vstupního a výstupního zýsobníku z konfigurace vstupů a výstupů. DP master zapisuje informace o přiřazení parametrů a konfiguraci vstupů a výstupů do modulu EM 277 PROFIBUS DP. Potom EM 277 přenáší do CPU S7-200 adresu a délku vstupních a výstupních dat v paměti V.

Obrázek A-25 ukazuje paměťový model paměti V v CPU 224 a oblast adres vstupů a výstupů CPU DP mastera. V tomto příkladu DP master definoval vstupní/výstupní konfiguraci 16 výstupních bytů a 16 vstupních bytů a offset V paměti 5000. Výstupní a vstupní zásobník v CPU 224 jsou oba 16 bytů dlouhé (dáno v konfiguraci vstupu a výstupu). Zásobník pro výstupní data začíná na V5000; vstupní zásobník následuje bezprostředně za výstupním a začíná na V5016. Výstupní data (ze zařízení master) se zapisují do paměti V od V5000. Vstupní data (do zařízení master) se čtou z paměti V od V5016.



Tip

Pokud pracujete s datovou jednotkou (konzistentní data) tři byty nebo s datovými jednotkami většími než čtyři byty, je nutné použít SFC14 na čtení vstupů DP slave a SFC15 na adresaci výstupů DP slave. Více informací najdete v *Referenčních manuálech System software for S7-300 and S7-400 a Standard Function*.

Tabulka A-39 uvádí konfigurace, které jsou podporovány modulem EM 277 PROFIBUS – DP. Implicitní konfigurace modulu EM 277 je dva word pro vstup a dva word pro výstup.

Tabulka A-39 Volby konfigurace EM 277

Konfigurace	Vstupy masteru	Výstupy masteru	Konzistence dat
1	1 word	1 word	Konzistence po wordech
2	2 word	2 word	
3	4 word	4 word	
4	8 word	8 word	
5	16 word	16 word	
6	32 word	32 word	
7	8 word	2 word	
8	16 word	4 word	
9	32 word	8 word	
10	2 word	8 word	
11	4 word	16 word	
12	8 word	32 word	
13	2 byty	2 byty	Konzistence po bytech
14	8 bytů	8 bytů	
15	32 byty	32 byty	
16	64 byty	64 byty	
17	4 byty	4 byty	Zásobníková konzistence
18	8 bytů	8 bytů	
19	12 bytů	12 bytů	
20	16 bytů	16 bytů	

Umístění vstupního a výstupního zásobníku v paměti V CPU S7-200 můžete nakonfigurovat libovolně. Implicitně nastavená adresa vstupního a výstupního zásobníku je VB0. Umístění vstupního a výstupního zásobníku je součástí informací o přiřazení parametrů, které master zapisuje do CPU S7-200. Master musí být konfigurován tak, aby rozpoznal svoje zařízení slave a do každého zapisoval požadované parametry a konfiguraci vstupů a výstupů.

Pro konfiguraci zařízení DP master použijte následující nástroje:

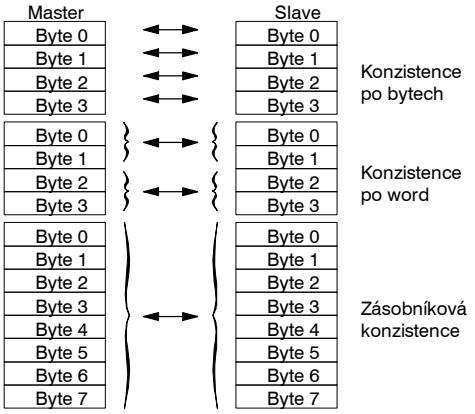
- Pro SIMATIC S5 master použijte software COM PROFIBUS Windows
- Pro SIMATIC S7 master použijte programovací software STEP 7
- Pro SIMATIC 505 master použijte COM PROFIBUS a TISOFT2 nebo SoftShop

Podrobné informace o používání těchto konfiguračních a programovacích balíků najdete v příručkách pro daný software. Podrobné informace o síti PROFIBUS a jejích komponentách najdete v *Manuálu ET 200 Distributed I/O System.*

Konzistence dat

PROFIBUS podporuje tři typy konzistence dat:

- ❑ Konzistence po bytech zajišťuje, že jsou byty přenášeny jako celé jednotky.
- ❑ Konzistence po wordech zajišťuje, že přenosy word nemohou být přerušeny jinými procesy v CPU (dva byty, ze kterých se word skládá, jsou vždy přenášeny dohromady a nemohou být rozděleny). Konzistenci po wordech použijte, pokud mají přenášená data hodnotu integer .



Obr. A-26 Konzistence dat; po bytech; wordech; zásobníková

- ❑ Zásobníková konzistence zajišťuje, že je celý datový zásobník přenášen jako jeden celek a není přerušovaný žádným jiným procesem v CPU. Tato konzistence se používá, pokud mají přenášená data hodnotu double word, obsahují plovoucí desetinnou čárku nebo patří – li celá skupina hodnot k jednomu výpočtu nebo položce.

Konzistence dat se nastavuje jako součást konfigurace vstupů a výstupů zařízení master. Zvolená konzistence dat se zapíše do DP slave jako součást jeho inicializace. Obě zařízení, DP master i DP slave, používají zvolenou konzistenci dat na zajištění nepřerušovaného vzájemného přenosu datových hodnot (byty, word nebo zásobníky). Různé typy konzistence jsou uvedeny na obrázku A-26.

Upozornění pro uživatelský program

Po úspěšné konfiguraci modulu EM 277 PROFIBUS – DP zařízením DP master vstoupí EM 277 a DP master do režimu výměny dat. V tomto režimu master zapisuje výstupní data do modulu EM 277 PROFIBUS – DP a modul EM 277 odpovídá aktuálními vstupními daty CPU S7-200. EM 277 nepřetržitě aktualizuje svoje vstupy z CPU S7-200, aby mohl poskytovat nejnovější vstupní data zařízení DP master. Pak modul přenáší výstupní data do CPU S7-200. Výstupní data ze zařízení master jsou uložena do paměti V (výstupní zásobník) se začátkem na adrese, kterou DP master dodal během inicializace. Vstupní data pro master se čtou z míst paměti V (vstupní zásobník), která následují bezprostředně za výstupními daty.

Výstupní data ze zařízení master musí být přesunuta uživatelským programem v CPU S7-200 z výstupního zásobníku do datových oblastí, kde mají být užívána. Obdobně musí být vstupní data pro master přesunuta z různých datových oblastí do vstupního zásobníku pro přenos do masteru.

Výstupní data z DP masteru jsou uložena do paměti V okamžitě po provedení uživatelského programu v rámci programového cyklu. Současně jsou do masteru kopírována vstupní data z paměti V do EM 277 pro přesun.

Výstupní data z masteru se zapisují do paměti V pouze tehdy, když toto zařízení dodává nová data.

Vstupní data pro master jsou do něj přenášena při další vzájemné datové výměně.

V době tvorby uživatelského programu pro CPU S7-200 CPU musí být známa počáteční adresa datových zásobníků v paměti V a jejich velikost.

Stavové informace

Každému inteligentnímu modulu je přiděleno 50 bytů speciální paměti (SM) na základě jeho fyzického umístění. Modul aktualizuje místa v paměti SM odpovídající vzájemné pololze modulu a CPU (vzhledem k ostatním modulům). Pokud je to první modul, aktualizuje SMB200 až SMB249. Pokud je to druhý modul, aktualizuje SMB250 až SMB299 atd. Viz tabulku A-40.

Tabulka A-40 Byty speciální paměti SMB200 až SMB549

Byty speciální paměti SMB200 až SMB549						
Inteligentní modul na pozici 0	**Inteligentní modul na pozici 1**	**Inteligentní modul na pozici 2**	**Inteligentní modul na pozici 3**	**Inteligentní modul na pozici 4**	**Inteligentní modul na pozici 5**	**Inteligentní modul na pozici 6**
SMB200 až SMB249	SMB250 až SMB299	SMB300 až SMB349	SMB350 až SMB399	SMB400 až SMB449	SMB450 až SMB499	SMB500 až SMB549

Tato místa v paměti SM ukazují implicitní hodnoty, pokud nebyla navázána DP komunikace se zařízením master. Po tom, co master zapsal do modulu EM 277 PROFIBUS–DP parametry a konfiguraci vstupu a výstupu, ukazují tato místa v paměti SM konfiguraci nastavenou DP masterem. Dříve než použijete informace z míst v paměti SM uvedených v tabulce A-41 nebo data v zásobníku paměti V, musíte zkontrolovat stavový byte protokolu (například SMB224 pro místo 0). Tím se ujistíte, že je EM 277 v danou dobu v režimu výměny informací se zařízením master.



Tip

Velikost vstupních/výstupních zásobníků EM 277 PROFIBUS–DP nebo jejich umístění není možné konfigurovat zápisem do paměti SM. Pro provoz v režimu DP může modul EM 277 PROFIBUS–DP nakonfigurovat pouze DP master.

Tabulka A-41 Speciální paměťové byty pro EM 277 PROFIBUS–DP

Inteligentní modul na pozici 0	...	Inteligentní modul na pozici 6	Popis
SMB200 až SMB215	...	SMB500 až SMB515	Název modulu (16 ASCII znaků) ”EM277 ProfibusDP”
SMB216 až SMB219	...	SMB516 až SMB519	Číslo verze firmware (4 znaky ASCII) xxxx
SMW220	...	SMW520	Chybový kód 16#0000 Bez chyby 16#0001 Není napájení ze strany uživatele 16#0002 až 16#FFFF Rezervováno
SMB222	...	SMB522	Adresa stanice DP slave, jak je nastavena přepínači adres (0 – 99 desítkově)
SMB223	...	SMB523	Rezervováno
SMB224	...	SMB524	Stavový byte standardního DP protokolu MSB LSB 0 0 0 0 0 0 S1 S0 S1 S0 Popis stavového bytu standardního DP 0 0 Komunikace DP nebyla od zapnutí zdroje inicializována 0 1 Zjištěna chyba konfigurace/přiřazení parametrů 1 0 Aktuálně v režimu výměny dat 1 1 Výpadek režimu výměny dat
SMB225	...	SMB525	Protokol DP Standard – adresa masteru, kterému patří slave (0 až 126)
SMW226	...	SMW526	Protokol DP Standard – adresa výstupního zásobníku v paměti V jako posunutí od VB0
SMB228	...	SMB528	Protokol DP Standard – počet bytů výstupních dat
SMB229	...	SMB529	Protokol DP Standard – počet bytů vstupních dat
SMB230 až SMB249	...	SMB530 až SMB549	Rezervováno – vymazáno při zapnutí

Poznámka: Místa v paměti SM jsou aktualizována pokaždé, když modul DP slave přijme informace o konfiguraci/přiřazení parametrů. Tato místa jsou aktualizována dokonce i v případě zjištění chyby konfigurace/přiřazení parametrů. Údaje jsou vymazány při každém zapnutí.

Stavové LED diody EM 277 PROFIBUS–DP

Na předním panelu modulu EM 277 PROFIBUS–DP jsou čtyři stavové kontrolky (LED), které indikují provozní stav DP portu:

- ❏ Po zapnutí CPU S7-200 zůstává LED DX MODE nerozsvícená, dokud nedojde k pokusu o DP komunikaci.
- ❏ Po úspěšné inicializaci DP komunikace (modul EM 277 PROFIBUS–DP vstoupil do režimu výměny dat se zařízením master) začne LED DX MODE svítit zeleně a zůstane rozsvícená, dokud není ukončen režim výměny dat.
- ❏ Dojde–li k přerušení DP komunikace, které způsobí, že modul EM 277 opustí režim výměny dat, LED DX MODE zhasne a rozsvítí se červená kontrolka DP ERROR. Tento stav setrvává, dokud není vypnuta CPU S7-200 nebo není obnovena výměna dat.
- ❏ Pokud je chyba v konfiguraci vstupů a výstupů nebo v informacích o parametrech, které DP master zapisuje do modulu EM 277, bliká červeně LED DP ERROR.
- ❏ Pokud není přivedeno uživatelské napájení 24 V DC, nebude svítit LED POWER .

Tabulka A-42 shrnuje významy stavových LED EM 277.

Tabulka A-42 Stavové LED modulu EM 277 PROFIBUS–DP

LED	NESVÍTÍ	ČERVENÁ	BLIKAJÍCÍ ČERVENÁ	ZELENÁ
CPU FAULT	Modul je v pořádku	Interní porucha modulu	– –	– –
POWER	Není uživatelské napájení 24 V DC	– –	– –	Uživatelské napájení 24 V DC v pořádku
DP ERROR	Bez chyby	Opuštěn režim výměny dat	Chyba přiřazení parametrů/konfigurace	– –
DX MODE	Není v režimu výměny dat	– –	– –	V režimu výměny dat

Poznámka: Když je modul EM 277 PROFIBUS–DP používán výhradně jako MPI slave, je rozvícena pouze zelená LED POWER.

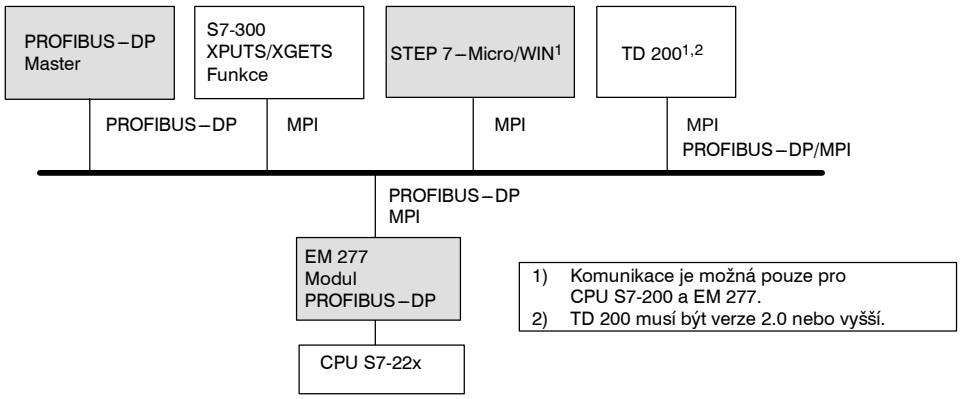
Další vlastnosti konfigurace

Modul EM 277 PROFIBUS–DP je možné použít jako komunikační rozhraní pro ostatní zařízení MPI master, bez ohledu na to, zda se používá jako PROFIBUS–DP slave. Modul může zajistit spojení mezi S7-300/400 a S7-200 pomocí funkcí XGET/XPUT v programu S7-300/400.
STEP 7–Micro/WIN a síťová karta (jako například CP5611) používající sadu parametrů MPI nebo PROFIBUS, panel operátora (OP) nebo TD 200 (verze 2.0 nebo novější, objednací číslo 6ES7 272–0AA20–0YA0) mohou komunikovat s automatem S7-200 prostřednictvím modulu EM 277 PROFIBUS–DP.

Modul EM 277 PROFIBUS–DP podporuje maximálně šest spojení (šest zařízení) kromě zařízení DP master. Jedno spojení je rezervováno pro programovací zařízení (PG) a jedno je rezervováno pro operátorský panel operátora (OP). Ostatní čtyři spojení mohou být použita libovolným zařízením MPI master. Aby mohl modul EM 277 PROFIBUS–DP komunikovat s více zařízeními master, musí všechna tato zařízení pracovat se stejnou přenosovou rychlostí. Jedna možná konfigurace sítě je znázorněna na obrázku A-27.

Pokud je modul EM 277 PROFIBUS–DP použit pro komunikaci MPI, musí MPI master, pro všechny zprávy posílané automatu S7-200, používat adresu stanice modulu, ke kterému je připojen. Zprávy MPI posílané modulu EM 277 PROFIBUS–DP jsou dále předávány automatu S7-200.

Modul EM 277 PROFIBUS–DP je typu slave a nemůže být použit pro komunikaci mezi programovatelnými automaty S7-200, které používají funkce NETR a NETW. Modul EM 277 PROFIBUS–DP také není možné použít pro komunikaci v režimu Freeport.



Obr. A-27 Síť PROFIBUS – DP/MPI

Databázový soubor zařízení: GSD

Různá zařízení PROFIBUS mají různé vlastnosti. Tyto charakteristiky se liší co do funkčnosti (například počet vstupních/výstupních signálů a diagnostických zpráv) nebo parametrů sběrnice, jako je například přenosová rychlost a monitorování času. Dané parametry jsou pro každý typ zařízení a výrobce jiné a obvykle jsou dokumentovány v technickém manuálu. Aby se dosáhlo jednoduché konfigurace PROFIBUS, jsou vlastnosti konkrétního zařízení specifikovány v elektronické formě nazývané databázový soubor zařízení neboli soubor GSD. Konfigurační nástroje založené na souborech GSD umožňují jednoduchou integraci zařízení od různých výrobců do jedné sítě.

Databázový soubor zařízení obsahuje vyčerpávající popis charakteristik zařízení v přesně definovaném fomátu. Soubory GSD jsou výrobcem vypracovány pro každý typ zařízení a jsou k dispozici pro uživatele PROFIBUS. Soubor GSD umožňuje konfiguračnímu systému načíst vlastnosti zařízení PROFIBUS a použít tyto informace při konfiguraci sítě.

Nejnovější verze softwaru COM PROFIBUS nebo STEP 7 obsahují konfigurační soubory pro modul EM 277 PROFIBUS – DP. Pokud vaše verze softwaru neobsahuje konfigurační soubor pro EM 277, je nejnovější soubor GSD (SIEM089D.GSD) k dispozici na internetové adrese www.profibus.com.

Pokud používáte zařízení typu master od jiného výrobce než Siemens, najdete potom způsob konfigurace tohoto zařízení master pomocí souboru GSD v dokumentaci dodávané výrobcem.

```
;================================================
; GSD File for the EM 277 PROFIBUS-DP with a DPC31
; MLFB  : 6ES7 277-0AA2.-0XA0
; DATE  : 26-March-2001
;================================================
#Profibus_DP
;General parameters
GSD_Revision          = 1
Vendor_Name           = "Siemens"
Model_Name            = "EM 277 PROFIBUS-DP"
Revision              = "V1.02"
Ident_Number          = 0x089D
Protocol_Ident        = 0
Station_Type          = 0
FMS_supp              = 0
Hardware_Release      = "1.00"
Software_Release      = "1.02"
9,6_supp              = 1
19,2_supp              = 1
45,45_supp              = 1
93,75_supp              = 1
187,5_supp              = 1
500_supp              = 1
1.5M_supp             = 1
3M_supp               = 1
6M_supp               = 1
12M_supp              = 1
MaxTsdr_9.6           = 60
MaxTsdr_19.2          = 60
MaxTsdr_45.45         = 250
MaxTsdr_93.75         = 60
MaxTsdr_187.5         = 60
MaxTsdr_500           = 100
MaxTsdr_1.5M          = 150
MaxTsdr_3M            = 250
MaxTsdr_6M            = 450
MaxTsdr_12M           = 800
Redundancy            = 0
Repeater_Ctrl_Sig     = 2
24V_Pins              = 2

; Slave-Specification:
OrderNumber="6ES7 277-0AA2.-0XA0"
Periphery="SIMATIC S5"
Slave_Family=10@TdF@SIMATIC

Freeze_Mode_supp      = 1
Sync_Mode_supp        = 1
Set_Slave_Add_Supp    = 0
Auto_Baud_supp        = 1
Min_Slave_Intervall   = 1
Fail_Safe             = 0
Max_Diag_Data_Len     = 6
Modul_Offset          = 0
Modular_Station       = 1
Max_Module            = 1
Max_Input_len         = 128
Max_Output_len        = 128
Max_Data_len          = 256

; UserPrmData-Definition
ExtUserPrmData=1 "I/O Offset in the V-memory"
Unsigned16 0 0-10239
EndExtUserPrmData
; UserPrmData: Length and Preset:
User_Prm_Data_Len=3
User_Prm_Data= 0,0,0
Max_User_Prm_Data_Len=3
Ext_User_Prm_Data_Const(0)=0x00,0x00,0x00
Ext_User_Prm_Data_Ref(1)=1
```

```
;================================================
; Continuation of GSD File
;================================================

; Module Definition List
Module = "2 Bytes Out/ 2 Bytes In      -" 0x31
EndModule
Module = "8 Bytes Out/ 8 Bytes In      -" 0x37
EndModule
Module = "32 Bytes Out/ 32 Bytes In       -"
0xC0,0x1F,0x1F
EndModule
Module = "64 Bytes Out/ 64 Bytes In       -"
0xC0,0x3F,0x3F
EndModule
Module = "1 Word Out/ 1 Word In        -" 0x70
EndModule
Module = "2 Word Out/ 2 Word In        -" 0x71
EndModule
Module = "4 Word Out/ 4 Word In        -" 0x73
EndModule
Module = "8 Word Out/ 8 Word In        -" 0x77
EndModule
Module = "16 Word Out/ 16 Word In      -" 0x7F
EndModule
Module = "32 Word Out/ 32 Word In          -"
0xC0,0x5F,0x5F
EndModule
Module = "2 Word Out/ 8 Word In            -"
0xC0,0x41,0x47
EndModule
Module = "4 Word Out/ 16 Word In           -"
0xC0,0x43,0x4F
EndModule
Module = "8 Word Out/ 32 Word In           -"
0xC0,0x47,0x5F
EndModule
Module = "8 Word Out/ 2 Word In            -"
0xC0,0x47,0x41
EndModule
Module = "16 Word Out/ 4 Word In           -"
0xC0,0x4F,0x43
EndModule
Module = "32 Word Out/ 8 Word In           -"
0xC0,0x5F,0x47
EndModule
Module = "4 Byte buffer I/O            -" 0xB3
EndModule
Module = "8 Byte buffer I/O            -" 0xB7
EndModule
Module = "12 Byte buffer I/O           -" 0xBB
EndModule
Module = "16 Byte buffer I/O           -" 0xBF
EndModule
```

Obr. A-28 Výpis souboru GSD pro modul EM 277 PROFIBUS

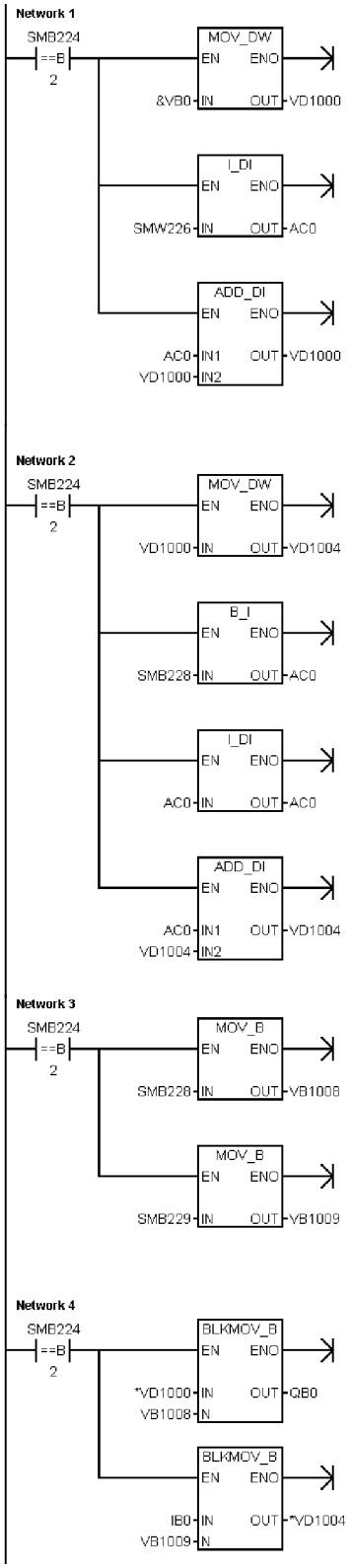
Vzorový program pro komunikaci DP s CPU

Vzorový program ve formátu STL pro modul PROFIBUS–DP umístěný na pozici 0 a pro CPU, který používá informace o DP portu v paměti SM, je uveden níže. Program určuje umístění zásobníku DP ze SMW226 a velikost zásobníku ze SMB228 a SMB229. Tyto informace jsou použity pro kopírování dat ve výstupním DP zásobníku do registru obrazu výstupů CPU. Obdobně jsou data v registru obrazu vstupů CPU kopírována do vstupního zásobníku v paměti V.

V následujícím vzorovém programu pro DP modul na pozici 0 poskytují konfiguraci zařízení DP slave některé oblasti paměti SM. Program používá následující data:

SMW220	Chybový stav DP modulu
SMB224	Stav DP
SMB225	Adresa masteru
SMW226	Offset výstupů v paměti V
SMB228	Počet bytů výstupních dat
SMB229	Počet bytů vstupních dat
VD1000	Ukazatel výstupních dat
VD1004	Ukazatel vstupních dat

Příklad DP komunikace s CPU




```
Network 1       //Výpočet ukazatele výstupních dat.
                //Pokud je v režimu výměny dat:
                //1. Výstupní zásobník má offset od VB0
                //2. Převede offset v paměti V na double integer
                //3. Připočte ho k adrese VB0, aby se získal
                //   ukazatel výstupních dat.
LDB=    SMB224, 2
MOVD    &VB0, VD1000
ITD     SMW226, AC0
+D      AC0, VD1000

Network 2       //Výpočet ukazatele vstupních dat.
                //Pokud je v režimu výměny dat:
                //1. Zkopíruje ukazatel výstupních dat
                //2. Získá počet výstupních bytů
                //3. Připočte ho k ukazateli výstupních dat, aby
                //   se získal spouštěcí ukazatel vstupních dat.
LDB=    SMB224, 2
MOVD    VD1000, VD1004
BTI     SMB228, AC0
ITD     AC0, AC0
+D      AC0, VD1004

Network 3       //Nastaví množství dat, která mají být
                //zkopírována.
                //Pokud je v režimu výměny dat:
                //1. Získá počet výstupních bytů, které mají být
                //   zkopírovány.
                //2. Získá počet vstupních bytů, které mají být
                //   zkopírovány.
LDB=   SMB224, 2
MOVB   SMB228, VB1008
MOVB   SMB229, VB1009

Network 4       //Přesune výstupy zařízení master na CPU
                //výstupy. Kopíruje výstupy CPU do
                //výstupů master. Pokud je v režimu výměny
                //dat:
                //1. Zkopíruje výstupy zařízení master na
                //   výstupy CPU.
                //2. Zkopíruje vstupy CPU na vstupy zařízení
                //   master.
LDB=    SMB224, 2
BMB     *VD1000, QB0, VB1008
BMB     IB0, *VD1004, VB1009
```

Specifikace modemového modulu EM 241

Tabulka A-43 Objednací číslo modemového modulu EM 241

Objednací číslo	Rozšiřovací modul	Vstupy EM	Výstupy EM	Odnímatelná svorkovnice
6ES7 241–1AA22–0XA0	Modemový modul EM 241	–	8[1]	Ne

1 Osm výstupů Q je použito jako logické řídicí prvky pro ovládání modemu a neřídí přímo žádné vnější signály.

Tabulka A-44 Obecná specifikace modemového modulu EM 241

Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	Požadavky V DC +5 V DC	Požadavky V DC +24 V DC
6ES7 241–1AA22–0XA0	Modemový modul EM 241	71,2 x 80 x 62	190 g	2,1 W	80 m A	70 mA

Tabulka A-45 Specifikace modemového modulu EM 241

Obecná	6ES7 241–1AA22–0XA0
Telefonní spojení	
Napěťové oddělení (telefonní linka od logiky a napájení)	1500 V AC (galvanicky)
Fyzické připojení	RJ11 (6 poloh, 4 vodiče)
Standardy modemu	Bell 103, Bell 212, V.21, V.22, V.22 bis, V.23c, V.32, V.32 bis, V.34 (implicitní)
Bezpečnostní prvky	Vstupní heslo Zpětné volání
Volba	Pulzní nebo tónová
Protokoly pro přenos zpráv	Číselné TAP (alfanumerický) UCP příkazy 1, 30, 51
Průmyslové protokoly	Modbus PPI
Požadavky na zdroj 24 V DC	
Rozsah napětí	20,4 až 28,8 V DC
Napěťové oddělení (napájení pole od logiky)	500 V AC po dobu 1 minuty

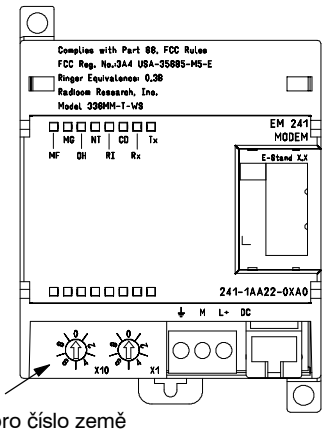
Modemový modul EM 241 nahrazuje funkci externího modemu připojeného ke komunikačnímu portu CPU. Je–li modul EM 241 instalován v systému S7-200, vše, co je zapotřebí pro komunikaci s CPU ze vzdáleného místa, je osobní počítač s externím modemem a STEP 7–Micro/WIN.

Informace o konfiguraci najdete v kapitole 7 – Síťová komunikace. Programování a vlastnosti modulu pro pokročilé najdete v kapitole 10 – Tvorba programu pro modemový modul.



Průvodce modemem

Pro konfiguraci modemového modulu EM 241 můžete použít STEP 7–Micro/WIN a jeho funkci ”Průvodce modemem”. Více informací o ”Průvodci modemem” najdete v kapitole 10.



Přepínač pro číslo země

Obr. A-29 Schéma svorkovnice modemového modulu EM 241

CPU S7-200, které podporují inteligentní moduly

Modemový modul EM 241 je inteligentní rozšiřovací modul určený pro spolupráci s CPU S7-200 uvedenými v tabulce A-46.

Tabulka A-46 Kompatibilita modemového modulu EM 241 s CPU S7-200

CPU	Popis
Model CPU 222 verze 1.10 nebo vyšší	CPU 222 DC/DC/DC a CPU 222 AC/DC/relé
Model CPU 224 verze 1.10 nebo vyšší	CPU 224 DC/DC/DC a CPU 224 AC/DC/relé
CPU 224XP ver. 2.0 nebo vyšší	CPU 224XP DC/DC/DC a CPU 224XP DC/DC/relé
Model CPU 226 verze 1.00 nebo vyšší	CPU 226 DC/DC/DC a CPU 226 AC/DC/relé

Instalace EM 241

Chcete – li instalovat EM 241, proveďte tyto kroky:

1. Nasaďte EM 241 na DIN lištu a zasuňte plochý kabel rozšiřovací sběrnice.
2. Připojte 24 V DC ze zdroje snímače v CPU nebo z vnějšího zdroje a připojte zemnicí svorku k ochrannému uzemnění.
3. Připojte telefonní linku jackem RJ11.
4. Nastavte přepínače pro kód země podle tabulky A-47. Přepínače musíte nastavit před zapnutím napájení CPU, aby byl načten správný kód země.
5. Zapněte napájení CPU. Měla by se rozsvítit zelená LED MG (Module Good – modul v pořádku).

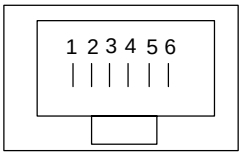
Teď je modul EM 241 připraven komunikovat.

Tabulka A-47 Kódy zemí podporované EM 241

Kód	Země	Standard Telecom
00	Austrálie	ACA TS – 002
01	Rakousko	CTR21
02	Belgie	CTR21
05	Kanada	IC CS03
06	Čína	GB3482
08	Dánsko	CTR21
09	Finsko	CTR21
10	Francie	CTR21
11	Německo	CTR21
12	Řecko	CTR21
16	Irsko	CTR21
18	Itálie	CTR21
22	Lucembursko	CTR21
25	Nizozemí	CTR21
26	Nový Zéland	PTC 200
27	Norsko	CTR21
30	Portugalsko	CTR21
34	Španělsko	CTR21
35	Švédsko	CTR21
36	Švýcarsko	CTR21
38	Velká Británie	CTR21
39	USA	FCC část 68

Jack RJ11

Obrázek A-30 ukazuje jack RJ11 zblízka. Můžete použít také adaptéry pro jiné standardní telefonní konektory. Více informací najdete v dokumentaci pro přechodový konektor.



Vývod	Popis
3	Vyzvánění
4	Hovor

Je povoleno obrácené zapojení.

Obr. A-30 Pohled na jack RJ11

Upozornění

Blesk nebo jiná nečekená vysoká napětí na telefonní linky, tedy stejně, jako jsou běžně prodávány na ochranu modemů osobních počítačů.

Použijte komerčně nabízenou přepěťovou ochranu telefonní linky, stejnou, jako jsou běžně prodávány na ochranu modemů osobních počítačů. Při výboji může dojít k poškození přepěťové ochrany. Vyberte si přepěťovou ochranu s indikátorem, který ukazuje, že je ochrana funkční.

Pravidelně kontrolujte přepěťovou ochranu, aby bylo jisté, že je trvale modul EM 241chráněn.

Specifikace polohovacího modulu EM 253

Tabulka A-48 Objednací číslo polohovacího modulu EM 253

Objednací číslo	Rozšiřovací modul	Vstupy EM	Výstupy EM	Odnímatelná svorkovnice
6ES7 253–1AA22–0XA0	Polohovací modul EM 253	–	8[1]	Ano

[1] Osm výstupů Q je použito jako logické řídicí prvky a neřídí přímo žádné vnější signály.

Tabulka A-49 Obecná specifikace polohovacího modulu EM 253

Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	Požadavky V DC +5 V DC	+24 V DC
6ES7 253–1AA22–0XA0	Polohovací modul EM 253	71,2 x 80 x 62	0,190 kg	2,5 W	190 mA	Viz níže

Tabulka A-50 Specifikace polohovacího modulu EM 253

Obecné	6ES7 253–1AA22–0XA0
Vlastnosti vstupu	
Počet vstupů	**5 vstupů**
Typ vstupu	Aktivní/pasivní (pasivní IEC typ 1, s výjimkou ZP)
Vstupní napětí Maximální trvalé povolené STP, RPS, LMT+, LMT– ZP Špička (všechny vstupy) Jmenovitá hodnota STP, RPS, LMT+, LMT– ZP Signál "1" log. (minimální) STP, RPS, LMT+, LMT– ZP Signál "0" log. (maximální) STP, RPS, LMT+, LMT– ZP	 30 V DC 30 V DC při 20 mA, maximální 35 V DC po 0,5 s 24 V DC při 4 mA, jmenovité 24 V DC při 15 mA, jmenovité 15 V DC při 2,5 mA, minimální 3 V DC při 8,0 mA, minimální 5 V DC při 1 mA, maximální 1 V DC při 1 mA, maximální
Napěťové oddělení (okolí od logiky) Optické oddělení (galvanické) Oddělení po skupinách	 500 V AC po dobu 1 minuty 1 bod pro STP, RPS a ZP 2 body pro LMT+ a LMT–
Doby zpoždění na vstupu STP, RPS, LMT+, LMT– ZP (počitatelná šířka pulzu)	 0,2 ms až 12,8 ms, volitelné uživatelem 2 σs minimálně
Připojení dvouvodičového snímače přiblížení (Bero) Povolený svodový proud	 1 mA, maximální
Délka kabelu Nestíněný STP, RPS, LMT+, LMT– ZP Stíněný STP, RPS, LMT+, LMT– ZP	 30 metrů Nedoporučuje se 100 metrů 10 metrů
Počet současně zapnutých vstupů	Všechny při 55 ∀C (horizontální), všechny při 45 ∀C (vertikální)

Tabulka A-50 Specifikace polohovacího modulu EM 253, pokračování

Obecné	6ES7 253–1AA22–0XA0	
Vlastnosti výstupů		
Počer integrovaných výstupů	6 výstupů (4 signály)	
Typ výstupu P0+, P0–, P1+, P1– P0, P1, DIS, CLR	 Ovladač RS422/485 Otevřený kolektor	
Výstupní napětí P0, P1, ovladače RS–422, rozdílové výstupní napětí Otevřený obvod Diody optoelektronického vazebního členu s 200T sériovým odporem 100T zatížení 54T zatížení P0, P1, DIS, CLR otevřený kolektor doporučené napětí, otevřený obvod povolené napětí, otevřený obvod Pohlcovaný proud Odpor v zapnutém stavu Svodový proud ve vypnutém stavu, 30 V DC Vnitřní zdvihací odpor, výstupní kolektor do T1	 3,5 V běžné 2,8 V minimální 1,5 V minimální 1,0 V minimální 5 V DC, k dispozici z modulu 30 V DC[1] 50 mA, maximální 15 T maximální 10 σA maximální 3,3 kT[2]	
Výstupní proud Počet výstupních skupin Současně zapnuté výstupy Svodový proud na výstup P0, P1, DIS, CLR Ochrana proti přetížení	 1 Všechny při 55 ϒC (horizontální), všechny při 45 ϒC (vertikální) 10 σA maximální Ne	
Napěťové oddělení (okolí od logiky) Optické oddělení (galvanické)	 500 V AC po dobu 1 minuty	
Zpoždění na výstupu DIS, CLR: Vyp. na zap. / Zapn. na vyp. (Off to On / On to Off)	 30 σs maximální	
Zkreslení pulzu P0, P1, výstupy, ovladače RS–422, 100 T vnější zatížení P0, P1 výstupy, otevřený kolektor, 5 V / 470 T vnější zatěž	 75 ns maximální 300 ns maximální	
Spínací frekvence P0+, P0–, P1+, P1–, P0 a P1	 200 kHz	
Délka kabelu Nestíněný Stíněný	 Nedoporučuje se 10 metrů	
Napájecí zdroj		
Napájecí napětí L+ Výstup napájení logiky Napájecí proud L+ při zatížení 5 V DC Zatěžovací proud 0 mA (bez zatížení) 200 mA (jmenovité zatěž)	11 až 30 V DC (třída 2, omezený výkon nebo napájení snímačů z PLC) +5 V DC +/– 10 %, 200 mA maximální <u>Vstup 12 V DC</u> 120 mA 300 mA	 <u>Vstup 24 V DC</u> 70 mA 130 mA
Napěťové oddělení Napájení L+ od logiky Napájení L+ od vstupů Napájení L+ od výstupů	 500 V AC po dobu 1 minuty 500 V AC po dobu 1 minuty Nemá	
Obrácená polarita	Vstup L+ a výstup +5V jsou chráněny diodou. Přivedení kladného napětí na kteroukoliv svorku M může mít, vzhledem ke spojením výstupu, mít za následek průtok proudu vedoucí k poškození.	

1 Provoz výstupů s otevřeným kolektorem nad 5 V DC může zvýšit vysokofrekvenční emise nad povolené mezní hodnoty. Pro váš systém nabo kabeláž mohou být nutná měření emise vysokofrekvenčního rušení.

2 Podle typu přijímače pulzů a kabelu, může přídavný vnější odpor zlepšit kvalitu pulzového signálu a odolnost proti rušení.

CPU S7-200, které podporují inteligentní moduly

Polohovací modul EM 253 je inteligentní rozšiřovací modul určený pro spolupráci s CPU S7-200 uvedenými v tabulce A-51.

Tabulka A-51 Kompatibilita polohovacího modulu EM 253 s CPU S7-200

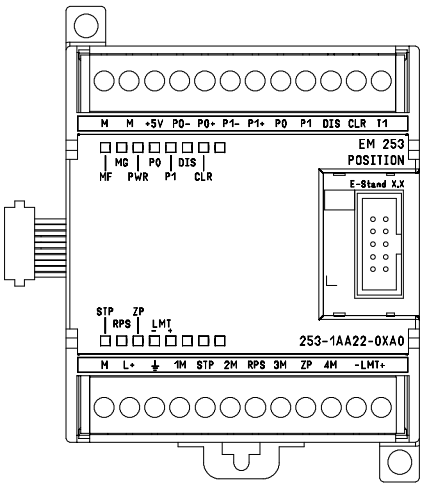
CPU	Popis
Model CPU 222 verze 1.10 nebo vyšší	CPU 222 DC/DC/DC a CPU 222 AC/DC/relé
Model CPU 224 verze 1.10 nebo vyšší	CPU 224 DC/DC/DC a CPU 224 AC/DC/relé
CPU 224XP verze 2.0 nebo vyšší	CPU 224XP DC/DC/DC a CPU 224XP DC/DC/relé
Model CPU 226 verze 1.00 nebo vyšší	CPU 226 DC/DC/DC a CPU 226 AC/DC/relé

Stavové LED polohovacího modulu EM 253

Stavové LED polohovacího modulu jsou uvedeny v tabulce A-52.

Tabulka A-52 Stavové LED polohovacího modulu

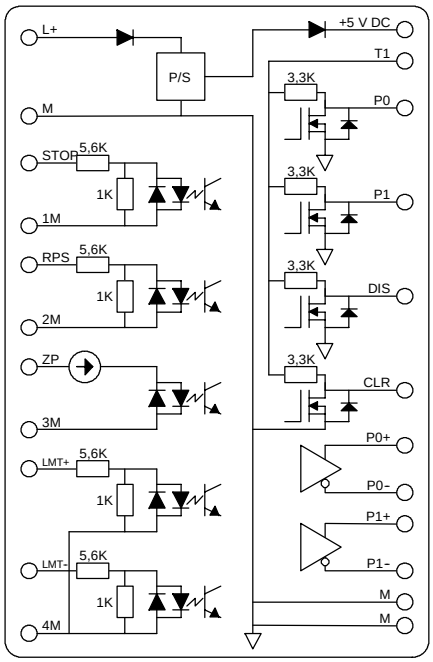
Lokální I/O	Kontrolka	Barva	Popis funkce
–	MF	Červená	Svítí, když modul zjistí kritickou chybu.
–	MG	Zelená	Svítí, když modul nemá žádnou poruchu, a bliká rychlostí 1 Hz, když je zjištěna chyba konfigurace.
–	PWR	Zelená	Svítí, když je na svorky modulu L+ a M přiváděno napětí 24 V DC.
Vstup	STP	Zelená	Svítí, když je sepnutý vstup ”stop”.
Vstup	RPS	Zelená	Svítí, když je sepnutý vstup přepínače referenčního bodu.
Vstup	ZP	Zelená	Svítí, když je sepnutý vstup nulového impulzu.
Vstup	LMT–	Zelená	Svítí, když je sepnutý vstup záporné mezní hodnoty.
Vstup	LMT+	Zelená	Svítí, když je sepnutý vstup kladné mezní hodnoty.
Výstup	P0	Zelená	Svítí, když výstup P0 vysílá pulzy.
Výstup	P1	Zelená	Svítí, když výstup P1 vysílá pulzy nebo když tento výstup indikuje kladný pohyb.
Výstup	DIS	Zelená	Svítí, když je výstup DIS aktivní.
Výstup	CLR	Zelená	Svítí, když je aktivní výstup pro smazání čítače odchylek.



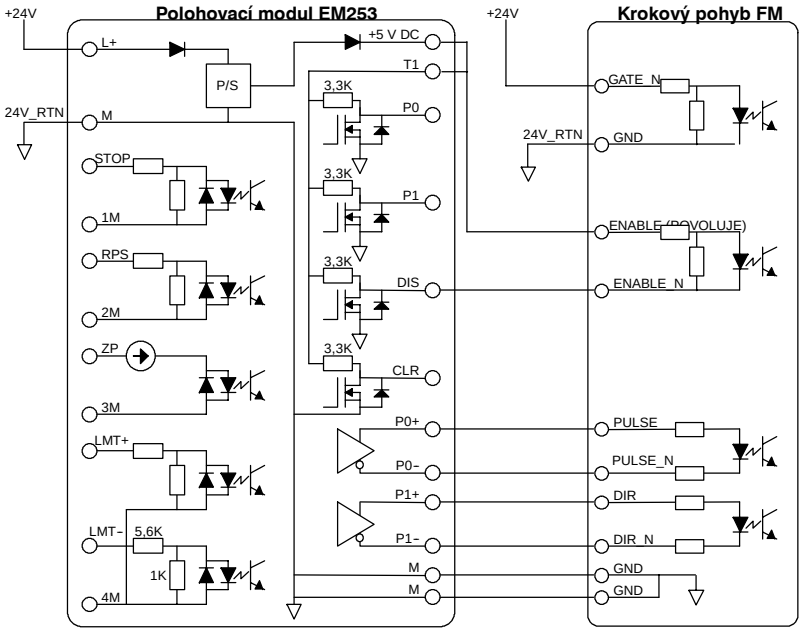
Obr. A-31 Polohovací modul EM 253

Schéma zapojení

Na následujících schematických obrázcích nejsou svorky uváděny v pořadí za sebou. Uspořádání svorek najdete na obrázku A-31.

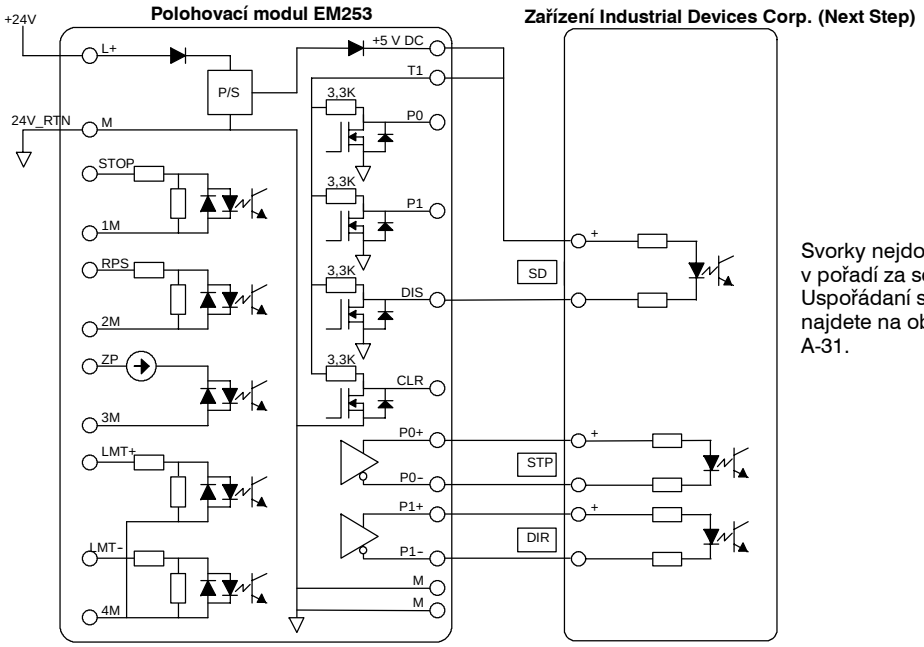


Obr. A-32 Vnitřní schéma pro vstupy a výstupy polohovacího modelu EM 253



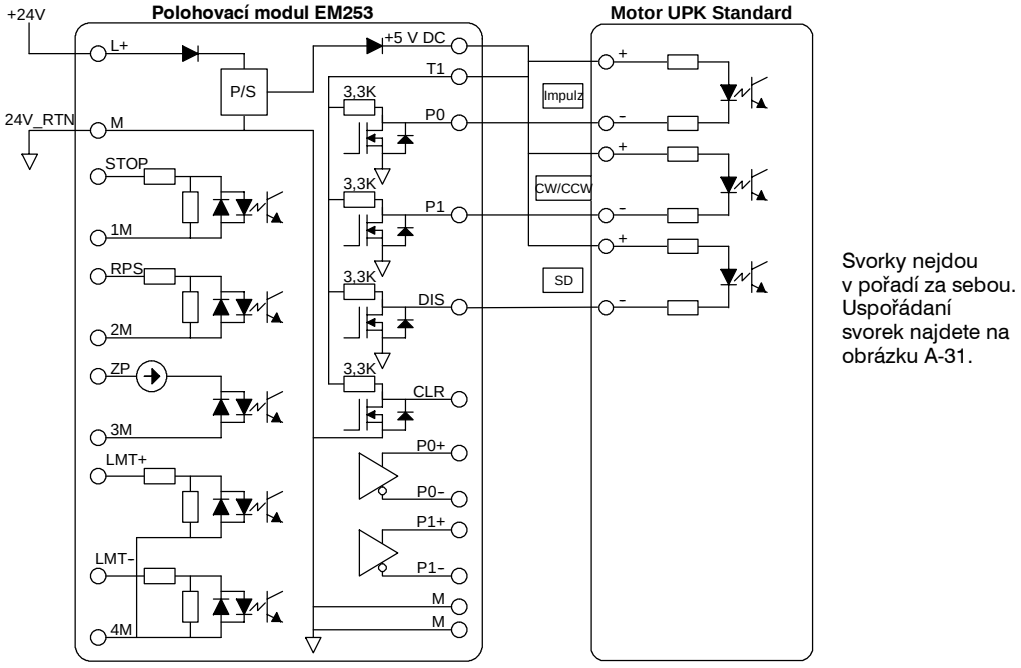
Svorky nejdou v pořadí za sebou. Uspořádání svorek najdete na obrázku A-31.

Obr. A-33 Připojení polohovacího modulu EM 253 ke krokovému pohonu SIMATIC FM



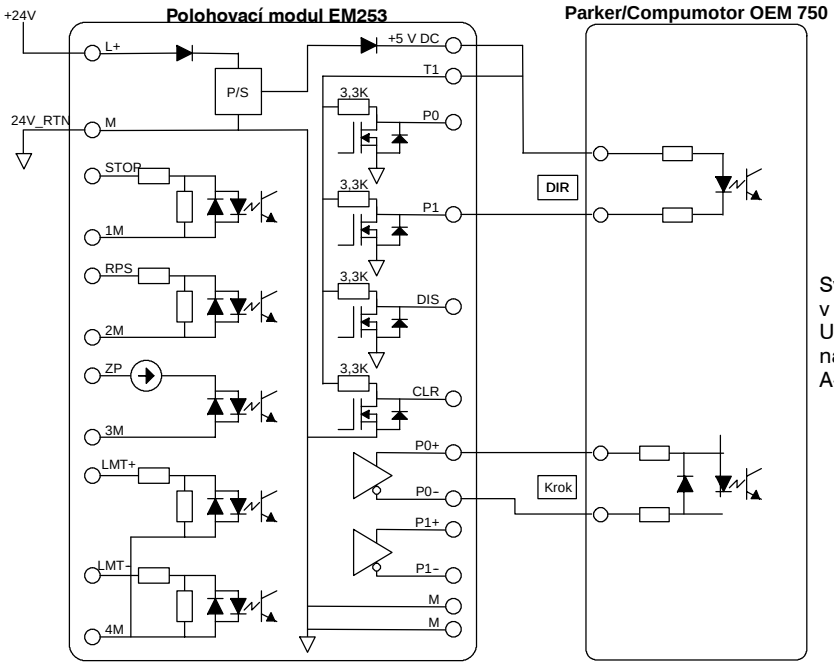
Svorky nejdou v pořadí za sebou. Uspořádaní svorek najdete na obrázku A-31.

Obr. A-34 Připojení polohovacího modulu EM 253 k zařízení Industrial Devices Corp. (Next Step)



Svorky nejdou v pořadí za sebou. Uspořádaní svorek najdete na obrázku A-31.

Obr. A-35 Připojení polohovacího modulu EM 253 k pohonu UPK Standard



Svorky nejdou v pořadí za sebou. Uspořádaní svorek najdete na obrázku A-31.

Obr. A-36 Připojení polohovacího modulu EM 253 k pohonu Parker/Compumotor OEM 750

Specifikace ethernetového modulu (CP 243 – 1)

Tabulka A-53 Objednací číslo ethernetového modulu (CP 243 – 1)

Objednací číslo	Rozšiřovací modul	Vstupy EM	Výstupy EM	Odnímatelná svorkovnice
6GK7 243 – 1EX00 – 0XE0	Ethernetový modul (CP 243 – 1)	–	8[1]	Ne

[1] Osm výstupů Q je použito jako logické řídicí prvky funkce Ethernet a neřídí přímo žádné vnější signály.

Tabulka A-54 Obecná specifikace ethernetového modulu (CP 243 – 1)

Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	Požadavek VDC	
					+5 VDC	+24 VDC
6GK7 243 – 1EX00 – 0XE0	Ethernetový modul (CP 243 – 1)	71,2 x 80 x 62	cca 150 g	1,75 W	55 mA	60 mA

Tabulka A-55 Specifikace ethernetového modulu (CP 243 – 1)

Obecné	6GK7 243 – 1EX00 – 0XE0
Přenosová rychlost	**10 Mbitů/s a 100 Mbitů/s**
Velikost paměti Flash	1 Mbyte
Velikost paměti SDRAM	8 Mbytů
Rozhraní Připojení k průmyslovému Ethernetu (10/100 Mbit/s)	Objímka RJ45 s 8 vývody
Vstupní napětí	20,4 až 28,8 V DC
Maximální počet spojení	Maximálně 8 spojení s S7 (XPUT/XGET a READ/WRITE) plus 1 spojení se STEP 7 – Micro/WIN na ethernetový modul (CP 243 – 1)[2]
Spouštěcí čas nebo čas restartu po nulování	Cca 10 sekund
Množství uživatelských dat	Jako klient: až 212 bytů pro XPUT/XGET Jako server: až 222 bytů pro XGET nebo READ až 212 bytů pro XPUT nebo WRITE

[2] K jedné CPU S7 – 200 by měl být připojen pouze jeden ethernetový modul (CP 243 – 1).

Ethernetový modul (CP 243 – 1) je komunikační procesor použitý pro spojení systému S7-200 s průmyslovým Ethernetem (IE). S7-200 je možné dálkově konfigurovat, programovat a diagnostikovat prostřednictvím Ethernetu a s použitím STEP 7 Micro/WIN. Přes Ethernet může S7-200 komunikovat s jiným programovatelným automatem S7 – 200, S7-300 nebo S7-400. Může také komunikovat s OPC serverem.

Průmyslový Ethernet je určen pro použití v průmyslu. Je možné ho použít s technologií průmyslové kroucené dvojlinky s potlačením šumu (ITP) nebo technologií průmyslové standardní kroucené dvojlinky (TP). Průmyslový Ethernet je možné realizovat tak, že poskytuje mnoho různých užití podle potřeb konkrétní aplikace, jako je spínání, vysokorychlostní redundance, rychlá připojení a sítě s redundancí. Při použití ethernetového modulu (CP 243 – 1) je programovatelný automat S7-200 kompatibilní s mnoha různými výrobky, které Ethernet podporují.

CPU S7-200, které podporují inteligentní moduly

Ethernetový modul (CP 243–1) je inteligentní rozšiřovací modul určený pro spolupráci s CPU S7-200 uvedenými v tabulce A-46.

Tabulka A-56 Kompatibilita ethernetového modulu (CP 243–1) s CPU S7-200

CPU	Popis
Model CPU 222 verze 1.10 nebo vyšší	CPU 222 DC/DC/DC a CPU 222 AC/DC/relé
Model CPU 224 verze 1.10 nebo vyšší	CPU 224 DC/DC/DC a CPU 224 AC/DC/relé
Model CPU 224XP verze 2.00 nebo vyšší	CPU 224XP DC/DC/DC a CPU 224XP AC/DC/relé
Model CPU 226 verze 1.00 nebo vyšší	CPU 226 DC/DC/DC a CPU 226 AC/DC/relé

Ethernetový modul (CP 243–1) se dodává s nastavenou, celosvětově jedinečnou MAC adresou, kterou není možné měnit.

Funkce

Ethernetový modul (CP 243–1) řídí nezávisle přenášení dat v rámci průmyslového Ethernetu.

- Komunikace je založena na TCP/IP
- Při komunikaci mezi CPU S7-200 a jinými řídicími systémy S7 nebo PC po Ethernetu jsou komunikační služby k dispozici jako klient a server. Je možné provozovat až osm spojených zařízení.
- Realizace počítačových aplikací je možná díky začlenění serveru S7-OPC
- Ethernetový modul (CP 243–1) umožňuje programovacímu softwaru S7-200, STEP 7–Micro/WIN, přímý přístup k S7-200 prostřednictvím Ethernetu


Ethernet

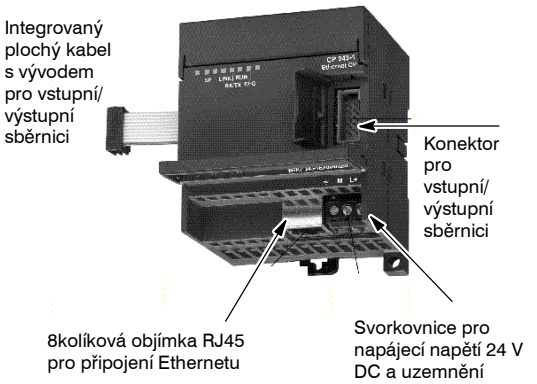
Konfigurace

Pro konfiguraci ethernetového modulu (CP 243–1) na připojení programovatelného automatu S7-200 k ethernetové síti můžete použít "Průvodce Ethernetem" v STEP 7–Micro/WIN. "Průvodce Ethernetem" vám pomůže definovat parametry pro ethernetový modul (CP 243–1), a pak vloží instrukce pro konfiguraci do složky s instrukcemi pro váš projekt. Chcete–li spustit "Průvodce Ethernetem", zvolte příkaz menu **Nástroje > Průvodce Ethernetem**. Průvodce používá následující informace: Adresa IP, maska pro dílčí síť, adresa mezisíťového počítače a typ komunikačního spojení.

Připojení

Ethernetový modul (CP 243–1) má následující připojení. Připojení jsou umístěna pod kryty předních dvířek.

- Svorkovnice pro napájecí napětí 24 V DC a uzemnění
- 8kolíková objímka RJ45 pro připojení Ethernetu
- Zástrčkový konektor pro vstupní/výstupní sběrnici
- Integrovaný plochý kabel s vývodem pro vstupní/výstupní sběrnici



Obr. A-37 Připojení ethernetového modulu (CP 243–1)

Další informace

Více informací o ethernetovém modulu (CP 243–1) najdete v *Technickém manuálu SIMATIC NET CP 243–1 Komunikační procesor pro průmyslový Ethernet.*

Specifikace internetového modulu (CP 243 – 1 IT)

Tabulka A-57 Objednací číslo internetového modulu (CP 243 – 1 IT)

Objednací číslo	Rozšiřovací modul	Vstupy EM	Výstupy EM	Odnímatelná svorkovnice
6GK7 243 – 1GX00 – 0XE0	Internetový modul (CP 243 – 1 IT)	–	8[1]	Ne

[1] Osm výstupů Q je použito jako logické řídicí prvky funkce IT a neřídí přímo žádné vnější signály.

Tabulka A-58 Obecná specifikace internetového modulu (CP 243 – 1 IT)

Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	Požadavky V DC	
					+5 V DC	+24 V DC
6GK7 243 – 1GX00 – 0XE0	Internetový modul (CP 243 – 1 IT)	71,2 x 80 x 62	cca 150 g	1,75 W	55 mA	60 mA

Tabulka A-59 Specifikace internetového modulu (CP 243 – 1 IT)

Obecné	6GK7 243 – 1GX00 – 0XE0
Přenosová rychlost	10 Mbitů/s a 100 Mbitů/s
Velikost paměti Flash	8 Mbytů jako ROM pro firmware internetového modulu (CP 243 – 1 IT), 8 Mbytů jako RAM pro systém souborů
Velikost paměti SDRAM	16 Mbytů
Zaručená životnost paměti Flash pro systém souborů	1 milion operací zápisu nebo mazání
Rozhraní Připojení k průmyslovému Ethernetu (10/100 Mbit/s)	 Objímka RJ45 s 8 vývody
Vstupní napětí	20,4 až 28,8 V DC
Maximální počet spojení	Maximálně 8 spojení s S7 (XPUT/XGET a READ/WRITE) plus 1 spojení se STEP 7 – Micro/WIN na internetový modul (CP 243 – 1 IT)[1]
Maximální počet spojení IT	1 pro FTP server 1 pro FTP klienta 1 pro klienta e-mailu 4 pro HTTP spojení
Spouštěcí čas nebo čas restartu po nulování	Cca 10 sekund
Množství uživatelských dat	Klient: až 212 bytů pro XPUT/XGET Server: až 222 bytů pro XGET nebo READ až 212 bytů pro XPUT nebo WRITE
Velikost pro e – mail, maximální	1024 znaků
Systém souborů: Délka cesty včetně velikosti souboru a označení disku Délka názvu souboru Hloubka vnoření adresáře	 maximálně 254 znaků maximálně 99 znaků maximálně 49
Porty serveru, které jsou k dispozici: HTTP kanál pro FTP příkazy kanály FTP dat pro FTP server navázání spojení S7 S7 server	 80 21 3100 až 3199 102 3000 až 3008

[1] K jedné CPU S7 – 200 by měl být připojen pouze jeden internetový modul (CP 243 – 1 IT).

Internetový modul (CP 243–1 IT) je komunikační procesor použitý pro spojení systému S7-200 s průmyslovým Ethernetem (IE). S7-200 je možné dálkově konfigurovat, programovat a diagnostikovat prostřednictvím Ethernetu a s použitím STEP 7 Micro/WIN. Přes Ethernet může S7-200 komunikovat s jiným programovatelným automatem S7–200, S7-300 nebo S7-400. Může také komunikovat s OPC serverem.

IT funkce internetového modulu (CP 243–1 IT) tvoří základ pro monitorování automatizačních systémů a v případě potřeby i manipulaci s nimi pomocí webového prohlížeče v PC připojeném k síti. Ze systému je možné posílat diagnostické zprávy prostřednictvím e-mailu. Použijeme–li IT funkce, je výměna celých souborů s jiným počítačem a systémy programovatelného automatu snadná.

Průmyslový Ethernet je síť pro úroveň řízení procesu a úroveň GSM otevřeného komunikačního systému SIMATIC NET. Fyzicky je průmyslový Ethernet elektrická síť založená na stíněných koaxiálních vedeních, kroucených dvojlinkách a optická síť s optickými kabely. Průmyslový Ethernet je definován mezinárodní normou IEEE 802.3.

CPU S7-200, které podporují inteligentní moduly

Internetový modul (CP 243–1 IT) je inteligentní rozšiřovací modul určený pro spolupráci s CPU S7-200 uvedenými v tabulce A-46.

Tabulka A-60 Kompatibilita internetového modulu (CP 243–1 IT) s CPU S7-200

CPU	Popis
Model CPU 222 verze 1.10 nebo vyšší	CPU 222 DC/DC/DC a CPU 222 AC/DC/relé
Model CPU 224 verze 1.10 nebo vyšší	CPU 224 DC/DC/DC a CPU 224 AC/DC/relé
Model CPU 224XP verze 2.00 nebo vyšší	CPU 224XP DC/DC/DC a CPU 224XP AC/DC/relé
Model CPU 226 verze 1.00 nebo vyšší	CPU 226 DC/DC/DC a CPU 226 AC/DC/relé

Internetový modul (CP 243–1 IT) má následující vlastnosti:

- Internetový modul (CPU 243–1 IT) je plně kompatibilní s ethernetovým modulem (CP 243–1). Uživatelské programy vytvořené pro ethernetový modul (CP 243–1) mohou být také používány internetovým modulem (CP 243–1 IT).

Internetový modul (CP 243–1 IT) se dodává s nastavenou celosvětově jedinečnou MAC adresou, kterou není možné měnit.



Tip
K jedné CPU S7-200 by měl být připojen pouze jeden internetový modul (CP 243–1 IT). Pokud se připojí více než jeden internetový modul (CP 243–1 IT), je možné, že CPU S7-200 nebude pracovat správně.

Funkce

Internetový modul (CP 243–1 IT) má následující funkce:

- Komunikace S7 je založena na TCP/IP
- IT komunikace
- Konfigurace
- Obvod sledování funkce
- Schopnost nastavených MAC adres (48bitová hodnota) být adresovány


Internet

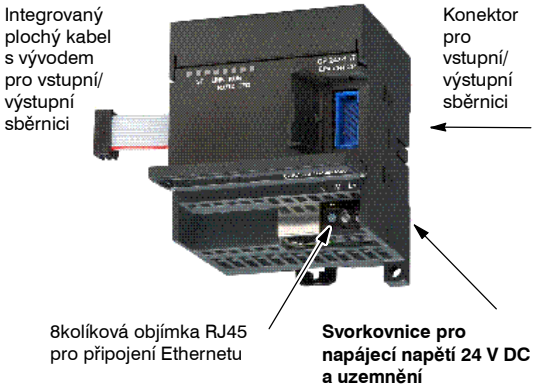
Konfigurace

Pro konfiguraci internetového modulu (CP 243–1 IT) na připojení programovatelného automatu S7-200 k ethernetové/internetové síti můžete použít ”Průvodce Internetem” v STEP 7–Micro/WIN. Internetový modul (CP 243–1 IT) má navíc funkci webového serveru, kterou je možné konfigurovat pomocí ”Průvodce Internetem”. Chcete–li spustit ”Průvodce Internetem”, zvolte příkaz menu **Nástroje > Průvodce Internetem**.

Připojení

Internetový modul (CP 243–1 IT) má následující spoje: Připojení jsou umístěna pod kryty předních dvířek.

- ❏ Svorkovnice pro napájecí napětí 24 V DC a uzemnění
- ❏ 8kolíková objímka RJ45 pro připojení Ethernetu
- ❏ Zástrčkový konektor pro vstupní/výstupní sběrnici
- ❏ Integrovaný plochý kabel s vývodem pro vstupní/výstupní sběrnici



Obr. A-38 Připojení internetového modulu (CP 243–1 IT)

Další informace

Více informací o internetovém modulu (CP 243–1 IT) najdete v *Technickém manuálu SIMATIC NET CP 243–1 IT Komunikační procesor pro průmyslový Ethernet a informační technologie.*

Specifikace modulu AS–Interface (CP 243–2)

Tabulka A-61 Objednací číslo modulu AS-Interface (CP 243–2)

Objednací číslo	Rozšiřovací modul	Vstupy EM	Výstupy EM	Odnímatelná svorkovnice
6GK7 243–2AX01–0XA0	Modul AS–Interface (CP 243–2)	8 digitálních a 8 analogových	8 digitálních a 8 analogových	Ano

Tabulka A-62 Obecná specifikace modulu AS-Interface (CP 243–2)

Objednací číslo	Název a popis modulu	Rozměry (mm) (š x v x h)	Hmotnost	Ztráty	Požadavky V DC +5 V DC	Požadavky V DC z AS–Interface
6GK7 243–2AX01–0XA0	Modul AS–Interface (CP 243–2)	71 x 80 x 62	cca 250 g	3,7 W	220 mA	100 mA

Tabulka A-63 Specifikace modulu AS-Interface (CP 243–2)

Obecné	6GK7 243–2AX01–0XA0
Délka cyklu	5 ms při 31 zařízeních slave 10 ms při 62 zařízeních AS–I slave, která používají režim rozšířené adresace
Konfigurace	Pomocí tlačítka na předním panelu nebo použijte příkaz celkové konfigurace (viz popis příkazů AS–I v manuálu *CP 243–2 AS–I Interface Master*)
Podporované profily AS–I master	M1e
Připojení ke kabelu AS–I	Přes svorkovnici S7-200. Povolené proudové zatížení ze svorky 1 až 3 nebo ze svorky 2 až 4 maximálně 3 A.
Adresový rozsah	Jeden digitální modul s 8 digitálními vstupy a 8 digitálními výstupy a jeden analogový modul s 8 analogovými vstupy a 8 analogovými výstupy

Vlastnosti

Spolu s automatem S7-200 mohou být naráz provozovány až dva moduly AS–Interface, což výrazně zvyšuje počet digitálních a analogových vstupů a výstupů, které máte k dispozici (maximálně 124 digitálních vstupů/124 digitálních výstupů na AS–Interface na CP). Čas potřebný pro konfiguraci je redukován díky schopnosti provést konfiguraci stisknutím tlačítka. Stavové LED redukují dobu prostoje v případě chyby tím, že ukazují stav CP a všech připojených zařízení slave, a tím, že monitorují hlavní napětí AS–Interface.

Modul AS–Interface má následující funkce:

- ❏ Podporuje analogové moduly.
- ❏ Podporuje všechny funkce master a umožouje připojení až 62 zařízení AS–Interface slave.
- ❏ Kontrolky na předním panelu zobrazují provozní stav a dostupnost připojených zařízení slave.
- ❏ Kontrolky na předním panelu zobrazují chyby (včetně chyby napětí AS–Interface, chyby konfigurace).
- ❏ Dvě svorky umožňují přímé připojení kabelu AS–Interface.
- ❏ Dvě tlačítka dávají informace o stavu zařízení slave, o přepínači pracovního režimu a mohou zajistit převzetí stávající konfigurace jako SET konfiguraci.


AS-i

Pro konfiguraci modulu AS–Interface (CP 243–2) můžete použít ”Průvodce AS-i” v STEP 7–Micro/WIN. ”Průvodce AS–Interface” vám pomůže použít data ze sítě AS-Interface ve vaší konfiguraci. Chcete–li spustit ”Průvodce AS–i”, zvolte příkaz menu **Nástroje > Průvodce AS–i**.

Činnost

V obrazu procesu S7-200 zabírá modul AS–Interface digitální vstupní byte (stavový byte), digitální výstupní byte (řídicí byte), 8 analogových vstupních a 8 analogových výstupních wordů. Modul AS–Interface využívá prostor dvou logických modulů. Stavový a řídicí byte je možné použít pro nastavení režimu modulu AS–Interface pomocí uživatelského programu. V závislosti na svém režimu AS–Interface ukládá buď I/O data zařízení AS–Interface slave, nebo diagnostické hodnoty, nebo povoluje zprávy zařízení master (například změna adresy slave) v oblasti analogových adres S7-200.

Všechna připojená zařízení AS–Interface slave je možné konfigurovat pouhým stisknutím tlačítka. Další konfigurace CP není potřeba.

Upozornění

Při použití modulu AS–Interface je nutné v CPU zablokovat analogovou filtraci.

Pokud není analogová filtrace v CPU zablokovaná, zničí se digitální data a chyby se nevrací jako bitové hodoty v analogovém word.

Zkontrolujte si, že je analogová filtrace v CPU zablokovaná.

Funkce

CP 243–2 je zařízení AS–Interface master pro třídu M1e, a to znamená, že podporuje všechny specifikované funkce. To umožňuje řídit až 31 digitálních zařízení slave na AS–Interface pomocí dvojnásobného přiřazení adres (A–B). Modul CP 243–2 může být nastaven na dva různé režimy:

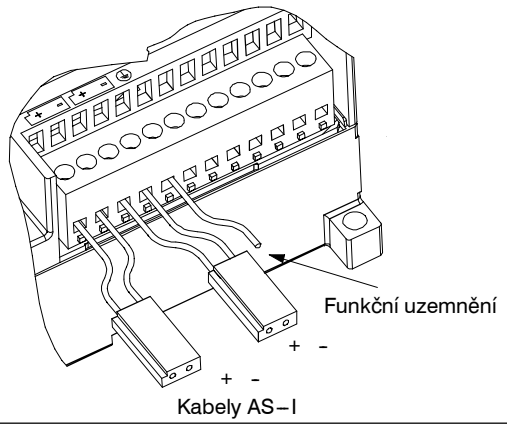
- Standardní režim: přístup na I/O data zařízení AS–Interface slave
- Rozšířený režim: volání zařízení master (například zápis parametru) nebo požadavek na diagnostické hodnoty

Svorky

Modul AS–Interface má následující připojení:

- Dvě připojení ke kabelu modulu AS–Interface (s vnitřním přemostěním)
- Jedno připojení pro funkční uzemnění

Svorky jsou umístěny pod krytem předního panelu, jak je znázorněno na obrázku A-39.



Obr. A-39 Připojení kabelu modulu AS–Interface

Upozornění

Kapacita zatížení kontaktů modulu AS–Interface je maximálně 3 A. Pokud je tato hodnota na kabelu modulu AS–Interface překročena, nesmí být AS–Interface začleněn do smyčky kabelu AS–Interface, ale musí být připojen samostatným kabelem (v tomto případě je využita pouze jedna dvojice svorek modulu AS–Interface). AS–Interface musí být spojen s uzemněním přes zemnicí svorku.

Tip

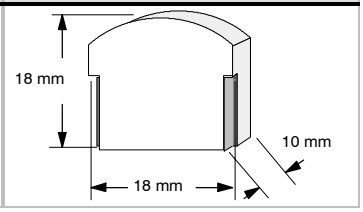
Modul AS–Interface má svorku pro uzemnění. Tato svorka by měla být připojena k vodiči PE s co nejmenším odporem.

Další informace

Více informací o modulu CP 243–2 AS–Interface master najdete v *Manuálu SIMATIC NET CP 243–2 AS-Interface master.*

Volitelné zásuvné moduly

Zásuvný modul	Popis	Objednací číslo
Paměťový modul	Paměťový modul, 32K (uživatelský program)	6ES7 291 – 8GE20 – 0XA0
Paměťový modul	Paměťový modul, 64K (uživatelský program, receptura a záznam dat)	6ES7 291 – 8GF23 – 0XA0
Paměťový modul	Paměťový modul, 256K (uživatelský program, receptura a záznam dat)	6ES7 291 – 8GH23 – 0XA0
Hodiny reálného času s baterií	Přesnost hodinového modulu: 2 minuty/měsíc při 25 ℃, 7 minut/měsíc při 0 ℃ až 55 ℃	6ES7 297 – 1AA23 – 0XA0
Bateriový modul	Bateriový modul (doba uchování dat): běžně 200 dnů	6ES7 291 – 8BA20 – 0XA0

Obecné vlastnosti		Rozměry
Baterie	3 V, 30 mAhodin, Renata CR 1025	
Velikost	9,9 mm x 2,5 mm	
Typ	Lithiová < 0,6 g	

Paměťový modul

Pro používání paměťových modulů jsou omezení mezi různými modely CPU. Paměťové moduly programované v CPU určitého čísla modelu mohou být čteny pouze v CPU se stejným nebo vyšším číslem modelu, jak je uvedeno v tabulce A-64:

Tabulka A-64 Omezení čtení u modelů paměťových modulů

Paměťový modul programovaný v ...	může být čten ...
CPU 221	CPU 221, CPU 222, CPU 224, CPU 224XP a CPU 226
CPU 222	CPU 222, CPU 224, CPU 224XP a CPU 226
CPU 224	CPU 224, CPU 224XP a CPU 226
CPU 224XP	CPU 224XP a CPU 226
CPU 226	CPU 226

Paměťové moduly 64K a 256K jsou určeny pouze pro práci s novými CPU s objednacím číslem uvedeným zde: 6ES7 21x – xx23 – 0XB0. Každé ”x” znamená, že toto číslo může být jakékoliv.

Doporučuje se nepoužívat paměťový modul 32K (6ES7 291 – 8GE20 – 0XA0) ve verzi CPU 23, protože paměťový modul 32K nemůže podporovat funkčnost nového CPU. Jestliže použijete verzi CPU 23 k uložení programu na paměťový modul 32K, programovaný modul je záměrně učiněn kompatibilním se staršími verzemi CPU. Žádná pokročilá funkce nových CPU nemůže být uložena v paměťovém modulu 32K.

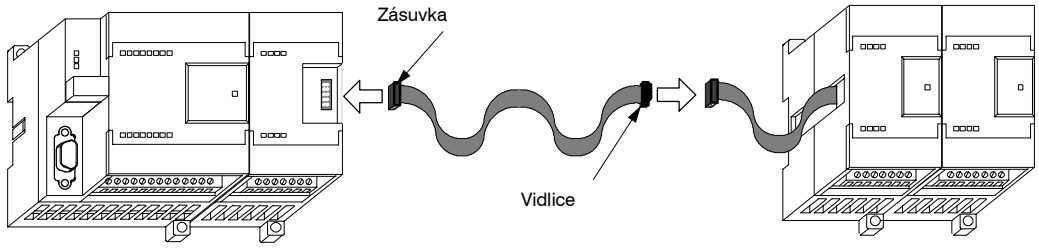
Možná máte uživatelské programy uložené v paměťových modulech 32K, které byly původně naprogramované staršími CPU (verze 20, 21 nebo 22). Tyto moduly mohou být čteny novými CPU podle omezení čísel modelů v tabulce A-64.

Modul hodin reálného času

Modul hodin reálného času (6ES7 297 – 1AA23 – 0XA0) je určen pro práci pouze s CPU 23. Starší verze modulů hodin reálného času (6ES7 297 – 1AA20 – 0XA0) není fyzicky ani elektricky kompatibilní s CPU 23.

Kabel pro rozšiřovací moduly

Obecné vlastnosti (6ES7 290–6AA20–0XA0)	
Délka kabelu	0,8 m
Hmotnost	25 g
Typ konektoru	10pinový plochý



Obr. A-40 Typická instalace kabelu pro rozšiřovací moduly

Tip
V řetězci CPU/rozšiřovací moduly je povolen pouze jeden takový kabel.

RS-232/PPI Multi-Master nebo USB/PPI Multi-Master kabel

Tabulka A-65 Specifikace RS-232/PPI Multi-Master a USB/PPI Multi-Master kabelů

Popis Objednací číslo	RS-232/PPI Multi-Master kabel pro S7-200 6ES7 901–3CB30–0XA0	USB/PPI Multi-Master kabel pro S7-200 6ES7–901–3DB30–0XA0
Obecné vlastnosti		
Napájecí napětí	14,4 až 28,8 V DC	14,4 až 28,8 V DC
Napájecí proud při jmenovitém napájecím napětí 24 V	max. 60 mA efektivní hodnota	max. 50 mA efektivní hodnota
Zpoždění od přijetí hrany stop bitu RS-232 po vyslání hrany stop bitu RS-485	–	–
Napěťové oddělení	RS-485 od RS-232: 500 VDC	RS-485 od USB: 500 VDC
Elektrické vlastnosti RS-485		
Rozsah souhlasného napětí	–7 V až +12 V, 1 sekunda, trvale 3 V efektivní hodnota	–7 V až +12 V, 1 sekunda, trvale 3 V efektivní hodnota
Vstupní impedance přijímače	5,4 kΩ min., včetně ukončení	5,4 kΩ min., včetně ukončení
Zakončení/symetrizace	10 kΩ k +5 V na B, PROFIBUS kolík č. 3 10 kΩ k zemi (GND) na A, PROFIBUS kolík č. 8	10 kΩ k +5 V na B, PROFIBUS kolík č. 3 10 kΩ k zemi (GND) na A, PROFIBUS kolík č. 8
Práh/citlivost přijímače	+/–0,2 V, 60 mV typická hystereze	+/–0,2 V, 60 mV typická hystereze
Napětí diferenciálního výstupu vysílače	2 V min. při R_L=100 Ω, 1,5 V min. při R_L=54 Ω	2 V min. při R_L=100 Ω, 1,5 V min. při R_L=54 Ω
Elektrické vlastnosti RS-232		
Vstupní impedance přijímače	3 kΩ min.	–
Práh/citlivost přijímače	0,8 V min. spodní úroveň, 2,4 V max. vysoká úroveň 0,5 V typická hystereze	–
Výstupní napětí vysílače	+/– 5 V min. při R_L = 3 kΩ	–
Elektrické vlastnosti strany USB		
Plná rychlost (12 MB/s), součást rozhraní člověk – systém (HID)		
Napájecí proud při 5 V	–	max. 50 mA
Proud při výpadku napájení	–	400 uA max.

Vlastnosti

S7-200 RS-232/PPI Multi-Master kabel je továrně nastaven pro optimální výkon s programovacím balíkem STEP 7–Micro/WIN 3.2 Service Pack 4 (nebo novějším). Nastavení výrobce pro tento kabel se liší od PC/PPI kabelů. Konfigurace kabelu pro vaši aplikaci – viz obrázek 1.

S7-200 RS-232/PPI Multi-Master kabel můžete nakonfigurovat, aby pracoval stejně jako PC/PPI kabel a aby byl kompatibilní s kteroukoliv verzí programovacího balíku STEP 7–Micro/WIN, nastavením přepínače 5 na režim PPI/Freeport a následujícím zvolením požadované přenosové rychlosti.

USB kabel vyžaduje pro svůj provoz programovací balík STEP 7–Micro/WIN 3.2 Service Pack 4 (nebo novější).



Tip

Informace o PC/PPI kabelu najdete ve třetím vydání *Manuálu programovatelného automatu S7–200*, objednací číslo (6ES7 298–8FA22–8BH0).

RS-232/PPI Multi-Master kabel pro S7-200

Tabulka A-66 S7-200 RS-232/PPI Multi–Master kabel – Vývody rozhraní lokálního režimu RS-485 až RS-232

Vývody rozhraní RS-485		Vývody lokálního rozhraní RS-232	
Číslo vývodu	**Popis signálu**	**Číslo vývodu**	**Popis signálu**
1	Nepřipojeno	1	Detekce nosné (DCD) (nepoužito)
2	24V zpětný vodič (logická zem RS-485)	2	Přijímaná data (RD) (výstup PC/PPI kabelu)
3	Signál B (RxD/TxD+)	3	Vysílaná data (TD) (vstup PC/PPI kabelu)
4	RTS (úroveň TTL)	4	Datový terminál připraven (DTR)[1]
5	Nepřipojeno	5	Uzemnění (logická zem RS-232)
6	Nepřipojeno	6	Pohotovost (DSR)[1]
7	24V napájení	7	Výzva k vysílání (RTS) (nepoužito)
8	Signál A (RxD/TxD–)	8	Připraven k vysílání (CTS) (nepoužito)
9	Volba protokolu	9	Indikátor vyzvánění (RI) (nepoužito)

[1] Vývody 4 a 6 jsou zapojeny vnitřně.

TabulkaA-67 S7-200 RS-232/PPI Multi–Master kabel – Vývody rozhraní dálkového režimu RS-485 až RS-232

Vývody rozhraní RS-485		Vývody rozhraní RS-232 v dálkovém režimu[1]	
Číslo vývodu	**Popis signálu**	**Číslo vývodu**	**Popis signálu**
1	Nepřipojeno	1	Detekce nosné (DCD) (nepoužito)
2	24V zpětný vodič (logická zem RS-485)	2	Přijímaná data (RD) (vstup PC/PPI kabelu)
3	Signál B (RxD/TxD+)	3	Vysílaná data (TD) (výstup PC/PPI kabelu)
4	RTS (úroveň TTL)	4	Datový terminál připraven (DTR)[2]
5	Nepřipojeno	5	Uzemnění (logická zem RS-232)
6	Nepřipojeno	6	Pohotovost (DSR)[2]
7	24V napájení	7	Výzva k vysílání (RTS) (výstup z PC/PPI kabelu)
8	Signál A (RxD/TxD–)	8	Připraven k vysílání (CTS) (nepoužito)
9	Volba protokolu	9	Indikátor vyzvánění (RI) (nepoužito)

[1] Pro modemy je potřeba převod ze zásuvky na vidlici a z 9 vývodů na 25.
[2] Vývody 4 a 6 jsou zapojeny vnitřně.

S7-200 RS-232/PPI Multi-Master kabel se STEP 7–Micro/WIN použijte místo PC/PPI kabelu nebo pro práci v režimu Freeport

Pro přímé připojení k osobnímu počítači:

- ❑ Nastavte režim PPI/Freeport (přepínač 5=0)
- ❑ Nastavte přenosovou rychlost (přepínače 1, 2 a 3)
- ❑ Nastavte lokální režim (přepínač 6=0). Nastavení lokálního režimu je stejné jako nastavení PC/PPI kabelu na DCE.
- ❑ Nastavte 11bitový režim (přepínač 7=0)

Pro připojení k modemu:

- ❑ Nastavte režim PPI/Freeport (přepínač 5=0)
- ❑ Nastavte přenosovou rychlost (přepínače 1, 2 a 3)
- ❑ Nastavte dálkový režim (přepínač 6=1). Nastavení dálkového režimu je stejné jako nastavení PC/PPI kabelu na DTE.
- ❑ Nastavte 10bitový nebo 11bitový režim (přepínač 7) tak, aby odpovídal nastavení počtu bitů na znak vašeho modemu.

S7-200 RS-232/PPI Multi-Master kabel používejte se STEP 7–Micro/WIN 3.2 Service Pack 4 (nebo novějším)

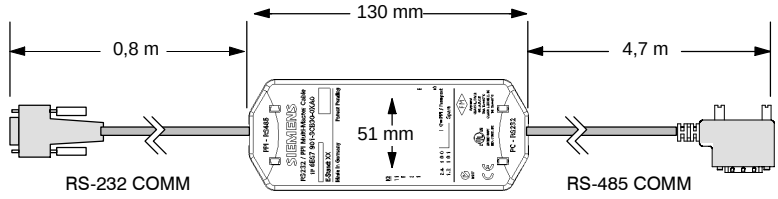
Pro přímé připojení k osobnímu počítači:

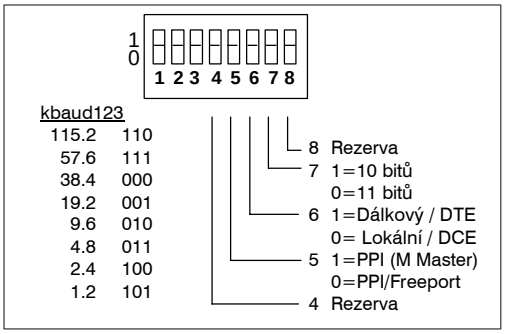
- ❑ Nastavte režim PPI (přepínač 5=1)
- ❑ Nastavte lokální režim (přepínač 6=0)

Pro připojení k modemu:

- ❑ Nastavte režim PPI (přepínač 5=1)
- ❑ Nastavte dálkový režim (přepínač 6=1)

Obrázek A-41 ukazuje rozměry, štítek a kontrolky (LED) S7-200 RS-232/PPI Multi-Master kabelu.





LED	Barva	Popis
Tx	Zelená	Indikátor vysílání RS-232
Rx	Zelená	Indikátor příjmu RS-232
PPI	Zelená	Indikátor vysílání RS–485

Obr. A-41 Rozměry, štítek a kontrolky (LED) S7-200 RS-232/PPI Multi-Master kabelu

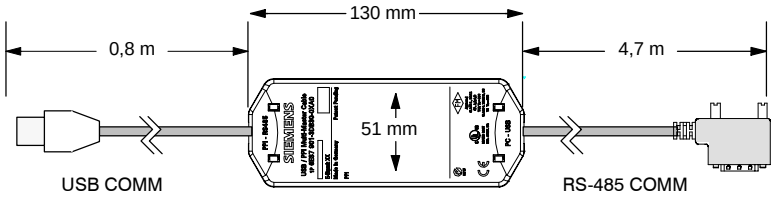
USB/PPI Multi-Master kabel pro S7-200

Abyste mohli používat USB kabel, musíte mít instalován STEP 7 – Micro/WIN 3.2 Service Pack 4 (nebo novější verzi). Doporučuje se používat USB kabel pouze se S7-200 CPU22x nebo novějšími. USB kabel nepodporuje komunikaci Freeport nebo download TP Designeru do TP070.

Tabulka A-68 S7-200 USB/PPI Multi-Master kabel – Vývody rozhraní RS-485 až USB řada "A"

Vývody rozhraní RS-485		Vývody rozhraní USB	
Číslo vývodu	**Popis signálu**	**Číslo vývodu**	**Popis signálu**
1	Nepřipojeno	1	USB – DataP
2	24V zpětný vodič (logická zem RS-485)	2	USB – DataM
3	Signál B (RxD/TxD+)	3	USB 5 V
4	RTS (úroveň TTL)	4	USB logická zem
5	Nepřipojeno		
6	Nepřipojeno		
7	24V napájení		
8	Signál A (RxD/TxD–)		
9	Výběr protokolu (log. 0 = 10 bitů)		

Obrátek A-42 ukazuje rozměry a kontrolky (LED) S7-200 USB/PPI Multi-Master kabelu.

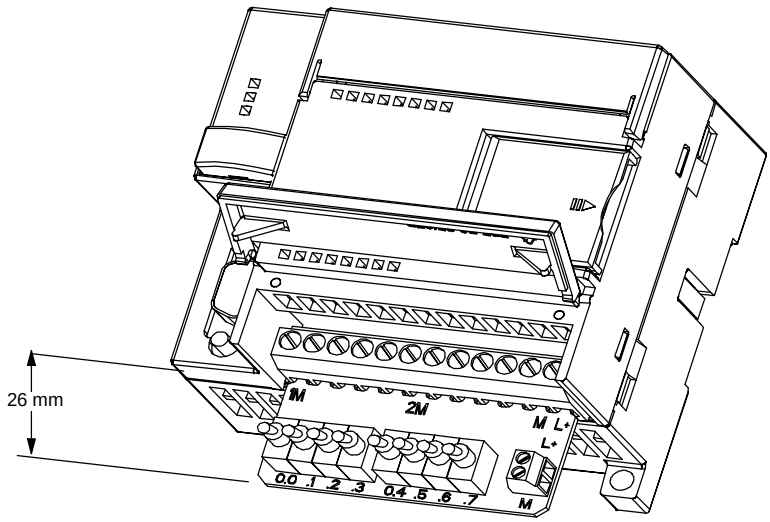


LED	Barva	Popis
Tx	Zelená	Indikátor vysílání USB
Rx	Zelená	Indikátor příjmu USB
PPI	Zelená	Indikátor vysílání RS-485

Obr. A-42 Rozměry a kontrolky (LED) S7-200 USB/PPI Multi-Master kabelu

Simulátory vstupů

Objednací číslo	8místný simulátor vstupů 6ES7 274–1XF00–0XA0	14místný simulátor vstupů 6ES7 274–1XH00–0XA0	24místný simulátor vstupů 6ES7 274–1XK00–0XA0
Velikost (d x š x h)	61 x 33,5 x 22 mm	91,5 x 35,5 x 22 mm	148,3 x 35,5 x 22 mm
Hmotnost	0,02 kg	0,03 kg	0,04 kg
Vstupy	8	14	24



Obr. A-43 Instalace simulátoru vstupů

Varování

Tyto simulátory vstupů nejsou schváleny pro použití na nebezpečných místech třídy I DIV 2 nebo třídy I, zóna 2. Přepínače představují eventuální nebezpečí jiskření.

Simulátory vstupů nepoužívejte na nebezpečných místech třídy I DIV 2 nebo třídy I, zóna 2.

Výpočet energetické bilance

B

CPU S7-200 má vnitřní napájecí zdroj, který napájí CPU, rozšiřovací moduly a další zařízení uživatele s napětím 24 V DC. Použijte následující informace jako průvodce pro stanovení, jaký proud napájení může CPU S7-200 dodávat pro vaši konfiguraci.

Požadavky na napájení

Každá CPU S7-200 dodává napětí 5 V DC a 24 V DC:

- Každá CPU má napájení pro snímače 24 V DC, které může dodávat 24 V DC pro lokální vstupy nebo pro cívky relé v rozšiřovacích modulech. Pokud požadavky na napájení 24 V DC překročí energetickou bilanci CPU, můžete přidat vnější zdroj 24 V DC, který bude napájet rozšiřovací moduly. Napájení 24 V DC musíte ke vstupům nebo cívkám relé ručně připojit.
- CPU také poskytuje 5 V DC napájení pro rozšiřovací moduly, pokud jsou připojeny. Pokud požadavky na 5 V DC napájení rozšiřovacích modulů překročí energetickou bilanci CPU, musíte rozšiřovací moduly odpojit a ponechat jich jen tolik, aby je tato bilance pokryla.

Specifikace v příloze A poskytuje informace o energetické bilanci jednotlivých CPU a požadavcích rozšiřovacích modulů na napájení.



Tip

Pokud je překročena energetická bilance CPU, možná nebudete moci připojit maximální počet modulů, který jinak CPU umožňuje.



Varování

Připojení externího napájecího zdroje 24 V DC paralelně se stejnosměrným zdrojem pro snímače S7-200 může vést ke konfliktu mezi těmito dvěma zdroji, protože se každý snaží nastavit svou vlastní úroveň výstupního napětí.

Následkem toho může být zkrácena životnost zařízení nebo může dojít k okamžité poruše jednoho nebo obou napájecích zdrojů a k následnému nepředvídatelnému chování řídicího systému. Nepředvídatelné chování by mohlo mít za následek smrt nebo vážné poranění osob a poškození zařízení.

Stejnosměrný zdroj pro snímače S7-200 a externí zdroj by měly dodávat napětí do různých bodů. Přitom mezi oběma zdroji napětí může existovat jen jedno propojení.

Vzor výpočtu požadavků na napájení

Tabulka B-1 uvádí vzor výpočtu požadavků na napájení pro automat S7-200, který obsahuje:

- CPU S7-200 224 AC/DC/relé
- 3 EM 223 8 DC vstupů/8 reléových výstupů
- 1 EM 221 8 DC vstupů

Tato instalace má celkem 46 vstupů a 34 výstupů.



Tip
CPU již přidělil napájení požadované pro pohon interních cívek relé. Nemusíte zahrnovat požadavky na napájení interních cívek relé do výpočtu energetické bilance.

V tomto příkladu CPU S7-200 dává pro rozšiřovací moduly dostatečný proud ze zdroje 5 V DC, ale nedává dostatečný proud ze zdroje pro snímače 24 V DC pro všechny vstupy a cívky relé v rozšiřovacích modulech. Vstupy a výstupy vyžadují 400 mA a CPU S7-200 CPU dává pouze 280 mA. Do této instalace je třeba zapojit další zdroj nejméně 120 mA při 24 V DC, aby bylo možné provozovat všechny 24 V DC vstupy a výstupy.

Tabulka B-1 Výpočty energetické bilance pro vzorovou konfiguraci

Energetická bilance CPU	**5 VDC**	**24 VDC**
CPU 224 AC/DC/Relé	660 mA	280 mA

minus

Požadavky systému	**5 VDC**	**24 VDC**
CPU 224, 14 vstupů		14 * 4 mA = 56 mA
3 EM 223, potřebné napájení 5 V	3 * 80 mA = 240 mA	
1 EM 221, potřebné napájení 5 V	1 * 30 mA = 30 mA	
3 EM 223, každý s 8 vstupy		3 * 8 * 4 mA = 96 mA
3 EM 223, každý s 8 cívkami relé		3 * 8 * 9 mA = 216 mA
1 EM 221, každý s 8 vstupy		8 * 4 mA = 32 mA
Požadavky celkem	270 mA	400 mA

rovná se

Proudová bilance	**5 VDC**	**24 VDC**
Proudová bilance celkem	390 mA	[120 mA]

Výpočet vašich požadavků napájení

Pro stanovení, jaké napětí (nebo proud) může CPU S7-200 pro vaši konfiguraci dodávat, použijte následující tabulku. Energetické bilance vašeho typu CPU a požadavky napájení rozšiřovacích modulů najdete v příloze A.

Energetická bilance	**5 VDC**	**24 VDC**

minus

Požadavky systému	**5 VDC**	**24 VDC**
Požadavky celkem		

rovná se

Proudová bilance	**5 VDC**	**24 VDC**
Proudová bilance celkem		

Chybové kódy

C

Informace o chybových kódech pomáhají vyřešit problémy s CPU S7-200.

Přehled kapitoly

Kódy a hlášení kritických chyb 462
Problémy programování při běhu programu 463
Porušení pravidel kompilace 464

Kódy a hlášení kritických chyb

Kritické chyby způsobí, že automat zastaví provádění uživatelského programu. Kritická chyba může v závislosti na své závažnosti způsobit, že S7-200 nebude schopen provádět některé nebo všechny své funkce. Cílem ošetření kritických chyb je přivést automat S7-200 do bezpečného stavu, ve kterém může odpovídat na dotazy o stávajících chybových podmínkách.

Při zjištění kritické chyby provede S7–200 následující úlohy:

- ❏ Přejde do režimu STOP
- ❏ Rozsvítí LED SF/DIAG (červená) a STOP
- ❏ Vypne výstupy

V tomto stavu S7-200 zůstane, dokud není kritická chyba opravena. Chcete–li si prohlédnout chybové kódy, zvolte příkaz **PLC > Informace** z hlavního menu. Tabulka C-1 uvádí seznam a popis kódů kritických chyb, které lze přečíst z automatu S7-200.

Tabulka C-1 Kódy a hlášení kritických chyb, které uvádí automat S7–200

Chybové kódy	Popis
0000	Bez závažné chyby
0001	Chyba kontrolního součtu v programu
0002	Chyba kontrolního součtu při kompilaci LAD programu
0003	Chyba časové prodlevy obvodu sledování funkce programového cyklu (WATCHDOG)
0004	Permanentní paměť selhala
0005	Permanentní paměť: chyba kontrolního součtu v uživatelském programu
0006	Permanentní paměť: chyba kontrolního součtu v parametrech konfigurace (SDB0)
0007	Permanentní paměť: chyba kontrolního součtu ve vnucených datech
0008	Permanentní paměť: chyba kontrolního součtu v tabulce implicitních výstupů
0009	Permanentní paměť: chyba kontrolního součtu v uživatelských datech, DB1
000A	Paměťová karta: selhala
000B	Paměťová karta: chyba kontrolního součtu v uživatelském programu.
000C	Paměťová karta: chyba kontrolního součtu v parametrech konfigurace (SDB0)
000D	Paměťová karta: chyba kontrolního součtu ve vnucených datech (force)
000E	Paměťová karta: chyba kontrolního součtu v tabulce implicitních výstupů
000F	Paměťová karta: chyba kontrolního součtu v uživatelských datech, DB1
0010	Interní softwarová chyba
0011[1]	Chyba nepřímé adresace komparačního kontaktu
0012[1]	Nepřípustná hodnota s plovoucí desetinnou čárkou komparačního kontaktu
0013	Program není spustitelný v tomto S7-200
0014[1]	Chyba rozsahu komparačního kontaktu

[1] Chyby komparačního kontaktu jsou jediné, které generují kritickou i nekritickou chybu. Důvodem pro generování nekritické chyby je uložení adresy chyby.

Problémy programování při běhu programu

Během provádění uživatelského programu mohou vzniknout nekritické chyby (jako jsou chyby adresace). V takovém případě generuje automat S7-200 chybový kód v době vykonávání programu. Tabulka C-2 uvádí seznam a popis kódů nekritických chyb.

Tabulka C-2 Problémy programování v době vykonávání programu

Chybové kódy	Popis
0000	Bez kritické chyby; bez chyby
0001	HSC blok aktivován před vykonáním HDEF
0002	Vstup je již využit pro HSC; nelze tedy použít jako přerušení
0003	Vstup je již využit jako vstup přerušení nebo pro jiný HSC; nelze tedy použít pro HSC
0004	Pokus o provedení instrukce, která není povolena v podprogramu přerušení
0005	Pokus o vykonávání druhého HSC/PLS se stejným číslem jako ještě neukončený předcházející (HSC/PLS v podprogramu přerušení je v konfliktu s HSC/PLS v hlavním programu)
0006	Chyba nepřímé adresace
0007	Chyba dat TODW (Zápis reálného času) nebo TODR (Čtení reálného času)
0008	Překročen maximální počet vnořených podprogramů
0009	Současné provádění instrukcí XMT/RCV na portu 0
000A	Pokus o předefinování HSC vykonáváním další instrukce HDEF pro stejný HSC
000B	Současné provádění instrukcí XMT/RCV na portu 1
000C	Hodinový modul nepřístupný pro TODR, TODW nebo komunikaci
000D	Pokus o předefinování pulzního výstupu v době, kdy je aktivní
000E	Počet segmentů profilu PTO byl nastaven na 0
000F	Nepřípustná číselná hodnota v instrukci komparačního kontaktu
0010	Příkaz není povolen v aktuálním provozním režimu PTO
0011	Neplatný kód příkazu PTO
0012	Neplatná tabulka profilu PTO
0013	Neplatná tabulka smyčky PID
0091	Mimo rozsah (informace o adresách): zkontrolujte rozsah operandů
0092	Chyba při čítání nebo chyba v instrukci čítače (informace o čítání): ověřte maximální hodnotu
0094	Chyba rozsahu pro zápis do pevné paměti s udáním adresy
009A	Pokus o přepnutí do režimu Freeport v době uživatelského přerušení
009B	Nepřípustný index (operace s řetězcem, ve kterém je specifikována jako hodnota výchozího místa 0)
009F	Paměťový modul chybí nebo neodpovídá

Porušení pravidel kompilace

Když provádíte download programu, automat S7-200 program nejdříve zkompiluje. Pokud S7-200 zjistí, že program porušuje nějaké pravidlo kompilace (příkladem může být nepřípustná instrukce), přeruší se download a generuje se kód nekritické chyby, porušující pravidla kompilace.
Tabulka C-3 uvádí seznam a popis chybových kódů, které jsou generovány, pokud jsou porušena pravidla kompilace.

Tabulka C-3 Porušení pravidel kompilace

Chybové kódy	Chyby kompilace nekritické
0080	Program je příliš dlouhý pro kompilaci; zmenšete délku programu
0081	Podtečení zásobníku, rozdělte network na více networků.
0082	Neznámá instrukce; zkontrolujte zápis instrukce.
0083	Chybí MEND nebo instrukce není povolena v hlavním programu: přidejte instrukci MEND nebo odstraňte nesprávnou instrukci.
0084	Rezervováno
0085	Chybí FOR; přidejte instrukci FOR nebo smažte instrukci NEXT
0086	Chybí NEXT, přidejte instrukci NEXT nebo smažte instrukci FOR
0087	Chybí návěští (LBL, INT, SBR); přidejte požadované návěští.
0088	Chybí RET nebo instrukce není povolena v podprogramu: přidejte RET na konec podprogramu nebo odstraňte nesprávnou instrukci.
0089	Chybí RETI nebo instrukce není povolena v podprogramu přerušení: přidejte RETI na konec podprogramu přerušení nebo odstraňte nesprávnou instrukci.
008A	Rezervováno
008B	Nepřípustné JMP do segmentu SCR nebo z něho
008C	Duplikátní návěští (LBL, INT, SBR); přejmenujte jedno z návěští.
008D	Neznámé návěští (LBL, INT, SBR); zajistěte dodržení max. počtu návěští.
0090	Neznámý parametr, zkontrolujte parametry instrukce.
0091	Chyba rozsahu; zkontrolujte rozsah operandů.
0092	Chyba v parametru instrukce čítače, zkontrolujte maximální hodnotu parametru.
0093	Překročena úroveň zanoření FOR/NEXT
0095	Chybějící instrukce LSCR
0096	Chybějící instrukce SCRE nebo nedovolená instrukce před instrukcí SCRE
0097	Uživatelský program obsahuje nečíslované i číslované instrukce EV/ED
0098	Nepřípustná editace v režimu RUN (pokus o editaci programu s nečíslovanými instrukcemi EV/ED)
0099	Příliš velký počet skrytých programových segmentů (instrukce HIDE)
009B	Nepřípustný index (operace s řetězcem, ve kterém je specifikována jako hodnota výchozího místa 0)
009C	Překročena maximální délka instrukce
009D	Neplatný parametr detekovaný v SBD0
009E	Příliš mnoho řetězců PCALL
009F až 00FF	Rezervováno

Speciální paměťové bity (SM)

D

Speciální paměťové bity poskytují různé stavové informace a řídicí funkce a zároveň slouží jako prostředek komunikace mezi automatem S7-200 a uživatelským programem. Speciální paměťové bity je možné používat jako bity, byty, word nebo double word.

Přehled kapitoly

SMB0: Stavové bity 466
SMB1: Stavové bity 466
SMB2: Znak přijatý v režimu Freeport 467
SMB3: Chyba parity v režimu Freeport 467
SMB4: Přeplnění fronty 467
SMB5: Stav vstupů a výstupů 468
SMB6: Registr ID CPU 468
SMB7: Rezervován 468
SMB8 až SMB21: ID a chybové registry I/O modulu 469
SMW22 až SMW26: Délka programového cyklu 470
SMB28 a SMB29: Analogové potenciometry 470
SMB30 a SMB130: Řídicí registry pro Freeport 470
SMB31 a SMW32: Řízení zápisu do trvalé paměti (EEPROM) 471
SMB34 a SMB35: Intervaly pro časovaná přerušení 471
SMB36 až SMB65: Registr HSC0, HSC1 a HSC2 471
SMB66 až SMB85: Registry PTO/PWM 473
SMB86 až SMB94 a SMB186 až SMB194: Řízení příjmu zpráv 474
SMW98: Chyby na rozšiřovací I/O sběrnici 475
SMB130: Řídicí registr pro Freeport (viz SMB30) 475
SMB131 až SMB165: Registr HSC3, HSC4 a HSC5 475
SMB166 až SMB185: Tabulka definice profilu PTO0, PTO1 476
SMB186 až SMB194: Řízení příjmu zpráv (viz SMB86 až SMB94) 476
SMB200 až SMB549: Stav inteligentních modulů 477

SMB0: Stavové bity

Jak je popsáno v tabulce D-1, SMB0 obsahuje osm stavových bitů, které automat S7-200 aktualizuje na konci každého programového cyklu.

Tabulka D-1 Speciální paměťový byte SMB0 (SM0.0 až SM0.7)

SM bity	Popis (pouze ke čtení)
SM0.0	Tento bit má vždy hodnotu 1.
SM0.1	Tento bit je v log. 1 v prvním programovém cyklu. Jedním z použití je volání inicializačního podprogramu.
SM0.2	Tento bit přejde do stavu log. 1 na jeden programový cyklus, pokud byla ztracena remanentní data. Je možné ho použít buď jako chybový bit paměti, nebo jako mechanismus vyvolání speciální spouštěcí sekvence.
SM0.3	Tento bit přejde do stavu log. 1 na jeden programový cyklus, když systém přešel po studeném restartu do režimu RUN. Je možné ho použít pro inicializaci stroje před spuštěním vlastního provozu.
SM0.4	Tento bit generuje hodinový pulz, který jev log. 1 po 30 sekund a v log. 0 po dobu 30 sekund v pracovním cyklu 1 minuta. To umožňuje vytvořit snadno použitelné zpoždění nebo 1minutový pulz.
SM0.5	Tento bit generuje hodinový pulz, který je v log. 1 po dobu 0,5 sekundy, a pak vypnutý po 0,5 sekundy a pak v log. 0 po dobu 0,5 sekundy v pracovním cyklu 1 sekunda. To umožňuje vytvořit snadno použitelné zpoždění nebo sekundový pulz.
SM0.6	Tento bit slouží jako hodiny programového cyklu; v log. 1 je po dobu jednoho programového cyklu a pak v log. 0 během příštího programového cyklu. Je možné ho použít jako vstup čítače programových cyklů.
SM0.7	Tento bit odráží polohu přepínače pracovního režimu (vypnutý znamená polohu TERM, zapnutý polohu RUN). Pokud tento bit použijete pro povolení režimu Freeport ve chvíli, když je přepínač v poloze RUN, je možné použít normální komunikaci s programovacím zařízením, pokud je přepínač v poloze TERM.

SMB1: Stavové bity

Jak je uvedeno v tabulce D-2, obsahuje SMB1 indikátory různých potenciálních chyb. Tyto bity jsou nastavovány instrukcemi při jejich provádění.

Tabulka D-2 Speciální paměťový byte SMB1 (SM1.0 až SM1.7)

SM bity	Popis (pouze ke čtení)
SM1.0	Tento bit přejde do log. 1 provedením určitých instrukcí při nulovém výsledku operace.
SM1.1	Tento bit přejde do log. 1 provedením určitých instrukcí, pokud je výsledkem zjištěna nepřípustná číselná hodnota.
SM1.2	Tento bit přejde do log. 1, když má některá matematická operace záporný výsledek.
SM1.3	Tento bit přejde do log. 1 při pokusu o dělení nulou.
SM1.4	Tento bit přejde do log. 1, když se instrukce přidání do tabulky (Add to Table) pokusí tabulku přeplnit.
SM1.5	Tento bit přejde do log. 1, když se instrukce LIFO nebo FIFO pokusí číst z prázdné tabulky.
SM1.6	Tento bit přejde do log. 1, když dojde k pokusu o převedení jiné než BCD hodnoty na binární.
SM1.7	Tento bit přejde do log. 1, když nemůže být ASCII hodnota převedena na platnou hexadecimální hodnotu.

SMB2: Znak přijatý v režimu Freeport

SMB2 je zásobník pro znak přijatý z Freeportu. Jak je popsáno v tabulce D-3, každý znak přijatý během režimu Freeport je umístěn na tuto adresu kvůli jednoduchému přístupu z programu kontaktního schématu.



Tip
SMB2 a SMB3 jsou sdíleny mezi portem 0 a portem 1. Když se po přijetí znaku na portu 0 provede podprogram přerušení přiřazený k této události (událost 8), obsahuje SMB2 znak přijatý na portu 0 a SMB3 obsahuje stav parity přijatého znaku. Když se po přijetí znaku na portu 1 provede podprogram přerušení přiřazený k této události (událost 25), obsahuje SMB2 znak přijatý na portu 1 a SMB3 stav parity přijatého znaku.

Tabulka D-3 Speciální paměťový byte SMB2

SM byte	Popis (pouze ke čtení)
SMB2	Tento byte obsahuje každý znak, který je přijat z portu 0 nebo portu 1 během komunikace v režimu Freeport.

SMB3: Chyba parity v režimu Freeport

SMB3 se používá pro režim Freeport a obsahuje bit chyby parity, který je nastaven při detekci chyby parity u přijatého znaku. Jak je ukázáno v tabulce D-4, SM3.0 se zapne, když je zjištěna chyba parity. Tento bit použijte pro detekci chyby zprávy.

Tabulka D-4 Speciální paměťový byte SMB3 (SM3.0 až SM3.7)

SM bity	Popis (pouze ke čtení)
SM3.0	Chyba parity z portu 0 nebo portu 1 (0 = bez chyby; 1 = zjištěna chyba)
SM3.1 až SM3.7	Rezervováno

SMB4: Přeplnění fronty

Jak je popsáno v tabulce D-5, obsahuje SMB4 bity přeplnění fronty přerušení, dále stavový indikátor, který ukazuje, zda jsou přerušení povolena nebo zablokována, a paměťový bit nečinnosti vysílače. Bity přeplnění fronty indikují, že přerušení jsou generována rychleji, než je možné zpracovat, nebo že přerušení byla zablokována instrukcí globálního zablokování přerušení.

Tabulka D-5 Speciální paměťový byte SMB4 (SM4.0 až SM4.7)

SM bity	Popis (pouze ke čtení)
SM4.0[1]	Tento bit přejde do log. 1, když došlo k přeplnění fronty komunikačních přerušení.
SM4.1[1]	Tento bit přejde do log. 1, když došlo k přeplnění fronty vstupních přerušení.
SM4.2[1]	Tento bit přejde do log. 1, když došlo k přeplnění fronty časovaných přerušení.
SM4.3	Tento bit přejde do log. 1, když je zjištěn problém programování při běhu programu.
SM4.4	Tento bit odráží stav globálního povolení přerušení. Zapne se, když jsou přerušení povolena.
SM4.5	Tento bit přejde do log. 0, když je vysílač nečinný (port 1).
SM4.6	Tento bit přejde do log. 1, když je vysílač nečinný (port 1).
SM4.7	Tento bit se zapne při vnucení (force) nějaké hodnoty.

[1] Stavové bity 4.0, 4.1 a 4.2 používejte pouze v podprogramu přerušení. Při vyprázdnění fronty jsou tyto stavové bity vynulovány a řízení se vrací do hlavního programu.

SMB5: Stav vstupů a výstupů

Jak popisuje tabulka D-6, obsahuje SMB5 stavové bity indikující chyby, které byly zjištěny v systému vstupů a výstupů. Tyto bity poskytují přehled zjištěných chyb vstupů a výstupů.

Tabulka D-6 Speciální paměťový byte SMB5 (SM5.0 až SM5.7)

SM bity	Popis (pouze ke čtení)
SM5.0	Tento bit přejde do log. 1, existují – li nějaké chyby vstupů a výstupů.
SM5.1	Tento bit přejde do log. 1, jestliže bylo na I/O sběrnici připojeno příliš mnoho digitálních vstupů/výstupů.
SM5.2	Tento bit přejde do log. 1, jestliže bylo na I/O o příliš mnoho digitálních vstupů/výstupů.
SM5.3	Tento bit přejde do log. 1, jestliže bylo na I/O sběrnici připojeno příliš mnoho inteligentních modulů.
SM5.4 až SM5.7	Rezervováno.

SMB6: Registr ID CPU

Jak popisuje tabulka D-7, SMB6 je identifikační registr CPU S7-200. SM6.4 až SM6.7 identifikují typ CPU S7-200. SM6.0 až SM6.3 jsou rezervovány pro použití v budoucnosti.

Tabulka D-7 Speciální paměťový byte SMB6

SM bity	Popis (pouze ke čtení)
Formát	MSB 7 … LSB 0: x x x x r r r r — ID registr CPU
SM6.0 až SM6.3	Rezervováno
SM6.4 až SM6.7	xxxx = 0000 = CPU 222 0010 = CPU 224 0110 = CPU 221 1001 = CPU 226/CPU 226XM

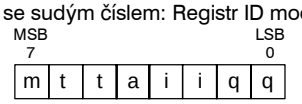
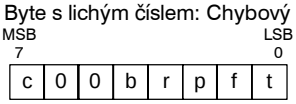
SMB7: Rezervováno

SMB7 je rezervován pro použití v budoucnosti.

SMB8 až SMB21: ID a chybové registry I/O modulů

SMB8 až SMB21 jsou organizovány do dvojic bytů pro rozšiřovací moduly na pozici 0 až 6. Jak popisuje tabulka D-8, v každé dvojici je byte se sudým číslem identifikační registr modulu. Tyto byty identifikují typ modulu, typ vstupů a výstupů a jejich počet. Byte s lichým číslem v každé dvojici je chybový registr modulu. Tyto byty indikují jakékoliv chyby zjištěné ve vstupech/výstupech tohoto modulu.

Tabulka D-8 Speciální paměťové byty SMB8 až SMB21

SM byte	Popis (pouze ke čtení)
Formát	Byte se sudým číslem: Registr ID modulu  m: Přítomnost modulu 0 = Přítomen 1 = Nepřítomen tt: Typ modulu 00 Jiný než inteligentní vstupní/výstupní modul 01 Inteligentní modul 10 Rezervováno 11 Rezervováno a: Typ vstupu/výstupu 0 = Digitální 1 = Analogový ii: Vstupy 00 žádné vstupy 01 2 AI nebo 8 DI 10 4 AI nebo 16 DI 11 8 AI nebo 32 DI qq: Výstupy 00 žádné výstupy 01 2 AQ nebo 8 DQ 10 4 AQ nebo 16 DQ 11 8 AQ nebo 32 DQ Byte s lichým číslem: Chybový registr modulu  c: Chyba konfigurace b: Chyba sběrnice nebo chyba parity r: Chyba překročení rozsahu p: Chyba – nepřítomnost uživatelského napájení f: Chyba – spálená pojistka t: Chyba – uvolněná svorkovnice 0 = bez chyby 1 = chyba
SMB8 SMB9	Registr ID modulu 0 Chybový registr modulu 0
SMB10 SMB11	Registr ID modulu 1 Chybový registr modulu 1
SMB12 SMB13	Registr ID modulu 2 Chybový registr modulu 2
SMB14 SMB15	Registr ID modulu 3 Chybový registr modulu 3
SMB16 SMB17	Registr ID modulu 4 Chybový registr modulu 4
SMB18 SMB19	Registr ID modulu 5 Chybový registr modulu 5
SMB20 SMB21	Registr ID modulu 6 Chybový registr modulu 6

SMW22 až SMW26: Délka programového cyklu

Jak je uvedeno v tabulce D-9, SMW22, SMW24 a SMW26 obsahují informace o programovém cyklu: minimální dobu, maximální dobu a dobu posledního programového cyklu v milisekundách.

Tabulka D-9 Speciální paměťová word SMW22 až SMW26

SM word	Popis (pouze ke čtení)
SMW22	Doba posledního programového cyklu v milisekundách
SMW24	Minimální doba programového cyklu, zaznamenaná od začátku režimu RUN v milisekundách
SMW26	Maximální doba programového cyklu, zaznamenaná od začátku režimu RUN v milisekundách

SMB28 a SMB29: Analogové potenciometry

Jak je uvedeno v tabulce D-10, obsahuje SMB28 číselnou hodnotu, která představuje polohu analogového potenciometru 0. SMB29 obsahuje hodnotu, která představuje polohu analogového potenciometru 1.

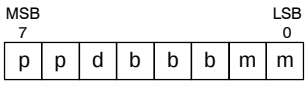
Tabulka D-10 Speciální paměťové byty SMB28 a SMB29

SM byte	Popis (pouze ke čtení)
SMB28	Tento byte uchovává hodnotu nastavenou analogovým potenciometrem 0. Daná hodnota je aktualizována jedenkrát za programový cyklus v STOP/RUN.
SMB29	Tento byte uchovává hodnotu nastavenou analogovým potenciometrem 1. Daná hodnota je aktualizována jedenkrát za programový cyklus v STOP/RUN.

SMB30 a SMB130: Řídicí registry pro Freeport

SMB30 řídí komunikaci v režimu Freeport na portu 0; SMB130 řídí komunikaci na portu 1. Byty SMB30 a SMB130 jsou pro čtení i zápis. Jak je uvedeno v tabulce D-11, tyto byty konfigurují příslušné porty na komunikaci Freeport a zabezpečují přepnutí mezi režimem Freeport a systémovým protokolem.

Tabulka D-11 Speciální paměťový byte SMB30

Port 0	Port 1	Popis
Formát SMB30	Formát SMB130	Řídicí byte režimu Freeport 
SM30.0 a SM30.1	SM130.0 a SM130.1	mm: Volba protokolu 00 = Protokol Point-to-Point Interface (režim PPI/slave) 01 = protokol Freeport 10 = režim PPI/master 11 = Rezervováno (implicitně režim PPI/slave) Poznámka: Když vyberete kód mm = 10 (PPI master), automat S7-200 se stane v síti zařízením typu master a dovolí vykonání instrukcí NETR a NETW. Bity 2 až 7 jsou v režimech PPI ignorovány.
SM30.2 až SM30.4	SM130.2 až SM130.4	bbb: Přenosová rychlost Freeport 000 = 38.400 baud 001 = 19.200 baud 010 = 9.600 baud 011 = 4.800 baud 100 =2.400 baud 101 = 1.200 baud 110 = 115.200 baud 111 = 57.600 baud
SM30.5	SM130.5	d: Datové bity na znak 0 = 8 bitů na znak 1 = 7 bitů na znak
SM30.6 a SM30.7	SM130.6 a SM130.7	pp: Volba parity 00 = žádná parita 01 = sudá parita 10 = lichá parita 11 = lichá parita

SMB31 a SMW32: Řízení zápisu do trvalé paměti (EEPROM)

Prostřednictvím vašeho programu můžete uložit hodnotu uloženou v paměti V do permanentní paměti. Provede se to nahráním adresy místa, jehož obsah chcete uložit, do SMW32. Potom nahrajte do SMB31 příkaz na uložení hodnoty. Jakmile to provedete, neměňte hodnotu v paměti V, dokud automat S7-200 neresetuje SM31.7, to znamená, že operace ukládání byla ukončena.

Na konci každého programového cyklu S7-200 kontroluje, zda byl vydán příkaz k uložení hodnoty do trvalé paměti. Pokud byl příkaz vydán, je daná hodnota uložena do trvalé paměti.

Jak je uvedeno v tabulce D-12, definuje SMB31 velikost dat, která mají být uložena do trvalé paměti, a dává příkaz, který inicializuje operaci uložení. SMW32 uchovává počáteční adresu v paměti V pro data, která mají být uložena do trvalé paměti.

Tabulka D-12 Speciální paměťový byte SMB31 a speciální paměťové word SMW32

SM byte	Popis
Formát	SMB31: Příkaz softwaru MSB 7 … LSB 0 c 0 0 0 0 0 s s SMW32: adresa v paměti V MSB 15 … LSB 0 adresa v paměti V
SM31.0 a SM31.1	ss: Velikost dat 00 = byte 10 = word 01 = byte 11 = double word
SM31.7	c: Uložit do permanentní paměti 0 = Není požadavek na operaci uložení 1 = Uživatelský program požaduje uložení dat Automat S7-200 resetuje tento bit po každé operaci uložení.
SMW32	Adresa v paměti V pro data, která mají být uložena do trvalé paměti, je uložena v SMW32. Tato hodnota je daná jako offset od V0. Při provádění operace uložení je hodnota na této adrese v paměti V uložena na místo odpovídající paměti V v trvalé paměti.

SMB34 a SMB35: Intervaly pro časovaná přerušení

Jak je uvedeno v tabulce D-13, specifikuje SMB34 časový interval pro cyklické přerušení 0 a SMB35 časový interval pro cyklické přerušení 1. Můžete specifikovat časový interval od 1 ms do 255 ms. Hodnota časového intervalu je zachycena S7-200 v okamžiku, kdy je odpovídající událost časovaného přerušení přiřazena k podprogramu přerušení. Chcete–li změnit časový interval, musíte znovu přiřadit událost časovaného přerušení ke stejnému nebo jinému podprogramu přerušení. Událost časovaného přerušení je možné ukončit zrušením přiřazení události k podprogramu přerušení.

Tabulka D-13 Speciální paměťové byty SMB34 a SMB35

SM byte	Popis
SMB34	Tento byte specifikuje časový interval (v přírůstcích po 1 ms, od 1 ms do 255 ms) pro časované přerušení 0.
SMB35	Tento byte specifikuje časový interval (v přírůstcích po 1 ms, od 1 ms do 255 ms) pro časované přerušení 1.

SMB36 až SMB65: Registr HSC0, HSC1 a HSC2

Jak popisuje tabulka D-14, SMB36 až SM65 se používají na monitorování a řízení práce vysokorychlostních čítačů HSC0, HSC1 a HSC2.

Tabulka D-14 Speciální paměťové byty SMB36 až SMD62

SM byte	Popis
SM36.0 až SM36.4	Rezervováno
SM36.5	Stavový bit aktuálního směru počítání HSC0: 1 = počítání nahoru
SM36.6	Stavový bit rovnosti požadované a aktuální hodnoty HSC0: 1 = rovnost
SM36.7	Stavový bit příznaku aktuální hodnota větší než požadovaná u HSC0: 1 = větší
SM37.0	Aktivní úroveň řídicího bitu pro ”Reset”: 0 = ”Reset” je akt. v log. 1, 1 = ”Reset” je aktivní v log. 0
SM37.1	Rezervováno
SM37.2	Výběr rychlosti čítání pro kvadraturní čítače: 0 = 4 x rychlost čítání; 1 = 1 x rychlost čítání
SM37.3	Řídicí bit pro směr HSC0: 1 = počítání nahoru
SM37.4	Aktualizace směru HSC0: 1 = aktualizovat směr
SM37.5	Aktualizace požadované hodnoty HSC0: 1 = zapsat novou pož. hodnotu do nastavení HSC0
SM37.6	Aktualizace aktuální hodnoty HSC0: 1 = zapsat novou akt. hodnotu do stávající HSC0
SM37.7	Bit povolení HSC0: 1 = povolen
SMD38	Nová aktuální hodnota HSC0
SMD42	Nová požadovaná hodnota HSC0
SM46.0 až SM46.4	Rezervováno
SM46.5	Stavový bit aktuálního směru počítání HSC1: 1 = počítání nahoru
SM46.6	Stavový bit rovnosti požadované a aktuální hodnoty HSC1: 1 = rovnost
SM46.7	Stavový bit příznaku aktuální hodnota větší než požadovaná u HSC1: 1 = větší
SM47.0	Aktivní úroveň řídicího bitu pro ”Reset” HSC1: 0 = aktivní v úrovni log. 1, 1 = aktivní v úrovni log. 0
SM47.1	Aktivní úroveň řídicího bitu pro start HSC1: 0 = aktivní v úrovni log. 1, 1 = aktivní v úrovni log. 0
SM47.2	Volba rychlosti kvadraturního čítače HSC1: 0 = 4 x rychlost, 1 = 1 x rychlost
SM47.3	Řídicí bit pro směr HSC1: 1 = počítání nahoru
SM47.4	Aktualizace směru HSC1: 1 = aktualizovat směr
SM47.5	Aktualizace požadované hodnoty HSC1: 1 = zapsat novou pož. hodnotu do nastavení HSC1
SM47.6	Aktualizace aktuální hodnoty HSC1: 1 = zapsat novou současnou hod. do stávající hodnoty HSC1
SM47.7	Bit povolení HSC1: 1 = povolen
SMD48	Nová aktuální hodnota HSC1
SMD52	Nová požadovaná hodnota HSC1
SM56.0 až SM56.4	Rezervováno
SM56.5	Stavový bit aktuálního směru počítání HSC2: 1 = počítání nahoru
SM56.6	Stavový bit rovnosti požadované a aktuální hodnoty HSC2: 1 = rovnost
SM56.7	Stavový bit příznaku aktuální hodnota větší než požadovaná u HSC2: 1 = větší
SM57.0	Aktivní úroveň řídicího bitu pro ”Reset” HSC2: 0 = aktivní v úrovni log. 1, 1 = aktivní v úrovni log. 0
SM57.1	Aktivní úroveň řídicího bitu pro start HSC2: 0 = aktivní v úrovni log. 1, 1 = aktivní v úrovni log. 0
SM57.2	Volba rychlosti kvadraturního čítače HSC2: 0 = 4 x rychlost, 1 = 1 x rychlost
SM57.3	Řídicí bit pro směr HSC2: 1 = počítání nahoru
SM57.4	Aktualizace směru HSC2: 1 = aktualizovat směr
SM57.5	Aktualizace požadované hodnoty HSC2: 1 = zapsat novou pož. hodnotu do nastavení HSC2
SM57.6	Aktualizace aktuální hodnoty HSC2: 1 = zapsat novou současnou hod. do stávající hodnoty HSC2
SM57.7	Bit povolení HSC2: 1 = povolen
SMD58	Nová aktuální hodnota HSC2
SMD62	Nová požadovaná hodnota HSC2

SMB66 až SMB85: Registry PTO/PWM

Jak je uvedeno v tabulce D-15, SMB66 až SMB85 se používají pro monitorování a řízení funkce výstupu série pulzů a šířkové modulace. Úplný popis těchto bitů najdete v informacích o instrukcích pro vysokorychlostní pulzní výstup v kapitole 6.

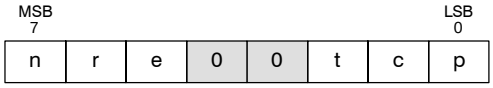
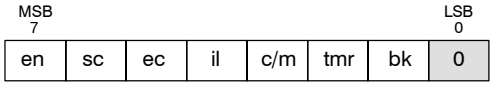
Tabulka D-15 Speciální paměťové byty SMB66 až SMB85

SM byte	Popis
SM66.0 až SM66.3	Rezervováno
SM66.4	PTO0 profil přerušen: 0 = bez chyby, 1 = přerušen pro chybu ve výpočtu změny
SM66.5	PTO0 profil přerušen: 0 = nebyl přerušen uživatelským příkazem, 1 = přerušen uživatelským příkazem
SM66.6	Přeplnění PTO0 (smazáno systémem při použití vnějších profilů, jinak musí být smazáno uživatelem): 0 = nepřeplněno, 1 = přeplněno
SM66.7	Bit nečinnosti PTO0: 0 = PTO pracuje, 1 = PTO nečinný
SM67.0	Aktualizace hodnoty doby cyklu PTO0/PWM0: 1 = zápis nové doby cyklu
SM67.1	Aktualizace hodnoty šířky pulzu PWM0: 1 = zápis nové šířky pulzu
SM67.2	Aktualizace hodnoty počtu pulzů PTO0: 1 = zápis nového počtu pulzů
SM67.3	Časová základna PTO0/PWM0: 0 = 1 σs/pulz, 1 = 1 ms/pulz
SM67.4	Synchronní aktualizace PWM0: 0 = asynchronní aktualizace, 1 = synchronní aktualizace
SM67.5	Práce PTO0: 0 = práce s jedním segmentem (doba cyklu a počet pulzů uloženy v paměti SM), 1 = práce s více segmenty (tabulka profilu uložena v paměti V)
SM67.6	Volba režimu PTO0/PWM0: 0 = PTO, 1 = PWM
SM67.7	Povolovací bit PTO0/PWM0: 1 = povolen
SMW68	Hodnota doby cyklu PTO0/PWM0 (2 až 65.535 jednotek časové základny);
SMW70	Hodnota šířky pulzu PWM0 (0 až 65.535 jednotek časové základny);
SMD72	Hodnota počtu pulzů PTO0 (1 až $2^{32}-1$);
SM76.0 až SM76.3	Rezervováno
SM76.4	Profil PTO1 přerušen: 0 = bez chyby, 1 = přerušen pro chybu ve výpočtu delta
SM76.5	Profil PTO1 přerušen: 0 = nebyl přerušen uživatelským příkazem, 1 = přerušen uživatelským příkazem
SM76.6	Přeplnění řetězce PTO1 (smazáno systémem při použití vnějších profilů, jinak musí být smazáno uživatelem): 0 = nepřeplněno, 1 = přeplněno
SM76.7	Bit nečinnosti PTO1: 0 = PTO pracuje, 1 = PTO nečinný
SM77.0	Aktualizace hodnoty doby cyklu PTO1/PWM1: 1 = zápis nové doby cyklu
SM77.1	Aktualizace hodnoty šířky pulzu PWM1: 1 = zápis nové šířky pulzu
SM77.2	Aktualizace hodnoty počtu pulzů PTO1: 1 = zápis nového počtu pulzů
SM77.3	Časová základna PTO1/PWM1: 0 = 1 σs/pulz, 1 = 1 ms/pulz
SM77.4	Synchronní aktualizace PWM1: 0 = asynchronní aktualizace, 1 = synchronní aktualizace
SM77.5	Práce PTO1: 0 = práce s jedním segmentem (doba cyklu a počet pulzů uloženy v paměti SM), 1 = práce s více segmenty (tabulka profilu uložena v paměti V)
SM77.6	Volba režimu PTO1/PWM1: 0 = PTO, 1 = PWM
SM77.7	Povolovací bit PTO1/PWM1: 1 = povolen
SMW78	Hodnota doby cyklu PTO1/PWM1 (2 až 65.535 jednotek časové základny);
SMW80	Hodnota šířky pulzu PWM1 (0 až 65.535 jednotek časové základny);
SMD82	Hodnota počtu pulzů PTO1 (1 až $2^{32}-1$);

SMB86 až SMB94 a SMB186 až SMB194: Řízení příjmu zpráv

Jak je uvedeno v tabulce D-16, SMB86 až SMB94 a SMB186 až SMB194 se používají pro řízení a čtení stavu instrukce pro příjem zpráv.

Tabulka D-16 Speciální paměťové byty SMB86 až SMB94 a SMB186 až SMB194

Port 0	Port 1	Popis
SMB86	SMB186	Stavový byte pro příjem zprávy  1 = Příjem zprávy ukončen uživatelským příkazem pro ukončení 1 = Příjem zprávy ukončen: chyba vstupních parametrů nebo chybí startovací či ukončovací podmínka 1 = Přijat ukončovací znak 1 = Příjem zprávy ukončen: vypršel čas 1 = Příjem zprávy ukončen: dosaženo maximálního počtu přenášených znaků 1 = Příjem zprávy ukončen pro chybu parity
SMB87	SMB187	Řídicí byte pro příjem zprávy  en: 0 = Funkce příjmu zpráv je zablokovaná. 1 = Funkce příjmu zpráv je povolená. Povolovací bit příjmu zpráv je kontrolován při každém provedení instrukce RCV. provedení instrukce RCV. sc: 0 = Ignorovat SMB88 nebo SMB188. 1 = Použít hodnotu SMB88 nebo SMB188 pro rozpoznání začátku zprávy. ec: 0 = Ignorovat SMB89 nebo SMB189. 1 = Použít hodnotu SMB89 nebo SMB189 pro rozpoznání konce zprávy. il: 0 = Ignorovat SMW90 nebo SMW190. 1 = Použít hodnotu SMW90 nebo SMW190 pro rozpoznání prázdného úseku. c/m: 0 = Časovač je použit na časování mezi znaky. 1 = Časovač je použit na časování zpráv. tmr: 0 = Ignorovat SMW92 nebo SMW192. 1 = Ukončit příjem, pokud byl překročen čas v SMW92 nebo SMW192 . bk: 0 = Ignorovat pauzu (break). 1 = Použít pauzu (break) pro detekci začátku zprávy.
SMB88	SMB188	Znak začátku zprávy
SMB89	SMB189	Znak konce zprávy
SMW90	SMW190	Doba prázdného úseku uvedená v milisekundách. První znak přijatý po skončení prázdného úseku je startovací znak.
SMW92	SMW192	Hraniční hodnota pro časovač, který měří čas mezi znaky/zprávami (v milisekundách). Pokud je hodnota překročena, je příjem zpráv ukončen.
SMB94	SMB194	Maximální počet přijatých znaků (1 až 255 bytů). Poznámka: Tato hodnota musí být nastavena na předpokládanou maximální velikost zásobníku i v případě, že není použito ukončení příjmu zprávy při dosažení maximálního počtu znaků.

SMW98: Chyby na rozšiřovací I/O sběrnici

Jak popisuje tabulka D-17, SMW98 poskytuje informace o počtu chyb na rozšiřovací I/O sběrnici.

Tabulka D-17 Speciální paměťové byty SMW98

SM byte	Popis
SMW98	Hodnota tohoto paměťového místa je zvýšena při každém zjištění chyby na rozšiřovací I/O sběrnici. Hodnota je vymazána při studeném restartu, může být vymazána uživatelem.

SMB130: Řídicí registr pro Freeport *(viz SMB30)*

Viz tabulku D-11.

SMB131 až SMB165: Registr HSC3, HSC4 a HSC5

Jak popisuje tabulka D-18, SMB131 až SMB165 se používají na monitorování a řízení práce vysokorychlostních čítačů HSC3, HSC4 a HSC5.

Tabulka D-18 Speciální paměťové byty SMB131 až SMB165

SM byte	Popis
SMB131 až SMB135	Rezervováno
SM136.0 až SM136.4	Rezervováno
SM136.5	Stavový bit aktuálního směru počítání HSC3: 1 = počítání nahoru
SM136.6	Stavový bit rovnosti požadované a aktuální hodnoty HSC3: 1 = rovnost
SM136.7	Stavový bit příznaku aktuální hodnota větší než požadovaná u HSC3: 1 = větší
SM137.0 až SM137.2	Rezervováno
SM137.3	Řídicí bit pro směr HSC3: 1 = počítání nahoru
SM137.4	Směr aktualizace HSC3: 1 = aktualizovat směr
SM137.5	Aktualizace požadované hodnoty HSC3: 1 = zapsat novou požadovanou hodnotu do HSC3
SM137.6	Aktualizace aktuální hodnoty HSC3: 1 = zapsat novou aktuální hodnotu do stávající hodnoty HSC3
SM137.7	Bit povolení HSC3: 1 = povolen
SMD138	Nová aktuální hodnota HSC3
SMD142	Nová požadovaná hodnota HSC3
SM146.0 až SM146.4	Rezervováno
SM146.5	Stavový bit aktuálního směru počítání HSC4: 1 = počítání nahoru
SM146.6	Stavový bit rovnosti požadované a aktuální hodnoty HSC4: 1 = rovnost
SM146.7	Stavový bit příznaku aktuální hodnota větší než požadovaná u HSC4: 1 = větší
SM147.0	Aktivní úroveň řídicího bitu pro ”Reset”: 0 = ”Reset” je aktivní na vysoké úrovni, 1 = ”Reset” je aktivní na nízké úrovni
SM147.1	Rezervováno
SM147.2	Výběr rychlosti čítání pro kvadraturní čítače: 0 = 4 x rychlost čítání, 1 = 1 x rychlost čítání
SM147.3	Řídicí bit pro směr HSC4: 1 = počítání nahoru
SM147.4	Směr aktualizace HSC4: 1 = aktualizovat směr
SM147.5	Aktualizace požadované hodnoty HSC4: 1 = zapsat novou požadovanou hodnotu do HSC4
SM147.6	Aktualizace aktuální hodnoty HSC4: 1 = zapsat novou aktuální hodnotu do stávající hodnoty HSC4
SM147.7	Bit povolení HSC4: 1 = povolen

Tabulka D-18 Speciální paměťové byty SMB131 až SMB165, pokračování

SM byte	Popis
SMD148	Nová aktuální hodnota HSC4
SMD152	Nová požadovaná hodnota HSC4
SM156.0 až SM156.4	Rezervováno
SM156.5	Stavový bit aktuálního směru počítání HSC5: 1 = počítání nahoru
SM156.6	Stavový bit rovnosti požadované a aktuální hodnoty HSC5: 1 = rovnost
SM156.7	Stavový bit příznaku aktuální hodnota větší než požadovaná u HSC5: 1 = větší
SM157.0 až SM157.2	Rezervováno
SM157.3	Řídicí bit pro směr HSC5: 1 = počítání nahoru
SM157.4	Směr aktualizace HSC5: 1 = aktualizovat směr
SM157.5	Aktualizace požadované hodnoty HSC5: 1 = zapsat novou požadovanou hodnotu do HSC5
SM157.6	Aktualizace aktuální hodnoty HSC5: 1 = zapsat novou aktuální hodnotu do stávající hodnoty HSC5
SM157.7	Bit povolení HSC5: 1 = povolen
SMD158	Nová aktuální hodnota HSC5
SMD162	Nová požadovaná hodnota HSC5

SMB166 až SMB185: Tabulka definování profilu PTO0, PTO1

Jak popisuje tabulka D-19, SMB166 až SMB185 se používají na zobrazení počtu aktivních kroků profilu a adresy tabulky profilu v paměti V.

Tabulka D-19 Speciální paměťové byty SMB166 až SMB185

SM byte	Popis
SMB166	Aktuální číslo záznamu aktivního kroku profilu pro PTO0
SMB167	Rezervováno
SMW168	Adresa tabulky profilu v paměti V pro PTO0 zadaná jako offset od V0.
SMB170	Stavový byte lineárního PTO0
SMB171	Výsledný byte lineárního PTO0
SMD172	Specifikuje generovanou frekvenci, když je lineární generátor PTO0 ovládán v manuálním režimu. Frekvence se specifikuje jako hodnota double integer v Hz. SMB172 je MSB a SMB175 je LSB
SMB176	Aktuální číslo záznamu aktivního kroku profilu pro PTO1
SMB177	Rezervováno
SMW178	Adresa tabulky profilu v paměti V pro PTO1 zadaná jako offset od V0.
SMB180	Stavový byte lineárního PTO1
SMB181	Výsledný byte lineárního PTO1
SMD182	Specifikuje generovanou frekvenci, když je lineární generátor PTO1 ovládán v manuálním režimu. Frekvence se specifikuje jako hodnota double integer v Hz. SMB182 je MSB a SMB178 je LSB

SMB186 až SMB194: Řízení příjmu zpráv *(viz SMB86 až SMB94)*

Viz tabulku D-16.

SMB200 až SMB549: Stav inteligentních modulů

Jak ukazuje tabulka D-20, SMB200 až SMB549 jsou vyhrazeny pro informace poskytované inteligentními rozšiřovacími moduly, například modulem EM 277 PROFIBUS–DP. Informace, jak modul využívá SMB200 až SMB549, najdete v příloze A uvádějící technická data konkrétního modulu.

U CPU S7-200 s verzí firmware starší než 1.2 musíte instalovat inteligentní modul přímo vedle CPU, aby byla zajištěna kompatibilita.

Tabulka D-20 Byty speciální paměti SMB200 až SMB549

Byty speciální paměti SMB200 až SMB549							
Inteligentní modul na pozici 0	**Inteligentní modul na pozici 1**	**Inteligentní modul na pozici 2**	**Inteligentní modul na pozici 3**	**Inteligentní modul na pozici 4**	**Inteligentní modul na pozici 5**	**Inteligentní modul na pozici 6**	**Popis**
SMB200 až SMB215	SMB250 až SMB265	SMB300 až SMB315	SMB350 až SMB365	SMB400 až SMB415	SMB450 až SMB465	SMB500 až SMB515	Název modulu (16 ASCII znaků)
SMB216 až SMB219	SMB266 až SMB269	SMB316 až SMB319	SMB366 až SMB369	SMB416 až SMB419	SMB466 až SMB469	SMB516 až SMB519	Verze firmware (4 znaky ASCII)
SMW220	SMW270	SMW320	SMW370	SMW420	SMW470	SMW520	Kód chyb
SMB222 až SMB249	SMB272 až SMB299	SMB322 až SMB349	SMB372 až SMB399	SMB422 až SMB449	SMB472 až SMB499	SMB522 až SMB549	Informace specifické pro konkrétní typ modulu

Objednací čísla S7-200

E

CPU	Objednací číslo
CPU 221 DC/DC/DC 6 vstupů/4 výstupů	6ES7 211 – 0AA22 – 0XB0
CPU 221 AC/DC/relé 6 vstupů/4 relé	6ES7 211 – 0BA22 – 0XB0
CPU 222 DC/DC/DC 8 vstupů/6 výstupů	6ES7 212 – 1AB22 – 0XB0
CPU 222 AC/DC/relé 8 vstupů/6 relé	6ES7 212 – 1BB22 – 0XB0
CPU 224 DC/DC/DC 14 vstupů/10 výstupů	6ES7 214 – 1AD22 – 0XB0
CPU 224 AC/DC/relé 14 vstupů/10 relé	6ES7 214 – 1BD22 – 0XB0
CPU 224XP DC/DC/DC 14 vstupů/10 výstupů	6ES7 214 – 1AD22 – 0XB0
CPU 224XP AC/DC/relé 14 vstupů/10 relé	6ES7 214 – 1BD22 – 0XB0
CPU 226 DC/DC/DC 24 vstupů/16 výstupů	6ES7 216 – 2AD23 – 0XB0
CPU 226 AC/DC/relé 24 vstupů/16 relé	6ES7 216 – 2BD23 – 0XB0
Rozšiřovací moduly	**Objednací číslo**
EM 221 24 V DC digitální 8 vstupů	6ES7 221 – 1BF22 – 0XA0
EM 221 digitální 8 AC vstupů (8 x 120/230 V AC)	6ES7 221 – 1EF22 – 0XA0
EM 221 digitální vstup 16 x 24 V DC	6ES7 221 – 1BH22 – 0XA0
EM 222 24 V DC digitální 8 výstupů	6ES7 222 – 1BF22 – 0XA0
EM 222 digitální výstup 8 x relé	6ES7 222 – 1HF22 – 0XA0
EM 222 digitální 8 AC výstupů (8 x 120/230 V AC)	6ES7 222 – 1EF22 – 0XA0
EM 222 digitální výstup 4 x 24 V DC – 5 A	6ES7 222 – 1BD22 – 0XA0
EM 222 digitální výstup 4 x relé – 10 A	6ES7 222 – 1HD22 – 0XA0
EM 223 24 V DC digitální kombinace 4 vstupů/4 výstupů	6ES7 223 – 1BF22 – 0XA0
EM 223 24 V DC digitální kombinace 4 vstupů/4 reléových výstupů	6ES7 223 – 1HF22 – 0XA0
EM 223 24 V DC digitální kombinace 8 vstupů/8 výstupů	6ES7 223 – 1BH22 – 0XA0
EM 223 24 V DC digitální kombinace 8 vstupů/8 reléových výstupů	6ES7 223 – 1PH22 – 0XA0
EM 223 24 V DC digitální kombinace 16 vstupů/16 výstupů	6ES7 223 – 1BL22 – 0XA0
EM 223 24 V DC digitální kombinace 16 vstupů/16 reléových výstupů	6ES7 223 – 1PL22 – 0XA0
EM 231 analogový vstup, 4 vstupy	6ES7 231 – 0HC22 – 0XA0
EM 231 analogový vstup RTD, 2 vstupy	6ES7 231 – 7PB22 – 0XA0
EM 231 analogový vstup z termočlánku, 4 vstupy	6ES7 231 – 7PD22 – 0XA0
EM 232 analogový výstup, 2 výstupy	6ES7 232 – 0HB22 – 0XA0
EM 235 analogová kombinace 4 vstupy/1 výstup	6ES7 235 – 0KD22 – 0XA0
Modemový modul EM 241	6ES7 241 – 1AA22 – 0XA0
Polohovací modul EM 253	6ES7 253 – 1AA22 – 0XA0
EM 277 PROFIBUS – DP slave	6ES7 277 – 0AA22 – 0XA0
(CP 243 – 2) Modul AS – Interface	6GK7 243 – 2AX01 – 0XA0
(CP 243 – 1 IT) Internetový modul (s elektronickou dokumentací na CD)	6GK7 243 – 1GX00 – 0XE0
(CP 243 – 1) Ethernetový modul (s elektronickou dokumentací na CD)	6GK7 243 – 1EX00 – 0XE0

Zásuvné moduly a kabely	Objednací číslo
Paměťový modul, 32K (uživatelský program)	6ES7 291 – 8GE20 – 0XA0
Paměťový modul, 64K (uživatelský program, receptura a protokolování dat)	6ES7 291 – 8GF23 – 0XA0
Paměťový modul, 256K (uživatelský program, receptura a protokolování dat)	6ES7 291 – 8GH23 – 0XA0
Hodiny reálného času s bateriovým modulem	6ES7 297 – 1AA23 – 0XA0
BC 293, bateriový modul pro CPU 22x	6ES7 291 – 8BA20 – 0XA0
Kabel pro rozšiřovací moduly, 0,8 metru, pro CPU 22x/EM	6ES7 290 – 6AA20 – 0XA0
Kabel, RS-232/PPI Multi-Master	6ES7 901 – 3CB30 – 0XA0
Kabel, USB/PPI Multi-Master	6ES7 901 – 3DB30 – 0XA0
Software	**Objednací číslo**
STEP 7 – Micro/WIN 32 (V4.0) jedna licence (CD-ROM)	6ES7 810 – 2CC03 – 0YX0
STEP 7 – Micro/WIN 32 (V4.0) licence upgrade (CD-ROM)	6ES7 810 – 2CC03 – 0YX3
S7-200 Toolbox: TP – Designer pro TP070, verze 1.0 (CD-ROM)	6ES7 850 – 2BC00 – 0YX0
Add-On pro STEP 7 – Micro/WIN: Knihovna instrukcí STEP 7 – Micro/WIN 32, V1.1 (CD-ROM)	6ES7 830 – 2BC00 – 0YX0
S7-200 PC Access V1.0 (OPC Server) jedna licence	6ES7 840 – 2CC01 – 0YX0
S7-200 PC Access V1.0 (OPC Server) multi licence	6ES7 840 – 2CC01 – 0YX1
Konfigurační software WinCC Flexible Micro	6AV6 610 – 0AA01 – 0AA0

Komunikační karty	Objednací číslo
CP 5411: Krátká AT ISA	6GK 1 541 – 1AA00
CP 5511: PCMCIA, typ II	6GK 1 551 – 1AA00
CP 5611: Karta PCI (verze 3.0 nebo vyšší)	6GK 1 561 – 1AA00
Příručky	**Objednací číslo**
Systémový manuál pro programovatelný automat S7-200 (v němčině)	6ES7 298 – 8FA24 – 8AH0
Systémový manuál pro programovatelný automat S7-200 (v angličtině)	6ES7 298 – 8FA24 – 8BH0
Systémový manuál pro programovatelný automat S7-200 (ve francouzštině)	6ES7 298 – 8FA24 – 8CH0
Systémový manuál pro programovatelný automat S7-200 (ve španělštině)	6ES7 298 – 8FA24 – 8DH0
Systémový manuál pro programovatelný automat S7-200 (v italštině)	6ES7 298 – 8FA24 – 8EH0
Manuál pro komunikaci S7-200 v protokolu Point-to-Point Interface (v angličtině/němčině)	6ES7 298 – 8GA00 – 8XH0
Uživatelský manuál pro dotykový panel TP070 (v angličtině)	6AV6 591 – 1DC01 – 0AB0
Návod k obsluze TP170 micro (v angličtině)	6AV6 691 – 1DB01 – 0AB0
Manuál pro komunikační procesor CP 243 – 2 SIMATIC NET AS-Interface Master (v angličtině)	6GK7 243 – 2AX00 – 8BA0
Uživatelský manuál pro WinCC Flexible Micro (v angličtině)	6AV6 691 – 1AA01 – 0AB0

Kabely, síťové konektory a repeatery	**Objednací číslo**
MPI kabel	6ES7 901 – 0BF00 – 0AA0
Síťový kabel PROFIBUS	6XVI 830 – 0AH10
Konektor pro síť s konektorem programovacího portu, svislý kabelový výstup	6ES7 972 – 0BB11 – 0XA0
Konektor pro síť (bez konektoru programovacího portu), svislý kabelový výstup	6ES7 972 – 0BA11 – 0XA0
Sběrnicový konektor RS-485 s kabelovým výstupem 35∀(bez konektoru programovacího portu)	6ES7 972 – 0BA40 – 0XA0
Sběrnicový konektor RS-485 s kabelovým výstupem 35∀(s konektorem programovacího portu)	6ES7 972 – 0BB40 – 0XA0
Svorkovnice CPU 22x/EM, 7 svorek, odnímatelná	6ES7 292 – 1AD20 – 0AA0
Svorkovnice CPU 22x/EM, 12 svorek, odnímatelná	6ES7 292 – 1AE20 – 0AA0
Svorkovnice CPU 22x/EM, 14 svorek, odnímatelná	6ES7 292 – 1AF20 – 0AA0
Svorkovnice CPU 22x/EM, 18 svorek, odnímatelná	6ES7 292 – 1AG20 – 0AA0
Repeater RS-485 IP 20, napěťově oddělený	6ES7 972 – 0AA00 – 0XA0
Operátorská rozhraní	**Objednací číslo**
Textový displej TD 200	6ES7 272 – 0AA30 – 0YA0
Textový displej TD 200C (uživatelské stínítko)[1]	6ES7 272 – 1AA10 – 0YA0
Operátorské rozhraní OP3	6AV3 503 – 1DB10T
Operátorské rozhraní OP7	6AV3 607 – 1JC20 – 0AX1
Operátorské rozhraní OP17	6AV3 617 – 1JC20 – 0AX1
Dotykový panel TP070	6AV6 545 – 0AA15 – 2AX0
Dotykový panel TP170 micro	6AV6 640 – 0CA01 – 0AX0
Různé	**Objednací číslo**
Zarážky DIN lišty	6ES5 728 – 8MAll
12místný blok svorek (CPU 221, CPU 222) sada - 10 kusů	6ES7 290 – 2AA00 – 0XA0
Sada náhradních dvířek, obsahuje po 4 ks od následujících: kryty svorkovnice pro 7, 12, 14, 18, 2x12, 2x14 svorek; přístupová dvířka CPU, přístupová dvířka EM	6ES7 291 – 3AX20 – 0XA0
8místný simulátor vstupů	6ES7 274 1XF00 – 0XA0
14místný simulátor vstupů	6ES7 274 1XH00 – 0XA0
24místný simulátor vstupů	6ES7 274 1XK00 – 0XA0
Překrývané segmenty stínítka TD 200C (balík prázdných překrývaných segmentů)	6ES7 272 – 1AF00 – 7AA0

1 Obsahuje jeden prázdný překrývaný segment pro uživatelské nastavení. Pro další prázdné překrývané segmenty si objednejte překrývané segmenty stínítka TD 200C.

Doby provádění instrukcí STL

F

Doby provádění instrukcí jsou velmi důležité, pokud aplikace obsahuje časově kritické funkce. Doby provádění instrukcí jsou uvedeny v tabulce F-2.



Tip

Když použijete doby provádění z tabulky F-2, měli byste vzít v úvahu vliv signálového toku na instrukci, vliv nepřímé adresace a použití určitých paměťových oblastí na dané doby provádění. Uvedené faktory mohou přímo ovlivnit doby provádění.

Vliv signálového toku

Tabulka F-2 uvádí dobu nutnou pro provedení instrukce, když je signálový tok přítomen (vrchol zásobníku = 1 nebo ON).

Jestliže signálový tok přítomen není, pak je doba provádění dané instrukce 1σs.

Vliv nepřímé adresace

Tabulka F-2 uvádí dobu potřebnou pro provedení instrukce při použití přímé adresace operandů a konstant.

Jestliže instrukce použijí nepřímo adresované operandy, prodlouží se doba provádění instrukce o 14 σs na každý nepřímo adresovaný operand použitý v instrukci.

Vliv přístupu k některým paměťovým oblastem

Přístup k některým paměťovým oblastem, jako jsou například AI, AQ, L a akumulátory, vyžaduje delší dobu provádění.

Tabulka F-1 uvádí dodatečný čas, který je třeba připočíst k době provádění instrukce, jestliže jsou tyto paměťové oblasti specifikovány v operandu.

Tabulka F-1 Čas potřebný pro přístup k paměťovým oblastem

Paměťová oblast	Čas přičítaný k době provádění
Lokální analogový vstup (AI) filtrace zablokována filtrace povolena	 9,4 σs 8,4 σs
Rozšiřovací analogový vstup (AI) filtrace zablokována filtrace povolena	 134 σs 8,4 σs
Lokální analogový výstup (AQ)	92 σs
Rozšiřovací analogový výstup (AQ)	48 σs
Lokální paměť (L)	2,8 σs
Akumulátory (AC)	2,8 σs

Tabulka F-2 Doby provádění instrukcí

Instrukce		σs
=	S použitím: I SM, T, C, V, S, Q, M L	0,24 1,3 10,5
+D		29
−D		29
* D		47
/ D		250
+I		25
−I		25
* I		37
/ I		64
=I	S použitím: Lokálních výstupů Rozšiřovacích výstupů	16 24
+R		71 Typ max. 99
−R		72 Typ max. 100
*R		56 Typ max. 166
/R		177 Typ max. 230
A	S použitím: I SM, T, C, V, S, Q, M L	0,22 0,72 6,1
AB < =, =, >=, >, <, <>		18
AD < =, =, >=, >, <, <>		27
AENO		0,4
AI	S použitím: Lokálních vstupů Rozšiřovacích vstupů	15 21
ALD		0,22
AN	S použitím: I SM, T, C, V, S, Q, M L	0,22 0,72 6,1
ANDB		19
ANDD		30
ANDW		25
ANI	S použitím: Lokálních vstupů Rozšiřovacích vstupů	15 21
AR <=, =, >=, >, <, <>		29
AS=, <>	Doba = Základní + (LM * N) Základní Násobitel délky (LM) N je počet porovnávaných znaků	 33 6,3
ATCH		12
ATH	Doba = Základní + (délka * LM) Základní (konstantní délka) Základní (proměnná délka) Násobitel délky (LM)	 23 31 10,2
ATT		36
AW < =, =, >=, >, <, <>		23
BCDI		35

Instrukce		σs
BITIM		16
BIR	S použitím: Lokálních vstupů Rozšiřovacích vstupů	23 30
BIW	S použitím: Lokálních výstupů Rozšiřovacích výstupů	24 32
BMB	Doba = Základní + (délka * LM) Základní (konstantní délka) Základní (proměnná délka) Násobitel délky (LM)	 10 28 5,7
BMD	Doba = Základní + (délka * LM) Základní (konstantní délka) Základní (proměnná délka) Násobitel délky (LM)	 11 29 10,6
BMW	Doba = Základní + (délka * LM) Základní (konstantní délka) Základní (proměnná délka) Násobitel délky (LM)	 10 28 8,6
BTI		16
CALL	Bez použití parametrů: S použitím parametrů: Doba = Základní + (doba operandu) Základní Doba operandu bit (vstupní, výstupní) byte (vstupní, výstupní) word (vstupní, výstupní) double word (vstupní, výstupní) Poznámka: zpracování výstupních operandů probíhá během návratu z podprogramu	9 14 10, 11 8, 7 10, 9 12, 10
CEVNT		24
CFND	Maximální doba = Základní + N1 * ((LM1 * N2) + LM2) Základní Násobitel délky 1 (LM1) Násobitel délky 2 (LM2) N1 je délka zdrojového řetězce N2 je délka znakového řetězce	 35 8,6 9,5
CITIM		23
COS		900 Typ max. 1070
CRET	Signálový tok přítomen Signálový tok nepřítomen	16 0,8
CRETI	Signálový tok nepřítomen	0,2
CSCRE		3,1
CTD	Při přechodu na vstupu pro čítání Jinak	27 19
CTU	Při přechodu na vstupu pro čítání Jinak	31 19
CTUD	Při přechodu na vstupu pro čítání Jinak	37 24
DECB		16
DECD		22
DECO		19
DECW		20
DISI		9
DIV		67

Instrukce		σs
DLED		14
DTA		302
DTI		21
DTCH		12
DTR		35 Typ max. 40
DTS		305
ED		8
ENCO		max. 24
END	Signálový tok nepřítomen	0,2
ENI		11
EU		8
EXP		720 Typ max. 860
FIFO	Doba = Základní + (délka * LM) Základní Násobitel délky (LM)	 30 7
FILL	Doba = Základní + (délka * LM) Základní (konstantní délka) Základní (proměnná délka) Násobitel délky (LM)	 15 29 3,2
FND <, =, >, <> Doba = Základní + (délka * LM) Základní Násobitel délky (LM)		 39 6,5
FOR	Doba = Základní + (Počet smyček * LM) Základní Násobitel smyčky (LM)	 35 28
GPA		16
HDEF		18
HSC		30
HTA	Doba = Základní + (délka * LM) Základní (konstantní délka) Základní (proměnná délka) Násobitel délky (LM)	 20 28 5,2
IBCD		52
INCB		15
INCD		22
INCW		20
INT	Obvykle s 1 přerušením	24
INVB		16
INVD		22
INVW		20
ITA		136
ITB		17
ITD		20
ITS		139
JMP		1,8
LBL		0,22

Instrukce		σs
LD	S použitím: I SM, T, C, V, S, Q, M L	0,22 0,8 6
LDB <=, =, >=, >, <, <>		18
LDD <=, =, >=, >, <, <>		27
LDI	S použitím: Lokálních vstupů Rozšiřovacích vstupů	15 21
LDN	S použitím: I SM, T, C, V, S, Q, M L	0,3 0,9 6,1
LDNI	S použitím: Lokálních vstupů Rozšiřovacích vstupů	15 21
LDR<=, =, >=, >, <, <>		29
LDS		0,22
LDS=, <> Doba = Základní + (LM * N) Základní Násobitel délky (LM) N je počet porovnávaných znaků		 33 6,3
LDW <=, =, >=, >, <, <>		24
LIFO		37
LN		680 Typ max. 820
LPP		0,22
LPS		0,24
LRD		0,22
LSCR		7,3
MOVB		15
MOVD		20
MOVR		20
MOVW		18
MUL		37
NETR		99
NETW	Doba = Základní + (LM * N) Základní Násobitel délky (LM) N je počet odesílaných bytů	 95 4
NEXT		0
NOP		0,22
NOT		0,22
O	S použitím: I SM, T, C, V, S, Q, M L	0,22 0,72 6,4
OB < =, =, >=, >, <, <>		18
OD < =, =, >=, >, <, <>		26
OI	S použitím: Lokálních vstupů Rozšiřovacích vstupů	15 21
OLD		0,22
ON	S použitím: I SM, T, C, V, S, Q, M L	0,22 0,72 6,4

Instrukce		σs
ONI	S použitím: Lokálních vstupů Rozšiřovacích vstupů	15 21
OR<=, =, >=, >, <, <>		29
ORB		19
ORD		29
ORW		25
OS=, < >	Doba + Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet porovnávaných znaků	 33 6,3
OW < =, =, >=, >, <, <>		24
PID	Běžně Manuální – automatický přenos Přepočet koeficientu Automatické ladění	400 max. 800 max. 770 max. 650
PLS:	S použitím: PWM PTO v režimu s jedním segmentem PTO v režimu s více segmenty	31 36 50
R	Délka = 1 a specifikována jako konstanta Základní pro čítače (C) Základní pro časovače (T) Základní pro všechny ostatní Jinak: Doba = Základní + (délka ∗ LM) Základní pro čítače Základní pro časovače (T) Základní pro všechny ostatní Násobitel délky (LM) pro operand C Násobitel délky (LM) pro operand T Násobitel délky (LM) pro všechny ostatní Je–li délka uložena jako proměnná, přidat k základní.	 9,3 16 2,9 8,6 8,3 14 5,1 9,9 0,5 17
RCV		51
RET		16
RI	Doba = Základní + (délka ∗ LM) Základní Násobitel délky (LM) s použitím lokálních výstupů Násobitel délky (LM) s použitím rozšiřovacích výstupů Je–li délka uložena jako proměnná, přidat k základní.	 8,9 13 21 17
RLB	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet posunů	 23 0,2
RLD	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet posunů	 28 1,4
RLW	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet posunů	 27 0,9
ROUND		56 Typ max. 110
RRB	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet posunů	 22 0,5
RRD	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM)) N je počet posunů	 28 1,7

Instrukce		σs
RRW	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet posunů	 26 1,2
RTA	Doba = Základní + (LM ∗ N) Základní (pro první číslici ve výsledku) Násobitel délky (LM) N je počet dalších číslic ve výsledku	 149 96
RTS	Doba = Základní + (LM ∗ N) Základní (pro první číslici ve výsledku x) Násobitel délky (LM) N je počet dalších číslic ve výsledku	 154 96
S	Pro délku = 1 a specifikovanou jako konstanta Jinak: Doba = Základní + (délka ∗ LM) Základní Násobitel délky (LM) Pokud je délka uložena jako proměnná, přičtěte Základní	2,9 14 0,5 17
SCAT	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet připojených znaků	 30 5,3
SCPY	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet kopírovaných znaků	 27 4,6
SCRE		0,24
SCRT		10
SEG		15
SFND	Maximální doba = Základní +((N1–N2) ∗ LM2) +(N2 ∗ LM1) Základní Násobitel délky 1 (LM1) Násobitel délky 2 (LM2) N1 je délka zdrojového řetězce N2 je délka vyhledávaného řetězce	 39 7,6 6,8
SHRB	Doba = Základní + (délka ∗ LM1) + ((délka /8) ∗ LM2) Základní (konstantní délka) Základní (proměnná délka) Násobitel délky 1 (LM1) Násobitel délky 2 (LM2)	 48 52 1,0 1,5
SI	Doba = Základní + (délka ∗ LM) Základní LM s použitím lokálního výstupu LM s použitím rozšiřovacího výstupu Pokud je délka uložena jako proměnná, přičtěte Základní	 8,9 13 21 17
SIN		900 Typ max. 1070
SLB	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet posunů	 23 0,2
SLD	Doba = Základní + (LM ∗ N) Základní Násobitel délky (LM) N je počet posunů	 29 1,1

Instrukce		σs
SLEN		21
SLW	Doba = Základní + (LM + N) Základní Násobitel délky (LM)) N je počet posunů	 27 0,6
SPA		371
SQRT		460 Typ max. 550
SRB	Doba = Základní + (LM + N) Základní Násobitel délky (LM)) N je počet posunů	 22 0,6
SRD	Doba = Základní + (LM + N) Základní Násobitel délky (LM) N je počet posunů	 28 1,5
SRW	Doba = Základní + (LM + N) Základní Násobitel délky (LM)) N je počet posunů	 27 1
SSCPY	Doba = Základní + (LM * N) Základní Násobitel délky (LM) N je počet kopírovaných znaků	 42 5,3
STD	Doba = Základní + (LM * N) Základní (pro 1. zdrojový znak) Násobitel délky (LM) N je počet dalších zdrojových znaků	 69 27
STI	Doba = Základní + (LM * N) Základní (pro první zdrojový znak) Násobitel délky (LM) N je počet dalších zdrojových znaků	 58 27

Instrukce		σs
STOP	Signálový tok nepřítomen	4
STR	Doba = Základní + (LM * N) Základní (pro 1. zdrojový znak) Násobitel délky (LM) N je počet dalších zdrojových znaků	 51 81
SWAP		17
TAN		1080 Typ max. 1300
TODR		331
TODRX	 Oprava letního času	391 Typ 783 Typ
TODW		436
TODWX		554
TOF		36
TON		33
TONR		32
TRUNC		53 Typ max. 106
WDR		7
XMT		42
XORB		19
XORD		29
XORW		25

Stručný přehled informací o S7-200

Pro usnadnění vyhledávání informací uvádí tato kapitola stručný přehled následujícího:

- Speciální paměťové bity
- Popisy událostí přerušení
- Stručný přehled rozsahu a vlastností pamětí CPU S7-200
- Vysokorychlostní čítače HSC0, HSC1, HSC2, HSC3, HSC4, HSC5
- Instrukce pro S7-200

Tabulka G-1 Speciální paměťové bity

Speciální paměťové bity			
SM0.0	Vždy zapnutý	SM1.0	Výsledek operace = 0
SM0.1	První programový cyklus	SM1.1	Přeplnění nebo nepřípustná hodnota
SM0.2	Remanentní data ztracena	SM1.2	Záporný výsledek
SM0.3	Studený start	SM1.3	Dělení nulou
SM0.4	30 s vypnutý / 30 s zapnutý	SM1.4	Plná tabulka
SM0.5	0,5 s vypnutý / 0,5 s zapnutý	SM1.5	Prázdná tabulka
SM0.6	Vypnutý na 1 programový cyklus / zapnutý na 1 programový cyklus	SM1.6	Chyba převodu BCD na binární číslo
SM0.7	Přepínač v poloze RUN	SM1.7	Chyba převodu ASCII na hexadecimální číslo

Tabulka G-2 Události přerušení seřazené podle priority

Číslo události	Popis přerušení	Prioritní skupina	Priorita ve skupině
8	Port 0: Příjem znaku	Komunikace (nejvyšší priorita)	0
9	Port 0: Ukončení přenosu		0
23	Port 0: Ukončení příjmu zprávy		0
24	Port 1: Ukončení příjmu zprávy		1
25	Port 1: Příjem znaku		1
26	Port 1: Ukončení přenosu		1
19	Přerušení od ukončení PTO 0	Diskrétní (střední priorita)	0
20	Přerušení od ukončení PTO 1		1
0	I0.0, Náběžná hrana		2
2	I0.1, Náběžná hrana		3
4	I0.2, Náběžná hrana		4
6	I0.3, Náběžná hrana		5
1	I0.0, Sestupná hrana		6
3	I0.1, Sestupná hrana		7
5	I0.2, Sestupná hrana		8
7	I0.3, Sestupná hrana		9
12	HSC0 CV=PV (aktuální hodnota = nastavená hodnota)		10
27	HSC0 změna směru		11
28	HSC0 externí reset		12
13	HSC1 CV=PV (aktuální hodnota = nastavená hodnota)		13
14	HSC1 změna směru na vstupu		14
15	HSC1 externí reset		15
16	HSC2 CV=PV (aktuální hodnota = nastavená hodnota)		16
17	HSC2 změna směru		17
18	HSC2 externí reset		18
32	HSC3 CV=PV (aktuální hodnota = nastavená hodnota)		19
29	HSC4 CV=PV (aktuální hodnota = nastavená hodnota)		20
30	HSC4 změna směru		21
31	HSC4 externí reset		22
33	HSC5 CV=PV (aktuální hodnota = nastavená hodnota)		23
10	Časované přerušení 0	Časovaná (nejnižší priorita)	0
11	Časované přerušení 1		1
21	Přerušení od časovače T32 CT=PT		2
22	Přerušení od časovače T96 CT=PT		3

Tabulka G-3 Rozsahy pamětí

Popis	CPU 221	CPU 222	CPU 224	CPU 224XP	CPU 226
Velikost uživatelského programu					
s editací v režimu RUN	4096 bytů	4096 bytů	8192 bytů	12288 bytů	16384 bytů
bez editace v režimu RUN	4096 bytů	4096 bytů	12288 bytů	16384 bytů	24576 bytů
Velikost uživatelských dat	2048 bytů	2048 bytů	8192 bytů	10240 bytů	10240 bytů
Registr obrazu vstupů	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7
Registr obrazu výstupů	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7
Analogové vstupy (pouze čtení)	AIW0 až AIW30	AIW0 až AIW30	AIW0 až AIW62	AIW0 až AIW62	AIW0 až AIW62
Analogové výstupy (pouze zápis)	AQW0 až AQW30	AQW0 až AQW30	AQW0 až AQW62	AQW0 až AQW62	AQW0 až AQW62
Paměť pro proměnné (V)	VB0 až VB2047	VB0 až VB2047	VB0 až VB8191	VB0 až VB10239	VB0 až VB10239
Lokální paměť (L)[1]	LB0 až LB63	LB0 až LB63	LB0 až LB63	LB0 až LB63	LB0 až LB63
Bitová paměť (M)	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7
Speciální paměť (SM)	SM0.0 až SM179.7	SM0.0 až SM299.7	SM0.0 až SM549.7	SM0.0 až SM549.7	SM0.0 až SM549.7
Pouze čtení	SM0.0 až SM29.7	SM0.0 až SM29.7	SM0.0 až SM29.7	SM0.0 až SM29.7	SM0.0 až SM29.7
Časovače	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)
Remanentní zpožděné zapnutí 1 ms	T0, T64	T0, T64	T0, T64	T0, T64	T0, T64
10 ms	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68
100 ms	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95
Zpožděné zapnutí/ vypnutí 1 ms	T32, T96	T32, T96	T32, T96	T32, T96	T32, T96
10 ms	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100
100 ms	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255
Čítače	C0 až C255	C0 až C255	C0 až C255	C0 až C255	C0 až C255
Vysokorychlostní čítače	HC0 až HC5	HC0 až HC5	HC0 až HC5	HC0 až HC5	HC0 až HC5
Sekvenční řídicí relé (S)	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7
Registry akumulátoru	AC0 až AC3	AC0 až AC3	AC0 až AC3	AC0 až AC3	AC0 až AC3
Skoky/návěští	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
Volání/podprogramy	0 až 63	0 až 63	0 až 63	0 až 63	0 až 127
Přerušovací podprogramy	0 až 127	0 až 127	0 až 127	0 až 127	0 až 127
Náběžné/sestupné hrany	256	256	256	256	256
Smyčky PID	0 až 7	0 až 7	0 až 7	0 až 7	0 až 7
Porty	Port 0	Port 0	Port 0	Port 0, Port 1	Port 0, Port 1

[1] LB60 až LB63 jsou rezervovány pro STEP 7–Micro/WIN, verze 3.0 nebo pozdější.

Tabulka G-4 Vysokorychlostní čítače HSC0, HSC3, HSC4 a HSC5

Režim	HSC0			HSC3	HSC4			HSC5
	Pulzy	Směr	Reset	Pulzy	Pulzy	Směr	Reset	Pulzy
0	I0.0			I0.1	I0.3			I0.4
1	I0.0		I0.2		I0.3		I0.5	
2								
3	I0.0	I0.1			I0.3	I0.4		
4	I0.0	I0.1	I0.2		I0.3	I0.4	I0.5	
5								
Režim	HSC0				HSC4			
	Pulzy nahoru	Pulzy dolů	Reset		Pulzy nahoru	Pulzy dolů	Reset	
6	I0.0	I0.1			I0.3	I0.4		
7	I0.0	I0.1	I0.2		I0.3	I0.4	I0.5	
8								
Režim	HSC0				HSC4			
	Fáze A	Fáze B	Reset		Fáze A	Fáze B	Reset	
9	I0.0	I0.1			I0.3	I0.4		
10	I0.0	I0.1	I0.2		I0.3	I0.4	I0.5	
11								
Režim	HSC0			HSC3				
	Pulzy			Pulzy				
12	Q0.0			Q0.1				

Tabulka G-5 Vysokorychlostní čítače HSC1 a HSC2

Režim	HSC1				HSC2			
	Pulzy	Pulzy dolů	Reset	Start	Pulzy	Směr	Reset	Start
0	I0.6				I1.2			
1	I0.6		I1.0		I1.2		I1.4	
2	I0.6		I1.0	I1.1	I1.2		I1.4	I1.5
3	I0.6	I0.7			I1.2	I1.3		
4	I0.6	I0.7	I1.0		I1.2	I1.3	I1.4	
5	I0.6	I0.7	I1.0	I1.1	I1.2	I1.3	I1.4	I1.5
Režim	HSC1				HSC2			
	Pulzy nahoru	Pulzy dolů	Reset	Start	Pulzy nahoru	Pulzy dolů	Reset	Start
6	I0.6	I0.7	I1.0		I1.2	I1.3		
7	I0.6	I0.7	I1.0		I1.2	I1.3	I1.4	
8	I0.6	I0.7	I1.0	I1.1	I1.2	I1.3	I1.4	I1.5
Režim	Fáze A	Fáze B	Reset	Start	Fáze A	Fáze B	Reset	Start
9	I0.6	I0.7			I1.2	I1.3		
10	I0.6	I0.7	I1.0		I1.2	I1.3	I1.4	
11	I0.6	I0.7	I1.0	I1.1	I1.2	I1.3	I1.4	I1.5

Booloevské instrukce		
LD	bit	Uloží hodnotu na vrchol zásobníku
LDI	bit	Uloží okamžitou hodnotu
LDN	bit	Uloží negovanou hodnotu
LDNI	bit	Nahraje negovanou okamžitou hodnotu
A	bit	A
AI	bit	AND s okamžitou hodnotou
AN	bit	AND s negovanou hodnotou
ANI	bit	AND s negovanou okamžitou hodnotou
O	bit	OR
OI	bit	OR s okamžitou hodnotou
ON	bit	OR s negovanou hodnotou
ONI	bit	OR s negovanou okamžitou hodnotou
LDBx	IN1, IN2	Uloží výsledek porovnání bytů IN1 (x:<, <=,=, >=, >, <>I) IN2
ABx	IN1, IN2	AND výsledek porovnání bytů IN1 (x:<, <=,=, >=, >, <>) IN2
OBx	IN1, IN2	OR výsledek porovnání bytů IN1 (x:<, <=,=, >=, >, <>) IN2
LDWx	IN1, IN2	Uloží výsledek porovnání wordů IN1 (x:<, <=,=, >=, >, <>) IN2
AWx	IN1, IN2	AND výsledek porovnání wordů IN1 (x:<, <=,=, >=, >, <>) IN2
OWx	IN1, IN2	OR výsledek porovnání wordů IN1 (x:<, <=,=, >=, >, <>) IN2
LDDx	IN1, IN2	Uloží výsledek porovnání double word IN1 (x:<, <=,=, >=, >, <>) IN2
ADx	IN1, IN2	AND výsledek porovnání double word IN1 (x:<, <=,=, >=, >, <>) IN2
ODx	IN1, IN2	OR výsledek porovnání double word IN1 (x:<, <=,=, >=, >, <>) IN2
LDRx	IN1, IN2	Uloží výsledek porovnání reálných čísel IN1 (x:<, <=,=, >=, >, <>) IN2
ARx	IN1, IN2	AND výsledek porovnání reálných čísel IN1 (x:<, <=,=, >=, >, <>) IN2
ORx	IN1, IN2	OR výsledek porovnání reálných čísel IN1 (x:<, <=,=, >=, >, <>) IN2
NOT		Negace zásobníku
EU		Detekce náběžné hrany
ED		Detekce sestupné hrany
=	Bit	Přiřazení hodnoty
=I	Bit	Okamžité přiřazení hodnoty
S	Bit, N	Set (nastavení bitu)
R	Bit, N	Reset (nulování bitu)
SI	Bit, N	Okamžité nastavení bitu
RI	Bit, N	Okamžité nulování bitu
LDSx	IN1, IN2	Uloží výsledek porovnání řetězce IN1 (x: =, <>) IN2
ASx	IN1, IN2	AND výsledek porovnání řetězce IN1 (x: =, <>) IN2
OSx	IN1, IN2	OR výsledek porovnání řetězce IN1 (x: =, <>) IN2
ALD		Logický součin s vrcholem zásobníku
OLD		Logický součet s vrcholem zásobníku
LPS		Duplikování horní hodnoty zásobníku
LRD		Zkopírování druhé hodnoty zásobníku
LPP		Vybrání hodnoty na vrcholu ze zásobníku
LDS	N	Duplikování bitu n zásobníku
AENO		Logický součin s hodnotou ENO

Instrukce matematiky, inkrementace a dekrementace		
+I	IN1, OUT	Přičtení integer, double integer nebo reálného čísla
+D	IN1, OUT	
+R	IN1, OUT	IN1+OUT=OUT
–I	IN1, OUT	Odečtení integer, double integer nebo reálného čísla
–D	IN1, OUT	
–R	IN1, OUT	OUT–IN1=OUT
MUL	IN1, OUT	Vynásobení integer (16*16–>32)
*I	IN1, OUT	Vynásobení integer, double integer nebo reálného čísla
*D	IN1, OUT	
*R	IN1, IN2	IN1 * OUT = OUT
DIV	IN1, OUT	Dělení integer (16/16–>32)
/I	IN1, OUT	Dělení integer, double integer nebo reálného čísla
/D,	IN1, OUT	
/R	IN1, OUT	OUT / IN1 = OUT
SQRT	IN, OUT	Druhá odmocnina
LN	IN, OUT	Přirozený logaritmus
EXP	IN, OUT	Přirozený exponenciál
SIN	IN, OUT	Sinus
COS	IN, OUT	Kosinus
TAN	IN, OUT	Tangens
INCB	OUT	Inkrementace bytu, word nebo double word
INCW	OUT	
INCD	OUT	
DECB	OUT	Dekrementace bytu, word nebo double word
DECW	OUT	
DECD	OUT	
PID	TBL, LOOP	Smyčka PID

Instrukce časovače a čítače		
TON	Txxx, PT	Časovač zpožděného zapnutí
TOF	Txxx, PT	Časovač zpožděného vypnutí
TONR	Txxx, PT	Remanentní časovač zpožděného zapnutí
BITIM	OUT	Začátek časového intervalu
CITIM	IN, OUT	Vypočítat časový interval
CTU	Cxxx, PV	Čítání nahoru
CTD	Cxxx, PV	Čítání dolů
CTUD	Cxxx, PV	Čítání nahoru/dolů

Instrukce hodin reálného času		
TODR	T	Přečtení času z hodin
TODW	T	Zapsání času do hodin
TODRX	T	Rozšířené čtení hodin reálného času
TODWX	T	Nastavení rozšířeného čtení hodin reálného času

Instrukce řízení programu		
END		Podmíněné ukončení programu
STOP		Přechod do režimu STOP
WDR		Reset watchdogu (300 ms)
JMP	N	Skok na definované návěští
LBL	N	Definuje návěští pro skok
CALL	N [N1,...]	Volání podprogramu [N1, ... až do 16 volitelných parametrů]
CRET		Podmíněný návrat z podprogramu
FOR	INDX,INIT,FINAL	Smyčka For/Next
NEXT		
LSCR	N	Uložení, přechod, podmíněné ukončení a ukončení sekvenčního relé
SCRT	N	
CSCRE		
SCRE		
DLED	IN	Diagnostická kontrolka

Instrukce přesunu, posunu a rotace		
MOVB MOVW MOVD MOVR	IN, OUT IN, OUT IN, OUT IN, OUT	Přesun hodnot typu byte, word, double word, reálné číslo
BIR BIW	IN, OUT IN, OUT	Přesun bytu s okamžitým čtením Přesun bytu s okamžitým zápisem
BMB BMW BMD	IN, OUT, N IN, OUT, N IN, OUT, N	Přesun bloku hodnot typu byte, word, double word
SWAP	IN	Záměna vyššího a nižšího bytu
SHRB	DATA, S_BIT, N	Posun bitů posuvného registru
SRB SRW SRD	OUT, N OUT, N OUT, N	Posun bytu, word, double word doprava
SLB SLW SLD	OUT, N OUT, N OUT, N	Posun bytu, word, double word doleva
RRB RRW RRD	OUT, N OUT, N OUT, N	Rotace bytu, word, double word doprava
RLB RLW RLD	OUT, N OUT, N OUT, N	Rotace bytu, word, double word doleva
Logické instrukce		
ANDB ANDW ANDD	IN1, OUT IN1, OUT IN1, OUT	Logický součin (AND) hodnot typu byte, word a double word
ORB ORW ORD	IN1, OUT IN1, OUT IN1, OUT	Logický součet (OR) hodnot typu byte, word a double word
XORB XORW XORD	IN1, OUT IN1, OUT IN1, OUT	Logická nonekvivalence (XOR) hodnot typu byte, word a double word
INVB INVW INVD	OUT OUT OUT	Inverze byte, word a double word (jednotkový doplněk)
Instrukce pro řetězce		
SLEN	IN, OUT	Délka řetězce
SCAT	IN, OUT	Spojování řetězců
SCPY	IN, OUT	Kopírování řetězce
SSCPY	IN, INDX, N, OUT	Kopírování části řetězce
CFND	IN1, IN2, OUT	Nalezení prvního znaku uvnitř řetězce
SFND	IN1, IN2, OUT	Nalezení části řetězce

Instrukce tabulky, vyhledávání a konverze		
ATT	DATA, TBL	Přidání dat do tabulky
LIFO FIFO	TBL, DATA TBL, DATA	Výběr dat z tabulky
FND= FND<> FND< FND>	TBL, PTN, INDX TBL, PTN, INDX TBL, PTN, INDX TBL, PTN, INDX	Nalezení datové hodnoty, která odpovídá srovnání (v tabulce)
FILL	IN, OUT, N	Vyplnění místa v paměti
BCDI IBCD	OUT OUT	Konverze BCD na integer Konverze hodnoty integer na BCD
BTI ITB ITD DTI	IN, OUT IN, OUT IN, OUT IN, OUT	Konverze byte na integer Konverze integer na byte Konverze integer na double integer Konverze double integer na integer
DTR TRUNC ROUND	IN, OUT IN, OUT IN, OUT	Konverze double word na reálné číslo Konverze real čísla na double integer Konverze real čísla na double integer
ATH HTA ITA DTA RTA	IN, OUT, LEN IN, OUT, LEN IN, OUT, FMT IN, OUT, FM IN, OUT, FM	Konverze ASCII na HEX hodnotu Konverze HEX hodnoty na ASCII Konverze hodnoty integer na ASCII Konverze double integer na ASCII Konverze reálného čísla na ASCII
DECO ENCO	IN, OUT IN, OUT	Dekódování Kódování
SEG	IN, OUT	Generuje hodnotu pro sedmisegmentový displej
ITS DTS RTS	IN, FMT, OUT IN, FMT, OUT IN, FMT, OUT	Konverze integer na řetězec Konverze double integer na řetězec Konverze reálného čísla na řetězec
STI STD STR	STR, INDX, OUT STR, INDX, OUT STR, INDX, OUT	Konverze části řetězce na integer Konverze části řetězce na double integer Konverze části řetězce na reálné číslo
Instrukce přerušení		
CRETI		Podmíněný návrat z přerušení
ENI DISI		Povolení přerušení Zákaz přerušení
ATCH DTCH	INT, EVNT EVNT	Přiřazení podprogramu přerušení události Zrušení přiřazení přerušení
Instrukce komunikace		
XMT RCV	TBL, PORT TBL, PORT	Přenos v režimu Freeport Příjem zprávy v režimu Freeport
NETR NETW	TBL, PORT TBL, PORT	Čtení ze sítě Zápis do sítě
GPA SPA	ADDR, PORT ADDR, PORT	Načtení adresy portu Nastavení adresy portu
Vysokorychlostní instrukce		
HDEF	HSC, MODE	Definuje režim vysokorychlostního čítače
HSC	N	Aktivace vysokorychlostního čítače
PLS	Q	Pulzní výstup

Rejstřík

Symboly

&, 33
*, 33

A

ACCEL_TIME (doba zrychlení), polohovací modul
  EM 253, 256
Adresa
  Modbus, 349
  nastavení vzdálené, 213
  přiřazení, 62
  síť, 211
  symbolická, 62
  vysokorychlostní čítače, 126
Adresa nejvyšší stanice (HSA), 233
Adresování
  akumulátory, 29
  analogové vstupy, 30
  analogové výstupy, 31
  bitová paměť, 27
  bity speciální paměti (SM), 30
  byte:bit, 26
  lokální paměť, 30
  lokální vstupy/výstupy, 32
  nepřímé (ukazatele), 33
  paměť časovače, 28
  paměť čítače, 28
  paměť pro proměnné, 27
  paměť S7–200, 26
  paměť sekvenčních řídicích relé (SCR), 31
  paměťové oblasti, 27–30
  přímé, 26
  registr obrazu vstupů, 27
  registr obrazu výstupů, 27
  rozšiřitelné vstupy/výstupy, 32
  vysokorychlostní čítače, 29
AENO, instrukce, 75
AI. *Viz* Analogový vstup (AI)
Aktualizace, vlny PWM, 136
Aktuální hodnota
  nastavení pro HSC, 125
  změna v HSC, 131
Akumulátory, 29
American Bureau of Shipping (ABS) Maritime
  Agency, 382
Analogové moduly, 3
  EM 231 analogový vstup, 404
  EM 231 RTD, 411
  EM 231 termočlánkový, 411
  EM 232 analogové výstupy, 408
  EM 235 analogové vstupy/výstupy, 405
Analogové potenciometry
  potenciometry, 50
  SMB28 a SMB29, 470
Analogové výstupy, konfigurování hodnot, 44
  instrukce ”AND”, 162
  příklad, 163
Analogový I/O modul EM 235
  blokové schéma vstupů, 407
  blokové schéma výstupů, 408
  formát vstupního datového word, 406
  formát výstupního datového word, 407
  instalace, 408
  kalibrace, 403
  konfigurace, 405
  přesnost a opakovatelnost, 409
  specifikace, 410
Analogový vstup (AI)
  adresace, 30
  filtrace, 45
Analogový vstupní modul EM 231
  blokové schéma vstupů, 406
  formát vstupního datového word, 406
  instalace, 408
  kalibrace, 403
  přesnost a opakovatelnost, 409
  specifikace, 410
Analogový výstup (AQ), adresování, 31
Analogový výstupní modul EM 232
  blokové schéma výstupů, 408
  formát výstupního datového word, 407
AND Load, instrukce, 75
AQ. *Viz* Analogový výstup (AQ)
AS–I Interface CP 243–2, specifikace, 448
Asynchronní aktualizace, instrukce PWM, 136
Automatická hystereze, 375
Automatické ladění PID, 372
Automatické ladění PID
  automatická hystereze, 375
  chyby, 377
  odchylka, 375
  předpoklady, 375
  PV mimo rozsah, 377
  sekvence, 376
  výjimečné podmínky, 377
  vysvětlení, 372
Automatický režim, smyčka PID, 151

B

Bateriový modul, 450
  objednací číslo, 480
BCD na integer, instrukce, 100
Běh programu, 14
Bezpečnostní obvody, navrhování, 52
Blokové schéma, modem EM 241, 435
Blokové schéma vstupů
  EM 231, 406
  EM 235, 407
Blokové schéma výstupů
  EM 232, 408
  EM 235, 408
Booleovské instrukce
  cívky, 73
  kontakty, 70
  logický zásobník, 75

RS a SR klopný obvod, 77
Bureau Veritas (BV), 382
Byte na integer, instrukce, 100
Bytová konzistence, PROFIBUS, 428
Bytový adresní formát, 26

C

CP karty
objednací čísla, 480
volba, 224
CP karty, podporované STEP 7 Micro/WIN, 225
CP5411, 480
CP5511, 480
CP5611, 480
CPU
digitální vstup/výstup, 2
hodiny reálného času, 2
ID registr (SMB6), 468
komunikační porty, 2
ochrana heslem, 48
ošetření chyb, 60
paměť, 2
počet připojení, 214
požadavky na napájení, 17, 457
programový cyklus, 24
prováděcí rychlost, 2
rozměry, 2
rozšiřovací moduly, 2
simulátory vstupu, 456
velikost šroubů pro instalaci, 18
vyhledávání v síti, 213
zálohování, 2
CPU, podporující inteligentní moduly, 323
CPU 224, vzorový program DP, 433
CPU moduly
autodiagnostika, 25
demontáž, 19
instalace, 18
objednací čísla, 385, 479
rozsahy paměti, 68, 491
schémata zapojení, 390 – 392
specifikace, 385, 386
specifikace napájení, 387
specifikace vstupů, 387
specifikace výstupů, 388
vlastnosti, 68
Cyklická přerušení, registry časového intervalu (SMB34, SMB35), 471

Č

Čas, nastavení, 78
Čas cyklu (funkce PTO), 133
Časovač měřící čas mezi znaky, 91
Časovač zpoždění náběžné hrany, instrukce (TON), 196, 201
příklad, 198
Časovač zpoždění sestupné hrany, instrukce (TOF), 196, 201
příklad, 199
Časovač zpráv, 91
Čísla, reprezentace, 26, 32
Číslo, reprezentace, 31
Čítač dolů, instrukce, 113
příklad, 115
Čítač nahoru, instrukce, 113
Čítače, vysokorychlostní, 50
Čtení dat ze vstupů, 24, 25
Čtení hodin reálného času, instrukce, 78
Čtení ze sítě, instrukce, 81, 82
chybové kódy, 82
příklad, 84

D

Data
příjem, 87, 92
uložení a obnovení, 36
vysílání, 87
Databázový soubor zařízení (GSD), EM 277 PROFIBUS – DP, 431 – 432
Datový blok, 54
Datum, nastavení, 78
DATx_WRITE, podprogram datového záznamu, 369
DECEL_TIME (Doba zpomalení), polohovací modul EM 253, 256
Definice vysokorychlostního čítače, instrukce (HDEF), 118
Dekódovat, instrukce, 112
příklad, 112
Dekrementace, instrukce, 144
příklad, 144
Dělení, instrukce, 140
Dělení integer se zbytkem, instrukce (DIV), 142
příklad, 142
Délka řetězce, instrukce, 184
Derivační člen, algoritmus PID, 148
Det Norske Veritas (DNV), 382
Diagnostická kontrolka, 47
Diagnostické informace, polohovací modul EM 253, 291
Diagnostika, autodiagnostika CPU, 25
Dialogové okno ”Informace PLC” 60
Digitální rozšiřovací modul, adresace, 32
Digitální vstupní obvod, 47
Digitální vstupy
čtení, 42
filtrace, 45
zachycení pulzu, 45
zápis do, 25
Digitální výstupy, konfigurace stavů, 43
DIP přepínače
RS – 232/PPI Multi – Master kabel, 454
RTD, 418 – 419
termočlánek, 414
Diskrétní moduly, 3
Doba oběhu tokenu, 233
porovnání, 235
Doby programového cyklu: SMW22 až SMW26, 470
Doby provádění, instrukce STL, 483
Dotykový panel TP070, 5

objednací čísla, 481
objednací číslo manuálu, 480
Double integer na ASCII, instrukce, 104
Double integer na integer, instrukce, 100
Double integer na real, instrukce, 100
Download
konfigurace datových záznamů, 36
program, 14, 36
receptury, 36
Druhá odmocnina, instrukce, 143
Dvoubodová (peer–to–peer) komunikace, 217–218

E

Editace, v režimu RUN, 244
Editace režimu RUN, 43
vypnutí, 48
Editor datového bloku
přiřazení adres, 62
přiřazení počátečních hodnot, 62
Editor FBD
konvence, 58
popis, 56
vlastnosti, 56
Editor LAD
konvence, 58
popis, 56
vlastnosti, 56
Editor STL
popis, 55
vlastnosti, 55
Editory
funkční bloky (FBD), 56
kontaktní schéma (LAD), 56
výpis instrukcí (STL), 55
EEPROM
chybové kódy, 462
uložení bitové paměti (M), 38
uložení proměnné paměti (V), 39, 471
Elektrická životnost, 383
Elektromagnetická
kompatibilita, S7–200, 383
normy pro emise, 383
normy pro odolnost, 383
EN, 59, 67
ENO, 59, 67
Ethernet, 214
TCP/IP, 215
Ethernetová síť, CP 243–1, 219
Ethernetový modul, průvodce, 444
Ethernetový modul CP 243–1
funkce, 444
komunikační procesory, 444
konfigurace, 444
objednací číslo, 443, 479
počet připojení, 215
připojení, 444
rádce, 444
specifikace, 443

F

Faktor obnovení mezery (GUF), 233
FIFO, instrukce, 190
příklad, 190
Filtrace
analogové vstupy, 45
digitální vstupy, 45
Formát telefonního čísla pro zprávu, modemový modul EM 24, 327
Formát textové zprávy, modemový modul EM 241, 328
Formát vstupního datového word
EM 231, 406
EM 235, 406
Formát výstupního datového word
EM 232, 407
EM 235, 407
Formát zprávy s přenosem dat CPU, modemový modul EM 241, 329
Freeport, konfigurace, PPI Multi–Master kabel, 231
Fronta časovaných přerušení, 157
Fronta přerušení od vstupů/výstupů, 157
Fronty, podprogramy přerušení, 157
Funkce, Modbus, 350
Funkce paměti
instrukce posunu, 179
instrukce posunutí bitů posuvného registru, 181
instrukce přesunu, 164
instrukce přesunu bloku dat, 166
instrukce rotace, 179
instrukce výměny, 183
Funkce PTO/PWM, registry (WMB66 až SMB85), 473
Funkce zachycování impulzů, 46
Funkční bloky *viz* Editor FBD

G

Germanisher Lloyd (GL), 382

H

Hardware, odstraňování poruch, 249
Heslo
funkce CPU, 48
konfigurace, 49
mazání, 49
obnovení ztraceného, 49
omezení přístupu, 49
Hex na ASCII, instrukce, 103
Hledání, instrukce, 193
Hledání řetězce, instrukce, 187
Hledání znaku, instrukce, 187
Hodiny
stavové bity, 466
zásuvný modul, 450
Hodiny, instrukce
čtení hodin reálného času, 78

nastavení čtení hodin reálného času, 79
nastavení hodin reálného času, 78
rozšířené čtení hodin reálného času, 79
Hodiny reálného času (TOD), 78
Hodiny TOD, 78
Hodnoty
data S7 – 200, 26, 33
přímá adresace, 26
stavová tabulka, 63
Hodnoty reálných čísel, 26, 31
Hodnoty s plovoucí desetinnou čárkou, 31, 149
Hodnoty tabulky profilu, generátory PTO/PWM, 138
HSC, 50
průvodce instrukcemi, 119
HyperTerminal, konfigurace RS – 232/PPI
Multi – Master kabelu, 239 – 483

CH

Chybové kódy, 462
chyby polohovacího modulu EM 253, 293
instrukce pro polohovací modul EM 253, 292
instrukce PWMxRUN, 254
instrukce USS protokolu, 342
kritické chyby, 462
polohovací modul EM 253, 292
porušení pravidel kompilace, 464
problémy programování v době provádění programu, 463
PTO instrukce, 264
Chybové kódy instrukcí, polohovací modul EM 253, 292
Chybové kódy modulu, polohovací modul EM 253, 293
Chybové kódy provádění, instrukce protokolu USS, 342
Chyby
automatické ladění PID, 377
instrukce čtení a zápisu v síti, 82
instrukce modemového modulu EM 241, 321
kompilace programu, 60
konfigurace modemového modulu EM 241, 313
kritické, 61
nekritické, 60
ošetření, 60
prohlížení, 60
provádění programu, 61
provádění protokolu Modbus slave, 353
SMB1, chyby provádění, 466
smyčky PID, 151
v době běhu programu, 60
vstupy a výstupy (I/O), 60
Chyby kompilace, 60
Chyby parity
SMB3, 467
SMB30 a SMB130, 92
Chyby provádění, 61
Chyby v době běhu programu, 60

I

Identifikační a chybový registr I/O modulu, 469
IEC instrukce časovače, 201
příklad, 201
IEC instrukce čítače
příklad, 116
sestupného čítače, 116
vzestupného čítače, 116
vzestupného/sestupného čítače, 116
Indukční zátěže, 22
Inicializace
vysokorychlostní čítače, 127
protokolu Modbus, 348
Inkrementace, instrukce, 144
příklad, 144
Instrukce
AENO, 75
AND, 162
AND Load, 75
ASCII na Hex, 103
BCD na integer 100bitové logiky, 70
bez výstupů, 59
časovač zpoždění náběžné hrany (TON), 196, 201
časovač zpoždění náběžné hrany s pamětí (TONR), 196
časovač zpoždění sestupné hrany (TOF), 196, 201
čtení, 75
čtení hodin reálného času, 78
čtení ze sítě, 81
definice vysokorychlostního čítače (HDEF), 118
dekódování, 112
dekrementace, 144
dělení, 140
dělení integer se zbytkem (DIV), 142
délka řetězce, 184
double integer na ASCII, 104
double integer na integer, 100
double integer na real, 100
druhá odmocnina, 143
FIFO, 190
for, 169
hex na ASCII, 103
hodiny reálného času, 78
impulzní časovač (TP), 201
inkrementace, 144
integer na ASCII, 103
integer na BCD, 100
integer na byte, 100
integer na double integer, 100
inverze byte, 161

inverze double word, 161
inverze word, 161
konec sekvenčního řídícího relé, 172
konverze bytu na integer, 100
konverze double integer na řetězec, 107, 110
konverze integer na řetězec, 107, 110
konverze reálného čísla na řetězec, 107, 110
konverze řetězce na double integer, 107, 110
konverze řetězce na integer, 107, 110
konverze řetězce na reálné číslo, 107, 110
kopírování části z řetězce, 186
kopírování řetězce, 184
kosinus, 143
LIFO, 190
MBUS_INIT, 351
MBUS_SLAVE, 353
modemový modul EM 241, 318
MODx_CTRL, 319
MODx_MSG, 320
MODx_XFR, 319
náběžná hrana, 70
načtení adresy portu, 95
načtení sekvenčního řídícího relé, 172
načtení zásobníku, 75
nalezení části řetězce, 187
nalezení znaku uvnitř řetězce, 187
naplnění paměti, 192
násobení, 140
násobení integer s 32bitovým výsledkem, (MUL), 142
nastavení, 73
nastavení adresy portu, 95
nastavení čtení hodin reálného času, 79
nastavení hodin reálného času, 78
návěští, 171
návrat z podprogramu, 204
návrat z přerušení, 153
negativní přechod, 70
next, 169
NOT, 70
odčítání, 140
odpojení přerušení, 153
okamžité čtení a přesun bytu, 165
okamžité nastavení, 73
OR, 162
OR načtení, 75
ořezání, 101
podmíněný konec sekvenčního řídícího relé, 172
podmíněný návrat z podprogramu, 204
podmíněný návrat z přerušení, 153
podmíněné ukončení, 167
polohovací modul EM 253, 273
pop, 75
porovnání, 59
posun bytu vlevo, 179
posun bytu vpravo, 179
posun double word vlevo, 179
posun double word vpravo, 179
posun word vlevo, 179
posun word vpravo, 179
posunutí bitů posuvného registru, 181
POSx_CFG, 284
POSx_CLR, 283
POSx_CTRL, 274
POSx_DIS, 282
POSx_GOTO, 276
POSx_LDOFF, 279
POSx_LDPOS, 280
POSx_MAN, 275
POSx_RSEEK, 278
POSx_RUN, 277
POSx_SRATE, 281
povolení přerušení, 153
prohledávání tabulky, 193
provádění, 25
protokol Modbus slave, 350
přechod sekvenčního řídícího relé, 172
přemístit blok, 166
přemístit blok double word, 166
přemístit blok word, 166
přerušení, 153–159
přesun a okamžitý zápis bytu, 165
přesun bytu, 164
přesun double word, 164
přesun reálného čísla, 164
přesun word, 164
přidávání do tabulky, 189
příjem, 86
přímé vynulování, 73
přímý kontakt, 70
přímý výstup, 73
připojení přerušení, 153
přirozený exponenciál, 143
přirozený logaritmus, 143
PTOx_ADV, 264
PTOx_CTRL, 260
PTOx_LDPOS, 263
PTOx_MAN, 262
PTOx_RUN, 261
pulzní šířkové modulace (PWM), 133
pulzní výstup (PLS), 133
push, 75
PWMx_RUN, 254
real na ASCII, 105
reset watchdogu, 167
rotace bytu doleva, 179
rotace bytu doprava, 179
rotace double word doleva, 179
rotace double word doprava, 179
rotace word doleva, 179
rotace word doprava, 179
rozšířené čtení hodin reálného času, 79
RS klopný obvod, 77
řízení smyčky (PID), 145
sčítání (Add), 140
segment, 102
sestupné čítače, 113
sestupný čítač, 116
sinus, 143
skok na návěští, 171
smyčky PID, 145
spojování řetězců, 184
SR klopný obvod, 77
standardní kontakt, 70
stop, 167
tabulky, 190–195

tangens, 143
tvorba pohybu, 302
ukončení, 167
USS protokol, 333
volání podprogramu, 204
vymazat událost pro přerušení, 153
výměna bytů, 183
vynulování, 73
vysílání, 86
vysokorychlostní čítač (HSC), 118
vysokorychlostní pulzní výstup (PLS), 133
výstup, 73
výstup série pulzů (PTO), 133
vzestupné čítače, 113
vzestupné/sestupné čítače, 114
vzestupný čítač, 116
vzestupný/sestupný čítač, 116
XOR, 162
zakázání přerušení, 153
zakódování, 112
zaokrouhlení, 101
zápis do sítě, 81
žádná operace, 73
Instrukce, stručný přehled, 493
Instrukce, vložení, 12
Instrukce ”ASCII na Hex”, 103
příklad, 106
Instrukce bitové logiky
instrukce cívky, 73
instrukce kontaktů, 70
instrukce logického zásobníku, 75
RS klopný obvod, 77
RS a SR klopný obvod, 77
Instrukce cívky
nastavení, 73
okamžitého nastavení, 73
přímého vynulování, 73
přímého výstupu, 73
vynulování, 73
výstupu, 73
žádná operace, 73
Instrukce časovače
IEC
časovač zpoždění náběžné hrany (TON), 201
časovač zpoždění sestupné hrany (TOF), 201
impulzní časovač (TP), 201
přerušení, 157
SIMATIC
časovač zpoždění náběžné hrany (TON), 196
časovač zpoždění náběžné hrany s pamětí (TONR), 196
časovač zpoždění sestupné hrany (TOF), 196
Instrukce čítače
definice vysokorychlostního čítače (HDEF), 118
IEC
sestupného čítače, 116
vzestupného čítače, 116
vzestupného/sestupného čítače, 116
SIMATIC
sestupné čítače, 113
vzestupné čítače, 113
vzestupného/sestupného čítače, 114
vysokorychlostní čítač (HSC), 118
Instrukce dolů čítače, 116
Instrukce hodin reálného času, 78
Instrukce ”Invertovat byte”, 161
Instrukce ”Invertovat double word”, 161
Instrukce ”Invertovat word”, 161
Instrukce inverze, příklad, 161
Instrukce komunikace
čtení ze sítě, 81
načtení adresy portu, 95
nastavení adresy portu, 95
příjmu, 86
vysílání, 86
zápis do sítě, 81
Instrukce kontaktů, 70
příklad, 72
Instrukce logického zásobníku
AENO, 75
AND Load, 75
čtení, 75
načtení zásobníku, 75
OR, 75
pop, 75
příklad, 76
push, 75
Instrukce logických operací
AND, OR, XOR, 162
inverze, 161
Instrukce matematiky
dekrementace, 144
dělení, 140
dělení integer se zbytkem (DIV), 142
inkrementace, 144
násobení, 140
násobení integer s 32bitovým výsledkem, (MUL), 142
odčítání, 140
sčítání (Add), 140
Instrukce matematiky celých čísel, příklad, 141
Instrukce matematiky reálných čísel, příklad, 141
Instrukce MODx_CTRL, modemový modul EM 241, 319
Instrukce MODx_XFR, modemový modul EM 241, 319
Instrukce MODx_MSG, modemový modul EM 241, 320
Instrukce nahoru/dolů čítače, 114
příklad, 115
Instrukce nonekvivalence (XOR), 162
příklad, 163
Instrukce numerických funkcí
druhá odmocnina, 143
kosinus, 143
přirozený exponenciál, 143
přirozený logaritmus, 143
sinus, 143
tangens, 143
Instrukce podprogramu
podmíněný návrat z podprogramu, 204
příklad, 207
volání podprogramu, 204
Instrukce porovnání, 59
porovnání bytu, 96
porovnání double word, 96

porovnání integer, 96
porovnání reálných čísel, 96
porovnání řetězců, 98
příklad, 96
Instrukce posunu
příklad, 180
typy, 179
Instrukce pro konverzi
ASCII, 103
ASCII na hex, 103
dekódování, 112
hex na ASCII, 103
ořezání, 101
příklad, 101
řetězec, 107, 110
segment, 102
standardní, 99
zakódování, 112
zaokrouhlení, 101
Instrukce pro řetězce
délky řetězce, 184
kopírování části z řetězce, 186
kopírování řetězce, 184
nalezení části řetězce, 187
nalezení znaku uvnitř řetězce, 187
spojování řetězců, 184
Instrukce pro standardní konverzi, 99
Instrukce protokolu USS
chybové kódy provádění, 342
návod k užívání, 333
USS4_DRV_CTRL, 335
USS4_INIT, 334
USS4_RPM_x and USS4_WPM_x, 338, 339
vzorový program, 341
Instrukce přechodu, 70
Instrukce přerušení
odpojení přerušení, 153
příklad, 159
připojení přerušení, 153
podmíněný návrat z přerušení, 153
povolení přerušení, 153
vymazat událost pro přerušení, 153
zakázání přerušení, 153
Instrukce přímých kontaktů, 70
Instrukce PTO, chybové kódy, 264
Instrukce PWMx_RUN, chybové kódy, 254
Instrukce rotace, 179
příklad, 180
typy, 179
Instrukce rozšířeného čtení hodin reálného času, 79
Instrukce rozšířeného nastavení hodin reálného času, 79
Instrukce řízení programu
instrukce skoku, 171
podmíněné ukončení, 167
reset watchdogu, 167
sekvenční řídicí relé (SCR), 172
smyčka for – next, 169
stop, 167
základní řízení programu, 167
Instrukce sekvenčního řídicího relé
divergence řízení, 175
konec sekvenčního řídicího relé, 172
konvergence řízení, 175
načtení sekvenčního řídicího relé, 172
omezení, 172
podmíněný konec sekvenčního řídicího relé, 172
přechod sekvenčního řídicího relé, 172
příklad, 172
Instrukce smyčky for – next
for, 169
next, 169
příklad, 170
Instrukce smyčky PID, 145
automatický režim, 151
derivační člen, 148
chyby, 151
integrační člen, 147
kontrola poruchových signálů, 151
konverze vstupů smyčky, 148
přímá, 150
konverze výstupů smyčky na integer s upraveným měřítkem, 149
manuální režim, 151
normalizace vstupů smyčky, 148
proměnné, 150
proporcionální člen, 147
rádce, 145
reverzní, 150
režimy, 151
rozsahy, 150
řízení smyčky, 148
tabulka smyčky, 152
typy řízení smyčky, 148
vysvětlení, 146
Instrukce standardního kontaktu, 70
Instrukce STL
doby provádění, 483
stručný přehled, 493
Instrukce tabulky
FIFO, 190
LIFO, 190
naplnění paměti, 192
prohledávání tabulky, 193
přidávání do tabulky, 189
Instrukce vymazání události pro přerušení, 153
Instrukce vysokorychlostního čítače (HSC), 118
příklad, 132
Instrukce vzestupného čítače, 116
Instrukce vzestupného/sestupného čítače, 116
Instrukční soubor IEC 1131 – 3, 57
Instrukční soubory
IEC 1131 – 3, 57
SIMATIC, 57
volba, 57
Instrukční strom, 11, 55
Integer na ASCII, instrukce, 103
příklad, 106
Integer na BCD, instrukce, 100
Integer na byte, instrukce, 100
Integer na double integer, instrukce, 100
Integer s upraveným měřítkem, konverze výstupů smyčky, 149
Integrační člen, algoritmus PID, 147
Inteligentní moduly, 3
CPU podporující, 323

stav (SMB200 až SMB549), 477
Internetová zařízení, CP 243 – 1 IT, 219
Internetový modul, průvodce, 447
Internetový modul CP 243 – 1 IT
funkce, 446
komunikační procesory, 446
konfigurace, 447
objednací číslo, 446, 479
počet připojení, 215
průvodce pro Internet, 447
připojení, 447
specifikace, 446

K

Kabel pro rozšiřovací moduly, 451
Kabel pro rozšiřovací moduly, instalace, 451
Kabeláž, 20, 21
Kabely
konfigurování
Freeport, 231
vzdálený modem, 230
objednací čísla, 480
rozšiřovací sběrnice, 451
RS – 232/PPI Multi – Master, 452
síť, 221, 222
symetrizace, 223
ukončení, 223
USB/PPI Multi – Master, 452
volba, 210
KalibraceEM 231, 403
EM 235, 403
vstupů, 403
Knihovna protokolu Modbus, 347
Knihovna USS protokolu, řízení pohonu MicroMaster, 331
Knihovny instrukcí, 64
Knihovny instrukcí STEP 7 – Micro/WIN 32, 480
Kódy zemí, podporované EM 241, 308
Kompatibilita
EM 231 RTD, 412
EM 231 termočlánkový, 412
EM 253 polohovací, 439
EM 277 PROFIBUS – DP, 424
modem EM 241, 436
Komunikace, 210
čas v pozadí, 43
konflikty, 236
modem, 231 – 232
podporované protokoly, 214
protokol Modbus slave, 348
přenosové rychlosti, 231, 232
rádiový modem, 232 – 233
s pohony MicroMaster, 332
S7 – 200, 10
výběr portu, 210
Komunikace PPI, změna na režim Freeport, 87
Komunikace s pohonem, výpočet potřebné doby, 332
Komunikace v režimu Freeport, 467
SMB30 a SMB130, 470
Komunikační fronta, 157
Komunikační karty, objednací čísla, 480
Komunikační port
přerušení, 156
režim Freeport, 225
rozmístění vývodů konektoru, 223
Komunikační procesor CP 243 – 2
činnost, 449
funkce, 449
připojení, 449
rádce, 448
Komunikační procesory
AS – Interface CP 243 – 2, 448
ethernetového modulu CP 243 – 1, 444
internetového modulu CP 243 – 1 IT, 446
Komunikační procesory. *Viz* CP karty
Komunikační protokol
definovaný uživatelem, 225
dvoubodové spojení (point – to – point) (PPI), 214, 235
mnohabodové spojení (multipoint) (MPI), 215, 235
PROFIBUS, 215, 235
TCP/IP, 215
volba, 214
Konec, instrukce, 167
Konec sekvenčního řídícího relé, instrukce, 172
Konektory, objednací čísla, 480
Konfigurace
EM 231, 404
EM 231 RTD, 418
EM 231 termočlánkový, 413
EM 235, 405
EM 277 PROFIBUS – DP, 426 – 427
ethernetového modulu CP 243 – 1, 444
funkce PTO/PWM, 136
hesla, 49
hodnoty analogových výstupů, 44
HyperTerminal, 239 – 483
internetového modulu CP 243 – 1 IT, 447
polohovacího modulu EM 253, 291
remanentní oblasti paměti, 44
síť, 221
stavy digitálních výstupů, 43
tabulka symbolů pro Modbus, 349
zobrazení stavu FBD a LAD, 246
zobrazení stavu STL, 247
Konflikty v síti, 236
Konstanty, 32
Kontrola poruchových signálů, smyčka PID, 151
Kontrolky (LED) diagnostické, 47
modemový modul EM 241, 314
Kontrolní panel pro EM 253, 290 – 292
Konvence
programování S7 – 200, 59
programové editory, 58
Konvergence řízení
instrukce sekvenčního řídicího relé, 175
příklad, 176
Konverze
vstupů smyčky, 148

výstupů smyčky na integer s upraveným měřítkem, 149
Konverze ASCII, instrukce, 103
Konverze double integer na řetězec, instrukce, 107, 110
Konverze integer na řetězec, instrukce, 107, 110
Konverze real na řetězec, instrukce, 107, 110
Konverze substring na double integer, instrukce, 107, 110
Konverze substring na integer, instrukce, 107, 110
Konverze substring na real, instrukce, 107, 110
Konzistence dat, PROFIBUS, 428
Konzistence wordů, PROFIBUS, 428
Kopírování řetězce, instrukce, 184
Kopírování řetězců, instrukce, 186
Kosinus, instrukce, 143
Kritické chyby, 61
prohlížení, 462
Krokové motory, řízení polohy otevřené smyčky, 255
Kroky, vytváření pro profil pohybu, 258

L

Ladění
editace v režimu RUN, 244
více programových cyklů, 248
vlastnosti, 244
vnucení hodnot, 248
Ladění softwaru, 243
Last – In – First – Out, instrukce, 190
příklad, 191
LIFO (kontaktní schéma) *Viz* Editor LAD
Lišta
montáž, 18
rozměry, 18
Lloyds Register of Shipping (LRS), 382
Logická spojení
MPI, 215
PPI, 214
Logický pop, instrukce, 75
Logický push, instrukce, 75
Logický read, instrukce, 75
Logika, řídicí, 24
Lokální vstup/výstup, 32

M

Manuální režim, smyčka PID, 151
Manuály, objednací čísla, 480
Mapování adresy pro S7 – 200, Modbus, 349
Maximální a spouštěcí/zastavovací rychlost, průvodce polohováním, 255
Maximální počet přijatých znaků, 92
MBUS_INIT, instrukce, 351
MBUS_SLAVE, instrukce, 353
Mezinárodní normy, 382
Micro/WIN. *Viz* STEP 7 – Micro/WIN
Mikro – PLC, navrhování, 52
Místa speciální paměti, polohovací modul EM 253, 298
Modem
s PPI Multi – Master kabelem, 231
vzdálený, PPI Multi – Master kabel, 230
Modem Expansion Wizard, 230
Modemový modul, 435
formát textové zprávy, 328
formát zprávy s přesunem dat CPU, 329
chyby z instrukcí, 321
instrukce, 318
konektor RJ11, 308
Modem Expansion Wizard, 314
MODx_CTRL, instrukce, 319
MODx_MSG, instrukce, 320
MODx_XFR, instrukce, 319
ochrana heslem, 311
přenos numerických zpráv pro pager 310
přenos textových zpráv, 310
přenos zpráv pro pager, 310
přesuny dat, 311
příklad, 323
rozhraní mezinárodní telefonní linky, 308
SMS, posílání zpráv, 310
stavové LED, 314
tabulka konfigurace, 313
telefonní číslo pro přenos zpráv, 327
textové zprávy, 310
vlastnosti, 308
zabezpečení zpětným voláním, 312
Modemový modul EM 241
adresy Modbus, 310
blokové schéma, 435
formát zprávy s přesunem dat CPU, 329
chyby z instrukcí, 321
instalace, 436
instrukce, 318
kompatibilita s CPU, 436
konektor RJ11, 308
místa speciální paměti, 323
MOD_XFR, instrukce, 319
Modem Expansion Wizard, 314
MODx_CTRL, instrukce, 319
MODx_MSG, instrukce, 320
ochrana heslem, 311
podporované země, 308
protokol Modbus RTU, 309
přenos numerických zpráv pro pager, 310
přenos textových zpráv, 310
přenos zpráv pro pager, 310
přesuny dat, 311
příklad, 323
rozhraní mezinárodní telefonní linky, 308
rozhraní STEP 7 – Micro/WIN, 309
Short Message Service – služba krátkých zpráv (SMS), 310
specifikace, 435
stavové LED, 314
tabulka konfigurace, 313, 325
telefonní číslo pro přenos zpráv, 327
vlastnosti, 308
zabezpečení zpětným voláním, 312
Moduly AS – Interface, objednací čísla manuálu, 480
Modul AS – Interface CP 243 – 2
objednací číslo, 448
vlastnosti, 448

Modul AS – Interface CP 243 – 2, objednací číslo 479
Modul EM 277 PROFIBUS – DP
  další vlastnosti, 430
  jako DP slave, 425
  kompatibilita s CPU, 424
  konfigurační soubor, 431 – 432
  konfigurování, 426 – 427
  počet připojení, 214
  protokol DP, 425
  přepínače adres, 424
  režim výměny dat, 428
  speciální paměťové byty, 429
  specifikace, 423
  stavové LED, 424, 430
  v síti PROFIBUS, 426
  volby konfigurace, 427
Montáž
  DIN lišta, 18
  elektrické rušení, 16
  EM 231, 408
  EM 235, 408
  kabel pro rozšiřovací moduly, 451
  modulu CPU, 18
  napájení, 17
  nároky na prostor, 16, 18
  návody, 16
  panel, 18
  požadavky na montáž, 18
  rozměry, 18
  rozšiřovacího modulu, 18
  S7 – 200, 17
  STEP 7 – Micro/WIN, 4
  vysokonapěťová zařízení, 16
  zařízení vytvářející teplo, 16
Montáž na DIN lištu, 18
Montáž na panel, 18
MPI, protokol, 215, 235
Multi – Master kabel, 9

N

Náběžná hrana, instrukce, 70, 71
Načtení sekvenčního řídicího relé, instrukce, 172
Načtení zásobníku, instrukce, 75
Najít v tabulce, instrukce, 193
  příklad, 195
Námořní agentura Nippon Kaiji Kyokai (NK), 382
Napájení, 8, 17
Napěťové oddělení
  návod pro kabeláž, 20
  síť, 221
Naplnění paměti, instrukce, 192
  příklad, 192
Nároky na prostor, 16
Násobení, instrukce, 140
Násobení integer na double integer (MUL), instrukce, 142
  příklad, 142
Nastavení
  aktuálních a požadovaných hodnot pro HSC, 125
  času, 78
  data, 78
  přenosové rychlosti, 211
  řídicího bytu (HSC), 124
  vzdálené adresy pro S7 – 200, 213
Nastavení adresy portu, instrukce, 95
Nastavení DIP přepínače, PPI Multi – Master kabel, 9, 227
Nastavení hodin reálného času, instrukce, 78
Nastavení komunikace, STEP 7 – Micro/WIN, 10
Nastavování
  pohonu MM3, 343
  pohonu MM4, 346
Nástrojové lišty, 55
Návěští, instrukce, 171
Navigační lišta, 55
Návrat z podprogramu, instrukce, 204
  příklad, 207
Návrat z přerušení, instrukce, 153
Návrh
  bezpečnostních obvodů, 52
  mikro – PLC, 52
Navrhování systému, mikro – PLC, 52
Nekritické chyby, 60
Nepodmíněné vstupy, 59
Neprogramovatelný terminál, konfigurace
  RS – 232/PPI Multi – Master kabelu, 239 – 483
Nepřímá adresace, 33
  & a *, 33
  změny ukazatelů, 34
NETR, NETW, průvodce instrukcemi, 81
Next, instrukce, 169
  příklad, 170
Normalizace vstupů smyčky, 148
Normy, státní a mezinárodní, 382
NOT, instrukce, 70
Nulový modem, 231

O

Objednací čísla, 479 – 922
  analogový rozšiřovací modul, 400, 411
  CPU moduly, 385
  digitální rozšiřovací modul, 394
  modul AS – Interface CP 243 – 2, 448
  PPI Multi – Master kabely, 452
  simulátory vstupu, 456
Oblast bitové paměti (M), 27
  uložení při ztrátě napájení, 38
Oblast lokální paměti (L), 30
Oblast paměti časovače (T), 28
Oblast paměti čítače (C), 28
Oblast paměti sekvenčních řídicích relé (S), 31
Oblast proměnné paměti (V), 27
Oblast speciální paměti (SM), 30
  analogové potenciometry, 50
Obnovení dat po zapnutí napájení, 39
Obnovení programu z paměťového modulu, 38
Obnovení ztraceného hesla, 49
Odčítání, instrukce, 140
Odchylka, 375
Odpojit přerušení, instrukce, 153
Odstraňování problémů
  hardware S7 – 200, 249

chybové kódy, 462
kritické chyby, 61
návod, 249
nekritické chyby, 60
Ochrana heslem, modemový modul EM 241, 311
Ochranná dioda, 22
Ochranné obvody, 22
Ochranný obvod, 383
Okamžitý set, instrukce, 73
Omezení přístupu, 48
OP3, OP7, OP17, objednací čísla, 481
Operátorská rozhraní, objednací číslo, 481
Operátorské stanice, specifikování, 52
Opětná montáž svorkovnice, 19
Oprávnění, 382
Optimalizace výkonu sítě, 233
OR, instrukce, 162
příklad, 163
OR Load, instrukce, 75
Oříznout, instrukce, 101
Otevření programového editoru, 11
Ovládací panel ladění PID, 378

P

Paměť
CPU, 2
přístup, 26
remanentní, 44
Paměť C, 28
Paměť I, 27
Paměť L, 30
Paměť M, 27
Paměť pro program, zvýšení, 48
Paměť Q, 27
Paměť remanentních dat, 36
Paměť S, 31
Paměť SM, 30
funkce PTO/PWM, 136
protokol Modbus slave, 348
Paměť T, 28
Paměť V, 27
kopírování do permanentní paměti, 40
přiřazení adres, 62
uložení do EEPROM, 39
Paměťové oblasti
mazání, 49
rozsahy operandů, 69
Paměťový modul, 36, 450
elektrostatický výboj, 36
chybové kódy, 462
uložení programu, 37
Paměťový modul, objednací číslo, 480
Paměťový modul, obnovení programu z, 38
Parametry
typy pro podprogramy, 205
v podprogramech, 205
Permanentní paměť, 36, 39
Permanentní paměť, kopírování paměti V, 40
PID, tabulka smyčky, rozšířená, 372
Počáteční hodnoty, přiřazení, 62
Podmíněné přechody, příklad, 177
Podmíněné vstupy, 59
Podmíněný konec, instrukce, 167
příklad, 168
Podmíněný návrat z podprogramu, instrukce, 204
Podmíněný návrat z přerušení, instrukce, 153
Podmínkový konec sekvenčního řídícího relé, instrukce, 172
Podmínky prostředí
doprava a skladování, 383
provozní, 383
Podprogram časovaného přerušení, příklad, 159
Podprogram RCPx_Read, instrukce receptury, 361
Podprogramy
návody, 54
parametr signálového toku, 205
příklad, 53
s parametry, 205
typy dat, 205
typy parametrů, 205
vnoření, 204
volání z přerušovacích podprogramů, 156
Podprogramy přerušení, 25, 42
fronty, 157
komunikační port, 156
náběžná/sestupná hrana, 156
návody, 54, 155
od času, 156
priorita, 157
příklad, 53
sdílení dat s hlavním programem, 155
systémová podpora, 155
typy podporované S7–200, 156
volání podprogramů z, 156
vstupy a výstupy (I/O), 156
vysvětlení, 155
Pohon MicroMaster
čtení a psaní, 338, 339
komunikace, 332
připojení, 342
řízení, 331
Pohon MM3
nastavení, 343
připojení, 342
Pohon MM4
nastavení, 346
připojení, 345
Pohony. *Viz* Pohony MicroMaster
Pohybové příkazy, polohovací modul EM 253, 300
Pohybový profil
definování, 257
polohovací modul, 257
vytvoření kroků, 258
Polohovací modul
ACCEL_TIME, 256
DECEL_TIME, 256
diagnostické informace, 291
eliminace vůle, 306
chybové kódy, 292
chybové kódy instrukcí, 292
chybové kódy modulu, 293
instrukce, 273
konfigurace, 267, 291
konfigurace profilů pohybu, 257

kontrolní panel pro EM 253, 290 – 292
návod pro instrukce, 273
pohybové příkazy, 300
Position Control Wizard, 267
POSx_CFG, 284
POSx_CLR, 283
POSx_CTRL, 274
POSx_DIS, 282
POSx_GOTO, 276
POSx_LDOFF, 279
POSx_LDPOS, 280
POSx_MAN, 275
POSx_RSEEK, 278
POSx_RUN, 277
POSx_SRATE, 281
programování, 266
režimy hledání RP, 303 – 307
speciální paměť, 298
tabulka konfigurace/profilů, 294
tvorba instrukcí, 302
vlastnosti, 265
vstupy a výstupy, 265
vzorový program řízení, 302
zobrazení a řízení práce, 290
Polohovací modul EM 253
ACCEL_TIME, 256
DECEL_TIME, 256
diagnostické informace, 291
eliminace vůle, 306
chybové kódy, 292
chybové kódy instrukcí, 292
chybové kódy modulu, 293
instrukce, 273
kompatibilita s CPU, 439
konfigurace, 267, 291
kontrolní panel pro EM 253, 290 – 292
návod pro instrukce, 273
pohybové příkazy, 300
Position Control Wizard, 267
POSx_CFG, 284
POSx_CLR, 283
POSx_CTRL, 274
POSx_DIS, 282
POSx_GOTO, 276
POSx_LDOFF, 279
POSx_LDPOS, 280
POSx_MAN, 275
POSx_RSEEK, 278
POSx_RUN, 277
POSx_SRATE, 281
programování, 266
příkazový byte, 299
příklady, 285 – 289
režimy hledání RP, 303 – 307
schéma zapojení, 441 – 442
speciální paměť, 298
tabulka konfigurace/profilů, 294
tvorba instrukcí, 302
vlastnosti, 265
vstupy a výstupy, 265
vzorový program řízení modulu, 302
zobrazení a řízení práce, 290
Porovnání bytu, instrukce, 96
Porovnání doby oběhu tokenu, 235
Porovnání double word, instrukce, 96
Porovnání integer, instrukce, 96
Porovnání real, instrukce, 96
Porovnání řetězců, instrukce, 98
Port, nastavení, PPI Multi – Master kabely, 220
Porušení pravidel kompilace, 464
Pořadí přerušovacích událostí, 158
Position Control Wizard, 133
Posunout register bit, instrukce, 181
příklad, 182
Posunout vlevo byte, instrukce, 179
Posunout vlevo double word, instrukce, 179
Posunout vpravo byte, instrukce, 179
Posunout vpravo double word, instrukce, 179
Posunout vpravo word, instrukce, 179
POSx_CFG, 284
POSx_CLR, 283
POSx_CTRL, 274
POSx_DIS, 282
POSx_GOTO, 276
POSx_LDOFF, 279
POSx_LDPOS, 280
POSx_MAN, 275
POSx_RSEEK, 278
POSx_RUN, 277
POSx_SRATE, 281
Potenciometry, analogové nastavení, 50
Potlačení šumu, vstupní filtr, 45
Povolení vstupu (EN), 59
Povolení vysokorychlostních čítačů, 131
Povolení výstupu (ENO), 59
Povolit přerušení, instrukce, 153
Požadavky komunikace, zpracování, 25
Požadavky na napájení, 17
CPU, 457
rozšiřovacího modulu, 457
tabulka pro výpočet, 459
výpočet, 457, 459
vzor, 458
Požadavky na zařízení, 4
Požadovaná hodnota
nastavení pro HSC, 125
změna v HSC, 131
PPI/Freeport mód, rádiový modem, 232
PPI mód, rádiový modem, 232
PPI Multi – Master kabel, 4
konfigurace, 230
konfigurace pro Freeport, 231
přepínač volby přenosové rychlosti, 231
rádiový modem, 232
režim Freeport, 227
RS – 232, 227
s modemem, 231
s rádiovým modemem, 232
výběr, 210, 224
Pracovní režimy, CPU
protokol Freeport, 226
stavové bity, 466
změna, 41
Priorita
podprogramů přerušení, 157
přerušovacích událostí, 158

PROFIBUS, zařízení master a slave, 211
PROFIBUS – DP
konzistence dat, 428
modul (EM 277), 426
standardní komunikace, 425
vzorový program, 433
Program
analogové vstupy, 25
běh, 14
download, 14, 36
editace v režimu RUN, 244
funkce ladění, 244
chyby kompilace, 60
chyby provádění, 61
podprogramy, 54
sdílení dat s přerušovacími podprogramy, 155
sledování, 14
stav sledování, 246
stavová tabulka, 63
strukturování, 53
tvorba, 10
uložení, 13, 36 – 40
určený počet programových cyklů, 248
vytvoření pomocí STEP 7 – Micro/WIN, 55
základní prvky, 53
Program, vzorový, řízení polohovacího profilu, 302
Programovací kabel, 4
Programovací software, objednací čísla, 480
Programování
polohovacího modulu EM 253, 266
vysokorychlostního čítače, 119
Programování v době provádění programu, chybové kódy, 463
Programové editory, 55
funkční bloky (FBD), 55
kontaktní schéma (LAD), 55
konvence, 58
otevření, 11
volba, 55
výpis instrukcí (STL), 55
Programový cyklus, 24
časovače, 198
určený počet, 248
Prohlížení, chyby, 60
Proměnné
monitorování pomocí stavové tabulky, 247
sledování, 63
smyčky PID, 150
symbolická adresace, 62
Proporcionální člen, algoritmus PID, 147
Protokol, komunikační
definovaný uživatelem, 225
volba, 214
Protokol definovaný uživatelem, režim Freeport, 225
Protokol historie, události CPU, 48
Protokol Modbus RTU, 350
funkce podporované modemovým modulem, 309
mapování adres, 310
modemový modul EM 241, 309
Protokol Modbus slave
adresy, 349
doba provádění, 348
chybové kódy provádění, 353
inicializace, 348
instrukce, 350
konfigurování tabulky symbolů, 349
mapování adres pro S7 – 200, 349
MBUS_INIT, 351
MBUS_SLAVE, 353
podporované funkce, 350
použité prostředky, 348
příklad programování, 354
speciální paměť, 348
tabulka CRC, 348
Protokol Modbus slave, požadavky, 348
Protokol PPI, 214, 235
síť s jedním zařízením typu master, 216
síť s více zařízeními typu master, 216
složitá síť, 217
Protokol PROFIBUS, 215, 235
Protokoly
podporované STEP 7 – Micro/WIN, 225
PROFIBUS – DP, 425
Provádění
instrukce, 25
řídicí logiky, 24
Provozní režim, profil pohybu, 257
Průvodce
AS – I, 448
Ethernetem, 444
HSC, 119
instrukce, HSC, 119
instrukce, NETR, NETW, 81
Internetem, 447
modemem, 230, 314
NETR, NETW, 81
PID, 145
polohováním, 267
TD 200, 5
Průvodce AS – I, 448
Průvodce polohováním, 267
PTOx_ADV, 264
PTOx_CTRL, 260
PTOx_LDPOS, 263
PTOx_MAN, 262
PTOx_RUN, 261
Průvodce pro modem, modemový modul EM 241, 314
Průvodce datovým záznamem
download projektu, 368
podprogram DATx_WRITE, 369
použití, 365
průzkumník S7 – 200, 368
přehled, 364
přidělování paměti, 367
příklad použití, 364
S7 – 200 CPU, které podporují, 364
terminologie, 364
Průvodce recepturami, 357
download, 360
instrukce, 361
přidělování paměti, 359
termíny, 357
Průzkumník S7 – 200, 36, 41
průvodce datovým záznamem, 368
Přehled řízení otevřené smyčky, 252

Přehození bytů, instrukce, 183
příklad, 183
Přechod sekvenčního řídícího relé, instrukce, 172
Překrývané segmenty stínítka, TD 200 C, objednací číslo, 481
Přenos, instrukce, 86
příklad, 93
režim Freeport, 86
vysílání dat, 87
Přenos numerických zpráv, modemový modul EM 241, 310
Přenos textových zpráv na pager, modemový modul EM 241, 310
Přenos zpráv, modemový modul, 310
Přenosová rychlost, síť, 221
Přenosy dat, modemový modul EM 241, 311
Přepěťová ochrana, 22
Přepínač režimu, 41
Přepínače adres, EM 277 PROFIBUS – DP, 424
Přeplnění fronty (SMB4), 467
Přerušení
přiřazení HSC, 126
vysokorychlostní čítače, 120
Přerušení od času, 156
Přerušení od vstupů/výstupů, 156
Přerušovací události
priorita, 158
stručný přehled, 490
typy, 154
Přesunout, instrukce, příklad, 183
Přesunout blok, instrukce, 166
Přesunout blok double word, instrukce, 166
Přesunout blok instrukce, příklad, 166
Přesunout blok word, instrukce, 166
Přesunout byte, instrukce, 164
Přesunout byte pro čtení okamžitě, instrukce, 165
Přesunout byte pro zápis okamžitě, instrukce, 165
Přesunout double word, instrukce, 164
Přesunout real, instrukce, 164
Přesunout word, instrukce, 164
Přidat do tabulky, instrukce, 189
příklad, 189
Přidělování paměti, průvodce datovým záznamem, 367
Příjem, instrukce, 86
časovač měřící čas mezi znaky, 91
časovač zpráv, 91
chyby parity, 92
maximální počet přijatých znaků, 92
podmínky ukončení, 89
podmínky začátku zprávy, 89
příjem dat, 87
příklad, 93
režim Freeport, 86
rozpoznání pauzy (break), 90
rozpoznání prázdného úseku, 89
rozpoznání startovacího znaku, 89
rozpoznání ukončovacího znaku, 91
SMB86 až SMB94, SMB186 až SMB194, 474
ukončení uživatelem, 92
Příjem dat, 92
Příkazový byte, polohovací modul EM 253, 299
Příklady
časovače SIMATIC, 198, 199, 200
čítače SIMATIC, 115
IEC časovače, 201
IEC instrukce čítače, 116
instrukce ”AND”, 163
instrukce ”ASCII na hex”, 106
instrukce čtení a zápisu v síti, 84
instrukce dekódování, 112
instrukce dekrementace, 144
instrukce DIV, 142
instrukce FIFO, 190
instrukce inkrementace, 144
instrukce ”Integer na ASCII”, 106
instrukce inverze, 161
instrukce kontaktů, 72
instrukce LIFO, 191
instrukce logického zásobníku, 76
instrukce matematiky celých čísel, 141
instrukce matematiky reálných čísel, 141
instrukce MUL, 142
instrukce naplnění paměti, 192
instrukce nastavení, 73
instrukce návratu z podprogramu, 207
instrukce podmíněného ukončení programu, 168
instrukce podprogramu, 207
instrukce porovnání, 96
instrukce posunu, 180
instrukce posunutí bitů posuvného registru, 182
instrukce pro standardní konverzi, 101
instrukce prohledávání tabulky, 195
instrukce přerušení, 159
instrukce přesunu, 183
instrukce přesunu bloku dat, 166
instrukce přidávání do tabulky, 189
instrukce příjmu, 93
instrukce ”Real na ASCII”, 106
instrukce reset watchdogu, 168
instrukce rotace, 180
instrukce segmentu, 102
instrukce sekvenčního řídícího relé, 172
instrukce sestupného čítače, 115
instrukce skoku na návěští, 171
instrukce smyčky for – next, 170
instrukce stop, 168
instrukce výměny, 183
instrukce vynulování, 73
instrukce vysílání, 93
instrukce vysokorychlostního čítače, 132
instrukce vzestupného/sestupného čítače, 115
instrukce XOR, 163
instrukce zakódování, 112
instrukce zapnutí remanentního zpoždění, 200
instrukce zpoždění vypnutí, 199
instrukce zpoždění zapnutí, 198
konvergence toků řízení, 176
modemový modul EM 241, 323
OR, instrukce, 163
podmíněné přechody, 177
podprogram, 53
podprogram časovaného přerušení, 159
podprogramy přerušení, 53
polohovací modul, 285 – 289
program protokolu USS, 341

protokol Modbus slave, programování, 354
režim Freeport, 226
režimy vysokorychlostních čítačů, 121
rozdělení toků řízení, 175
síť s oběhem tokenu, 234
volání podprogramu, 205
výpočet požadavků na napájení, 457
Přímé smyčky, 150
Připojení
k S7–200, 10
modemu k S7–200, 231
Multi–Master kabelu, 9
pohonu MM3, 342
pohonu MM4, 345
rádiového modemu k S7–200, 232
síťových zařízení, 235
Připojit přerušení, instrukce, 153
Přirozený exponent, instrukce, 143
Přirozený logaritmus, instrukce, 143
Přiřazení
adresy, 62
počátečních hodnot, 62
přerušení HSC, 126
síťových adres, 211
Přístup k word, 26
PTO_CTRL, 260
PTOx_CTRL, 260
PTOx_LDPOS, 263
PTOx_MAN, 262
PTOx_RUN, 261
Pulzní časovač, instrukce (TP), 201
Pulzní šířková modulace, 253
konfigurace výstupu PWM, 253
Pulzní šířková modulace, instrukce (PWM), 50, 133
čas cyklu, 135
hodnoty tabulky profilu, 138
konfigurace pamětí SM, 136
metody aktualizace, 136
Position Control Wizard, 133
řízení krokového motoru, 138
vysvětlení, 135
Pulzní výstup, instrukce (PLS), 133
Pulzní výstup, instrukce (PTO), 50, 133
čas cyklu, 134
hodnoty tabulky profilu, 138
konfigurace pamětí SM, 136
Position Control Wizard, 133
řetězení jednoho segmentu, 134
řetězení více segmentů, 135
řízení krokového motoru, 138
vysvětlení, 134
Pulzní výstupy
instrukce pulzně šířkové modulace (PWM), 133
instrukce pulzního výstupu (PLS), 133
instrukce výstupu série pulzů (PTO), 133
vysokorychlostní, 50
PV mimo rozsah, automatické ladění PID, 377
PWMx_RUN, 254
instrukce pulzní šířkové modulace, 254

R

Rádiový modem
PPI mód, 232
PPI/Freeport mód, 232
s PPI Multi–Master kabelem, 232
Real na ASCII, instrukce, 105
příklad, 106
Receptury, upload a download, 36
Registr HSC0, HSC1, HSC2 (SMB36 až SMB65), 471
Registr HSC3, HSC4, HSC5 (SMB131 až SMB165), 475
Registr obrazu vstupů, 25
Registr obrazu vstupů (I), 27
Registr obrazu vstupů a výstupů, 42
Registr obrazu výstupů, 24
Registr obrazu výstupů (Q), 27
Relé, 22
životnost, 383
Remanentní bitová paměť, 38
Remanentní paměť, 44
Remanentní zpožděné zapnutí, instrukce (TONR), 196
příklad, 200
Repeatery
objednací čísla, 480
síť, 222
Reset, instrukce, 73
příklad, 73
Reset okamžitý, instrukce, 73
Restartování, po kritické chybě, 61
Reverzní smyčky, 150
Režim Freeport
definování, 156
chyba parity (SMB3), 467
instrukce vysílání a příjmu, 86
povolení, 86
protokoly definované uživatelem, 225
přechod z PPI, 87
příjem znaku (SMB2), 467
příklad, 226
RS–232, 227
řízení znakovým přerušením, 92
Režim RUN, 14, 41
editace programu, 244
Režim STOP, 14, 41
Režim vyhledání RP, volby, 303–307
polohovací modul EM 253, 303–307
Režim výměny dat, EM 277 PROFIBUS–DP, 428
Režimy
smyčky PID, 151
vysokorychlostní čítače, 120

Režimy vyhledávání RP, 303
RJ11 konektor, modemový modul EM 241, 308, 436
Rotovat vlevo byte, instrukce, 179
Rotovat vlevo double word, instrukce, 179
Rotovat vlevo word, instrukce, 179
Rotovat vpravo byte, instrukce, 179
Rotovat vpravo double word, instrukce, 179
Rotovat vpravo word, instrukce, 179
Rozhraní, výběr komunikačních, 210
Rozlišení, časovač, 197, 198
Rozlišení časovače, 197, 198
Rozměry, CPU, 2
Rozmístění vývodů, komunikační port, 223
Rozpoznání pauzy (break), 90
Rozpoznání prázdného úseku, 89
Rozpoznání startovacího znaku, 89
Rozpoznání ukončovacího znaku, 91
Rozsah bytů a celých čísel, 26
Rozsahy, smyčka PID, 150
Rozsahy operandů, 69
Rozsahy paměti
  CPU moduly, 68
  stručný přehled, 491
Rozsahy teploty
  EM 231 RTD, 421 – 422
  EM 231 termočlánkový, 416 – 417
Rozšiřovací moduly, 3
  adresování vstupů a výstupů, 32
  analogové
    objednací čísla, 400, 411
    schémata zapojení, 402
    specifikace, 400
    specifikace vstupů, 400
    specifikace výstupů, 401
  demontáž, 19
  digitální, objednací čísla, 394
  diskrétní
    obecná specifikace, 394
    schémata zapojení, 397 – 399
    specifikace vstupů, 395
    specifikace výstupů, 396
  ID a chybový registr, 469
  instalace, 18
  objednací čísla, 479
  požadavky na napájení, 17, 457
  rozměry, 18
Rozšiřovací vstupy/výstupy, 32
  chyby sběrnice (SMW98), 475
RS klopný obvod, instrukce, 77
RS – 232/PPI Multi – Master kabel
  konfigurování pro HyperTerminal, 239 – 483
  nastavení DIP přepínače, 454
  nastavení portu, 220
  specifikace, 452
RS – 485, 222
RTD modul (EM 231), 418
RTD modul EM 231
  kompatibilita s CPU, 412
  konfigurace, 418
  rozsahy teploty a přesnost, 421 – 422
  specifikace, 411
  stavové kontrolky, 420
  svorkovnice, 412
  volba DIP přepínačů, 418 – 419
Rychlost modulace
  nastavení, 211
  optimální, 233
  síť, 221
  volby přepínače
    Multi – Master kabel, 9
    PPI Multi – Master kabel, 213, 225, 227, 231, 232

Ř

Řetězce, formát, 31
Řetězcová konstanta, 31
Řetězec, reprezentace, 32
Řídicí byte, nastavení pro HSC, 124
Řídicí logika, 24
Řídicí registry pro Freeport (SMB30 a SMB130), 470
Řízení
  chyby, 60
  složité komunikace, 236
Řízení krokového motoru, generátory PTO/PWM, 138
Řízení otevřené smyčky, 252
Řízení polohy otevřené smyčky, použití krokových motorů a servomotorů, 255
Řízení přerušení a příjmu znaku, 92
Řízení smyčky(PID) instrukce, 145 – 156
  chyby, 151
  konverze vstupů, 148
  konverze výstupů, 149
  přímá/reverzní, 150
  režimy, 151
  rozsahy/proměnné, 150
  upravení integrační sumy, 150
  volba typu, 148
Řízení větvení
  instrukce sekvenčního řídicího relé, 175
  příklad, 175

S

S7 – 200
  adresace, 26
  akumulátory, 29
  analogové vstupy (AI), 30
  analogové výstupy (AQ), 31
  CPU moduly, 2
  čtení a zápis dat, 24
  download, 36
  elektromagnetická kompatibilita, 383
  funkce zachycování pulzů, 46
  hardware, odstraňování poruch, 249
  chybové kódy, 462
  instalace, 17
  jako zařízení typu slave, 211, 425
  komponenty systému, 2
  konstantní hodnoty, 32
  konvence programování, 59
  modem, 231
  napájení, 8
  návod pro kabeláž, 21

návod pro montáž, 16
návod pro uzemnění, 21
oblast bitové paměti (M), 27
oblast lokální paměti (L), 30
oblast paměti časovače (T), 28
oblast paměti čítače (C), 28
oblast paměti sekvenčních řídicích relé (S), 31
oblast proměnné paměti (V), 27
oblast speciální paměti (SM), 30
odezva na kritickou chybu, 462
odstraňování poruch, 249
ochrana heslem, 48
ošetření chyb, 60
paměť, 26
paměť C, 28
paměť I, 27
paměť L, 30
paměť M, 27
paměť Q, 27
paměť S, 31
paměť SM, 30
paměť T, 28
paměť V, 27
podmínky prostředí, 383
podporované přerušovací podprogramy, 156
podprogramy přerušení, 155
programový cyklus, 24, 42
provádění řídicí logiky, 24
přenosové rychlosti, 211, 212–214
připojení STEP 7–Micro/WIN, 10
přístup k datům, 26
rádiový modem, 232
registr obrazu vstupů (I), 27
registr obrazu vstupů a výstupů, 42
registr obrazu výstupů (Q), 27
remanentní paměť, 44
režim RUN, 14, 41
režim STOP, 14, 41
rozměry, 2
rozsahy paměti, 68
rozšiřovací moduly, 3
RS–232, 227
schémata zapojení, 390–392
síťová adresa, 212–214
speciální možnosti, 42
stavová tabulka, 247
technická specifikace, 383
ukládání dat, 36
upload, 36
vysokorychlostní čítače, 29
vzory konfigurace sítě, 216, 217, 218, 219
S7–200 systémový manuál, objednací čísla, 480
S7–300, vzory konfigurace sítě, 217
S7–400, vzory konfigurace sítě, 217
Sečíst, instrukce, 140
Sedmisegmentový displej, 102
Sedmisegmentový displej, instrukce, 102
příklad, 102
Servomotory, řízení polohy otevřené smyčky, 255
Set, instrukce, 73
příklad, 73
sestupná hrana, instrukce, 70, 71
Schémata, tvorba schémat konfigurace, 53
Schémata konfigurace, 53
Schémata zapojení
analogové rozšiřovací moduly, 402
digitální rozšiřovací moduly, 397–399
moduly CPU, 390–392
polohovací modul EM 253, 441–442
vstupy a výstupy CPU, 390
Schválení, 382
Signálový tok, parametr podprogramu, 205
SIMATIC instrukce časovače, 196
příklad, 198, 199, 200
SIMATIC instrukce čítače
příklady, 115
sestupný, 113
vzestupný, 113
vzestupný/sestupný, 114
SIMATIC instrukční soubory, 57
Simulátory, objednací čísla, 481
Simulátory vstupu, 456
Sinus, instrukce, 143
Síť
adresa nejvyšší stanice (HSA), 233
adresy, 211
adresy zařízení, 214
CP karty, 224
doba oběhu tokenu, 233
faktor obnovení mezery (GUF), 233
kabel, 221, 222
komponenty, 221–224
komunikační port, 223
modem, 231
MPI, méně než 187,5 kbaud, 217
MPI, více než 187,5 kbaud, 218
optimalizace výkonu, 233
PPI Multi–Master kabel, 224
PPI s jedním zařízením typu master, 216
PPI s více zařízeními typu master, 216
PROFIBUS, 211
PROFIBUS–DP, 218
napěťové oddělení, 221
nastavení komunikace, 210–433
pokyny pro konfigurování, 221
přenosová rychlost, 221
přenosové rychlosti, 221
rádiový modem, 232
repeatery, 222
složitá, 236
složitá PPI, 217
symetrizace kabelu, 223
ukončení kabelu, 223
vyhledávání CPU, 213
výpočet vzdáleností, 221
vytvoření, 221
vzory konfigurace, 216, 217, 218
zařízení HMI, 225
zařízení typu master, 211
zařízení typu slave, 211
Síť MPI
méně než 187,5 kbaud, 217
více než 187,5 kbaud, 218
Síť PPI s jedním zařízením typu master, 216
Síť PPI s více zařízeními typu master, 216
Síť PROFIBUS–DP

repeatery, 222
rozmístění vývodů, 223
S7 – 315 – 2 a EM 277, 218
specifikace kabelu, 221
STEP 7 – Micro/WIN a HMI, 218
Síť s oběhem tokenu, příklad, 234
Skokové instrukce
návěští, 171
příklad, 171
skok na návěští, 171
Sledování, 14
procesních proměnných, 63
proměnných se stavovou tabulkou, 247
stavu programu, 246
Složitá síť PPI, 217
Služba textových zpráv, modemový modul EM 241, 310
SMB0: stavové bity, 466
SMB1: stavové bity, 466
SMB2: znak přijatý v režimu Freeport, 467
SMB3: chyba parity v režimu Freeport, 467
SMB4: přeplnění fronty, 467
SMB5: stav vstupů a výstupů, 468
SMB6: ID registr CPU, 468
SMB7: rezervován, 468
SMB8 až SMB21: identifikační a chybové registry vstup/výstupního modulu, 469
SMB28, SMB29 analogové nastavení, 50, 470
SMB30 a SMB130: řídicí registry pro Freeport, 470
SMB31 a SMW32: řízení zápisu do EEPROM, 471
SMB34 a SMB35: registry časovaného přerušení, 471
SMB36 až SMB65: registr HSC0, HSC1, HSC2, 471
SMB66 až SMB85: registry PTO/PWM, 473
SMB86 až SMB94, SMB186 až SMB194: řízení příjmu zpráv, 474
SMB130: řídicí registry pro Freeport, 470
SMB131 až SMB165: registr HSC3, HSC4, HSC5, 475
SMB166 až SMB185: tabulka definice profilu PTO0, PTO1, 476
SMB186 až SMB194: řízení příjmu zpráv, 474
SMB200 až SMB549: stav inteligentních modulů, 477
Směr, změna v HSC, 130
Směrnice
indukční zátěže, 22
instalace, 16
instrukce pro polohovací modul EM 253, 273
kabeláž, 20
konfigurace sítě, 221
napěťové oddělení, 20
navrhování systému mikro – PLC, 52
ochranné obvody, 22
podprogramy, 54
podprogramy přerušení, 54, 155
prostředí s vysokou vibrací, 18
uzemnění a kabeláž, 21
uzemnění a zapojení, 20
vertikální montáž, 18
změny ukazatelů pro nepřímé adresování, 34
žárovkové zátěže, 22
Směrnice pro instalaci stejnosměrného napětí, 21
Směrnice pro instalaci střídavého napětí, 21
SMS, modemový modul, 310
SMW22 až SMW26: doby programového cyklu, 470
SMW98: chyby na rozšiřovací vstup/výstupní sběrnici, 475
Soubor GSD, EM 277 PROFIBUS – DP, 431 – 432
Speciální paměť
modemový modul EM 241, 323
protokol Modbus slave, 348
Speciální paměťové bity, 466 – 474
stručný přehled, 489
Speciální paměťové byty, EM 277 PROFIBUS – DP, 429
Specifikace napájení, moduly CPU, 387
Specifikace vstupů
analogového rozšiřovacího modulu, 400
CPU modulů, 387
diskrétního rozšiřovacího modulu, 395
Specifikace výstupů
analogového rozšiřovacího modulu, 401
CPU modulů, 388
diskrétního rozšiřovacího modulu, 396
Spoje
CP 243 – 2, 449
ethernetového modulu CP 243 – 1, 444
internetového modulu CP 243 – 1 IT, 447
Spojit řetězce, instrukce, 184
Spuštění, STEP 7 – Micro/WIN, 9
SR klopný obvod, instrukce, 77
Standard RS – 232
PPI Multi – Master kabel, 227
režim Freeport, 227
Standardní DIN lišta, 17
Start, vysokorychlostní čítač, 123
Státní normy, 382
Stav
konec programového cyklu, 246
provádění, 246
sledování programu, 246
zobrazení v LAD a FBD, 246
zobrazení v STL, 247
Stav při běhu programu, 43
Stav vstupů a výstupů (SMB5), 468
Stavová tabulka, 63
vnucení hodnot, 248
sledování hodnot, 247
Stavové LED
EM 231 RTD, 420
EM 231 termočlánkový, 415
EM 277 PROFIBUS – DP, 424, 430
polohovací modul EM 253, 439
Stavový byte, vysokorychlostní čítač, 126
Stejnosměrné a reléové výstupy, 22
STEP 7 – Micro/WIN
CP karty, 225
instalace, 4
instrukční soubory
IEC 1131 – 3, 57
SIMATIC, 57
volba, 57
jako zařízení typu master, 211
nastavení komunikace, 10
nástroje pro ladění, 243
navázání komunikace s S7 – 200, 10
objednací čísla, 480

otevření, 55
požadavky na počítač, 4
požadavky na zařízení, 4
PPI Multi–Master kabel, 225
programovací balík, 4
programové editory, 55
přenosové rychlosti, 212–214
rozhraní pro EM 241, 309
síťová adresa, 212–214
spuštění, 9
tvorba programů, 55
vzory konfigurace sítě, 216–219
Stop, instrukce, 167
příklad, 168
Stručný přehled informací, 489
Strukturování programu, 53
Střídavé a reléové výstupy, 22
Svorkovnice
CPU 221 AC/DC/relé, 390
CPU 221 DC/DC/DC, 390
CPU 222 AC/DC/relé, 391, 392
CPU 222 DC/DC/DC, 391, 392
CPU 224 AC/DC/relé, 391, 392
CPU 224 DC/DC/DC, 391, 392
CPU 226 AC/DC/relé, 393
CPU 226 DC/DC/DC, 393
demontáž, 19
EM 221 DI 8 x 24 V DC, 398
EM 221 DI 8 x AC, 398
EM 222 DO 8 x 24 V DC, 398
EM 222 DO 8 x relé, 398
EM 223 4 x 24 V DC vstup/4 x 24 V DC výstup, 397
EM 223 DI 16/DO 16 x 24 V DC relé, 399
EM 223 DI 16/DO 16 x DC 24 V, 399
EM 223 DI 4/DO 4 x DC 24 V/relé, 397, 398
EM 223 DI 8/DO 8 x 24 V DC, 399
EM 223 DI 8/DO 8 x 24 V DC/relé, 399
EM 231 AI 4 x 12 bitů, 402
EM 231 RTD, 412
EM 231 termočlánkový, 412
EM 232 AQ 2 x 12 bitů, 402
EM 235 AI 4/AQ 1 x 12 bitů, 402
opakovaná montáž, 19
Symbolická adresace, 62
Symbolické názvy, tvorba seznamu, 53
Symetrizace
síťový kabel, 223
smyčka PID, 145, 147
Synchronní aktualizace, instrukce PWM, 136
Systémová podpora, pro podprogramy přerušení, 155
Systémový blok, 54

T

Tabulka CRC, protokol Modbus slave, 348
Tabulka definování profilu PTO0, PTO1 (SMB166 až SMB185), 476
Tabulka globálních proměnných, 62
Tabulka konfigurace, modemový modul EM 241, 313, 325
Tabulka konfigurace/profilů, polohovací modul EM 253, 294
Tabulka křížových odkazů, 244
Tabulka lokální proměnné, 55, 63
Tabulka profilů, polohovací modul EM 253, 294
Tabulka smyčky, 152
PID, 372
Tabulka symbolů, 62
adresace, 62
konfigurace pro Modbus, 349
Tangens, instrukce, 143
TC/IP, komunikační protokol, 215
TC/IP, protokol, 215
TD 200, průvodce, 5
Telefonní linka mezinárodní, modemový modul EM 241, 308
Teleservis, 309
Termočlánkový modul (EM 231)
konfigurace, 413
rozsahy teploty, 416–417
stavové kontrolky, 415
základní informace, 413
Termočlánkový modul EM 231
kompatibilita s CPU, 412
konfigurace, 413
rozsahy teploty a přesnost, 416–417
specifikace, 411
stavové kontrolky, 415
svorkovnice, 412
volba DIP přepínačů, 414
základní informace, 413
Textová zobrazovací jednotka TD 200C, objednací číslo, 481
Textové displeje
objednací číslo, 481
překrývané segmenty stínítka (TD 200C), 481
Textový displej TD 200, 5
Textový displej TD 200
objednací číslo, 481
objednací číslo manuálu, 480
TP–Designer pro TP070, verze 1.0, 480
Tranzistor, ochrana, 22
Trvalé uložení programu, 39
Typy dat, parametry podprogramu, 205

U

Údaj čas, protokol historie, 48
Události CPU, protokol historie, 48
Uchování paměti, 36–39
rozsahy, 44
Ukazatele, nepřímá adresace, 33
Ukončení
modulu CPU, 19
rozšiřovacího modulu, 19
svorkovnice, 19
Ukončení, síťového kabelu, 223
Ukončení uživatelem, 92
Uložení
bitové paměti (M) do EEPROM, 38
hodnoty do EEPROM, 471
programových dat S7–200, 36

programu, 13
programu v paměťovém modulu, 37
proměnné paměti (V) do EEPROM, 39
Umístění pracovního pásma, polohovací modul EM 253, 306
Upload, programu, 36
USB/PPI Multi – Master kabel
nastavení portu, 220
objednací číslo, 452
volba, 210
USS protokol, požadavky, 332
USS4_DRV_CTRL, instrukce, 335
USS4_INIT, instrukce, 334
USS4_RPM_x, instrukce, 338, 339
USS4_WPM_x, instrukce, 338, 339
Uzemnění, 20, 21
Uživatelské knihovny, 64

V

Vlastnosti
CPU moduly, 68
modemový modul EM 241, 308
polohovací modul EM 253, 265
Vložené proměnné v textových zprávách a zprávách SMS, 311
Vložení, instrukce, 12
Vnoření, podprogramy, 204
Vnucení hodnot (force), 247, 248
Volání podprogramu, instrukce, 204
příklad, 205
Volba
CP karty, 224
DIP přepínače termočlánku, 414
instrukčních souborů, 57
komunikačního protokolu, 214
PPI Multi – Master kabel, 224
pracovního režimu S7 – 200, 41
programového editoru, 55
RTD DIP přepínače, 418 – 419
Vstupy, 24, 25
diskrétního rozšiřovacího modulu, 395
filtrace, analogová, 45
filtrace, digitální, 45
kalibrace, 403
modulu CPU, 390
podmíněné/nepodmíněné, 59
polohovacího modulu EM 253, 265
vysokorychlostních čítačů, 120
start a nulování (HSC), 123
Vstupy a výstupy, adresace, 32
Vstupy a výstupy, čtení a zapisování, 42
Vstupy a výstupy, chyby, 60
Vstupy smyčky
konverze, 148
normalizace, 148
Vůle, nastavení pracovního pásma, 306
Výjimečné podmínky, automatické ladění PID, 377
Výkon, optimalizace výkonu sítě, 233
Výměna
aktuální hodnoty HSC, 131
nastavené hodnoty HSC, 131
směru vysokorychlostního čítače, 130
Vynulování vysokorychlostního čítače, 123
Vyplnění, instrukce, 192
Výpis instrukcí. *Viz* Editor STL
Výpočet
doby oběhu tokenu, 233
požadavků na napájení, 457 – 459
Vysokonapěťová izolační zkouška, 383
Vysokorychlostní čítače, 50
adresace, 126
časové diagramy, 121 – 123
definování režimů a vstupů, 120
funkce s ”Reset” a ”Start”, 123
inicializační sekvence, 127
naprogramování aktivního stavu, 124
nastavení aktuálních a požadovaných hodnot, 125
nastavení řídicího bytu, 124
paměťová oblast, adresace, 29
programování, 119
přerušení, 120
přiřazení přerušení, 126
režimy, 121, 492
řídicí byte, 119
SMB36 – SMB65, 471
stavový byte, 126
vysvětlení, 119
zablokování, 131
změna aktuální hodnoty, 131
změna nastavené hodnoty, 131
změna směru, 130
Vysokorychlostní pulzní výstup
činnost, 133
SMB36 – SMB65, 473
Vysokorychlostní pulzní výstupy, 50
Výstup, instrukce, 73
Výstup okamžitý, instrukce, 73
Výstup PTO, 258
Výstup pulzní šířkové modulace, PWMx_RUN, 254
Výstup PWM, konfigurování, 253
Výstup série impulzů
PTOx_ADV, 264
PTOx_CTRL, 260
PTOx_LDPOS, 263
PTOx_MAN, 262
PTOx_RUN, 261
Výstupy, 24
diskrétního rozšiřovacího modulu, 397
instrukce, bez, 59
modulu CPU, 390
polohovacího modulu EM 253, 265
Výstupy, analogové, konfigurování hodnot, 44
Výstupy, digitální, konfigurování stavů, 43
Výstupy a relé, 22
Výstupy smyčky, konverze na integer s upraveným měřítkem, 149
Vytvoření
programu, 10
programu s Micro/WIN, 55
protokolů definovaných uživatelem, 225
seznamu symbolických názvů, 53
schémat konfigurace, 53

Vytvoření sítě, 221
Vývody konektoru, rozmístění v komunikačním portu, 223
Vzdálená adresa, nastavení pro S7–200, 213
Vzorový program, 10

W

Watchdog reset, instrukce, 167
  příklad, 168

Z

Zabezpečení hesla, 48
Zabezpečení zpětným voláním, modemový modul EM 241, 312
Zabránění konfliktům v síti, 236
Zadání
  nové aktuální hodnoty HSC, 131
  nové nastavené hodnoty HSC, 131
Zachycení impulzu, 45
Zakázání přerušení, instrukce, 153
Zakódovat, instrukce, 112
  příklad, 112
Záložky, 244
Zaokrouhlit, instrukce, 101
Zápis do sítě, instrukce, 81, 82
  chybové kódy, 82
  příklad, 84
Zařízení
  analogového rozšiřovacího modulu, 400
  BUS–DP, 423
  diskrétního rozšiřovacího modulu, 394
  EM 231 RTD, 411
  EM 231 termočlánkový, 411
  EM 277 PROFI
  ethernetového modulu CP 243–1, 443
  internetového modulu CP 243–1 IT, 446
  mikro–PLC, 52RS–232/PPI Multi–Master kabel, 452
  modemu EM 241, 435
  modulu AS–Interface CP 243–2, 448
  modulu CPU, 385, 386
Zařízení HMI, 225
Zařízení typu master, 211
Zařízení typu slave, 211
  EM 277 PROFIBUS–DP, 425
  S7–200, 425
Zásobník, PROFIBUS, 428
Zásuvné moduly
  objednací čísla, 480
  paměť, 450
Zjistit adresy portu, instrukce, 95
Změny ukazatelů, 34
Zobrazení
  programových prvků, 55
  stavu programu, 246
Zobrazovací panely
  dotykový panel TP070, 5
  textový displej TD 200, 5
Zpětné volání, modemový modul EM 241, 312
Zpracování požadavků komunikace, 25
Zprávy, síť s oběhem tokenu, 234
Zřetězení, impulzy PTO, 134
Ztráta hesla, 49
Ztráta napájení, remanentní paměť, 38, 44
Zvýšení hodnoty ukazatelů, 34
Zvýšit paměť pro program, vypnout editaci v režimu RUN, 48

Ž

Žádná operace, instrukce, 73
Žárovkové zátěže, směrnice, 22

do

SIEMENS ENERGY & AUTOMATION INC
ATTN: TECHNICAL COMMUNICATIONS
ONE INTERNET PLAZA
PO BOX 4991
JOHNSON CITY TN USA 37602–4991

Od

Jméno: _ _ _

Formulář pro připomínky

Vaše poznámky a doporučení nám pomohou zlepšit kvalitu a užitečnost našich publikací. Využijte prosím první příležitosti, vyplňte tento dotazník a odešlete ho zpět společnosti Siemens.

Ohodnoťte prosím každou následující otázku podle svého názoru známkou od 1 (výborně) až do 5 (nedostatečně).

1. Splňuje obsah vaše požadavky? ☐
2. Našli jste snadno potřebné informace? ☐
3. Je text snadno pochopitelný? ☐
4. Odpovídá úroveň technických podrobností vašim požadavkům? ☐
5. Ohodnoťte prosím kvalitu grafiky a tabulek. ☐

Další poznámky:

Rozsahy paměti a vybavení S7-200

Popis	CPU 221	CPU 222	CPU 224	CPU 224XP	CPU 226
Velikost uživatelského programu s editací v režimu RUN bez editace v režimu RUN	4096 bytů 4096 bytů	4096 bytů 4096 bytů	8192 bytů 12288 bytů	12288 bytů 16384 bytů	16384 bytů 24576 bytů
Velikost uživatelských dat	2048 bytů	2048 bytů	8192 bytů	10240 bytů	10240 bytů
Registr - obrazu vstupů	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7	I0.0 až I15.7
Registr - obrazu výstupů	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7	Q0.0 až Q15.7
Analogové vstupy (pouze čtení)	AIW0 až AIW30	AIW0 až AIW30	AIW0 až AIW62	AIW0 až AIW62	AIW0 až AIW62
Analogové výstupy (pouze zápis)	AQW0 až AQW30	AQW0 až AQW30	AQW0 až AQW62	AQW0 až AQW62	AQW0 až AQW62
Paměť pro proměnné (V)	VB0 až VB2047	VB0 až VB2047	VB0 až VB8191	VB0 až VB10239	VB0 až VB10239
Lokální paměť (L)[1]	LB0 až LB63	LB0 až LB63	LB0 až LB63	LB0 až LB63	LB0 až LB63
Bitová paměť (M)	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7	M0.0 až M31.7
Speciální paměť (SM) Pouze čtení	SM0.0 až SM179.7 SM0.0 až SM29.7	SM0.0 až SM299.7 SM0.0 až SM29.7	SM0.0 až SM549.7 SM0.0 až SM29.7	SM0.0 až SM549.7 SM0.0 až SM29.7	SM0.0 až SM549.7 SM0.0 až SM29.7
Časovače	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)	256 (T0 až T255)
Remanentní zpožděné zapnutí- 1 ms	T0, T64	T0, T64	T0, T64	T0, T64	T0, T64
10 ms	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68	T1 až T4, a T65 až T68
100 ms	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95	T5 až T31, a T69 až T95
Zpožděné zapnutí/vypnutí 1 ms	T32, T96	T32, T96	T32, T96	T32, T96	T32, T96
10 ms	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100	T33 až T36, a T97 až T100
100 ms	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255	T37 až T63, a T101 až T255
Čítače	C0 až C255	C0 až C255	C0 až C255	C0 až C255	C0 až C255
Vysokorychlostní čítače-	HC0 až HC5	HC0 až HC5	HC0 až HC5	HC0 až HC5	HC0 až HC5
Sekvenční řídicí relé (S)	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7	S0.0 až S31.7
Registry akumulátoru	AC0 až AC3	AC0 až AC3	AC0 až AC3	AC0 až AC3	AC0 až AC3
Skoky/návěští	0 až 255	0 až 255	0 až 255	0 až 255	0 až 255
Volání/podprogramy	0 až 63	0 až 63	0 až 63	0 až 63	0 až 127
Přerušovací podprogramy	0 až 127	0 až 127	0 až 127	0 až 127	0 až 127
Náběžné/sestupné hrany	256	256	256	256	256
Smyčky PID	0 až 7	0 až 7	0 až 7	0 až 7	0 až 7
Porty	Port 0	Port 0	Port 0	Port 0, Port 1	Port 0, Port 1

[1] LB60 až LB63 jsou rezervovány pro STEP 7–Micro/WIN, verze 3.0 nebo pozdější.

STL	Strana
=	NO TAG
+D	NO TAG
–D	NO TAG
* D	NO TAG
/ D	NO TAG
+I	NO TAG
–I	NO TAG
=I	NO TAG
* I	NO TAG
/ I	NO TAG
+R	NO TAG
–R	NO TAG
*R	NO TAG
/R	NO TAG
A	NO TAG
AB < =	NO TAG
AB =	NO TAG
AB >	NO TAG
AB <	NO TAG
AB > =	NO TAG
AB <>	NO TAG
AD <	NO TAG
AD < =	NO TAG
AD =	NO TAG
AD >	NO TAG
AD > =	NO TAG
AD <>	NO TAG
AENO	NO TAG
AI	NO TAG
ALD	NO TAG
AN	NO TAG
ANDB	NO TAG
ANDD	NO TAG
ANDW	NO TAG
ANI	NO TAG
AR=	NO TAG
AR <	NO TAG
AR<=	NO TAG
AR >	NO TAG
AR>=	NO TAG
AR <>	NO TAG
AS=	NO TAG
AS<>	NO TAG
ATCH	NO TAG
ATH	NO TAG
ATT	NO TAG
AW <	NO TAG
AW < =	NO TAG
AW=	NO TAG
AW >	NO TAG

STL	Strana
AW > =	NO TAG
AW <>	NO TAG
BCDI	NO TAG
BIR	NO TAG
BITIM	NO TAG
BIW	NO TAG
BMB	NO TAG
BMD	NO TAG
BMW	NO TAG
BTI	NO TAG
CALL	NO TAG
CEVNT	NO TAG
CFND	187
CITIM	NO TAG
COS	NO TAG
CRET	NO TAG
CRETI	NO TAG
CSCRE	NO TAG
CTD	NO TAG
CTU	NO TAG
CTUD	NO TAG
DECB	NO TAG
DECD	NO TAG
DECO	NO TAG
DECW	NO TAG
DISI	NO TAG
DIV	NO TAG
DLED	NO TAG
DTA	NO TAG
DTCH	NO TAG
DTI	NO TAG
DTR	NO TAG
DTS	NO TAG
ED	NO TAG
ENCO	NO TAG
END	NO TAG
ENI	NO TAG
EU	NO TAG
EXP	NO TAG
FIFO	NO TAG
FILL	NO TAG
FND <	NO TAG
FND <>	NO TAG
FND =	NO TAG
FND >	NO TAG
FOR	NO TAG
GPA	NO TAG
HDEF	NO TAG
HSC	NO TAG
HTA	NO TAG

STL	Strana
IBCD	NO TAG
INCB	NO TAG
INCD	NO TAG
INCW	NO TAG
INVB	NO TAG
INVD	NO TAG
INVW	NO TAG
ITA	NO TAG
ITB	NO TAG
ITD	NO TAG
ITS	NO TAG
JMP	NO TAG
LBL	NO TAG
LD	NO TAG
LDB <=	NO TAG
LDB =	NO TAG
LDB >=	NO TAG
LDB >	NO TAG
LDB <	NO TAG
LDB <>	NO TAG
LDD >=	NO TAG
LDD <	NO TAG
LDD <=	NO TAG
LDD =	NO TAG
LDD >	NO TAG
LDD <>	NO TAG
LDI	NO TAG
LDN	NO TAG
LDNI	NO TAG
LDR=	NO TAG
LDR <	NO TAG
LDR<=	NO TAG
LDR >	NO TAG
LDR>=	NO TAG
LDR <>	NO TAG
LDS	NO TAG
LDS=	NO TAG
LDS<>	NO TAG
LDW <=	NO TAG
LDW <	NO TAG
LDW =	NO TAG
LDW >	NO TAG
LDW >=	NO TAG
LDW <>	NO TAG
LIFO	NO TAG
LN	NO TAG
LPP	NO TAG
LPS	NO TAG
LRD	NO TAG
LSCR	NO TAG

STL	Strana
MOVB	NO TAG
MOVD	NO TAG
MOVR	NO TAG
MOVW	NO TAG
MUL	NO TAG
NEXT	NO TAG
NETR	NO TAG
NETW	NO TAG
NOT	NO TAG
O	NO TAG
OB =	NO TAG
OB > =	NO TAG
OB >	NO TAG
OB <	NO TAG
OB < =	NO TAG
OB <>	NO TAG
OD <	NO TAG
OD < =	NO TAG
OD =	NO TAG
OD >	NO TAG
OD > =	NO TAG
OD <>	NO TAG
OI	NO TAG
OLD	NO TAG
ON	NO TAG
ONI	NO TAG
OR=	NO TAG
OR <	NO TAG
OR <=	NO TAG
OR >	NO TAG
OR >=	NO TAG
OR <>	NO TAG
ORB	NO TAG
ORD	NO TAG
ORW	NO TAG
OS=	NO TAG
OS<>	NO TAG
OW <	NO TAG
OW < =	NO TAG
OW =	NO TAG
OW >	NO TAG
OW > =	NO TAG
OW <>	NO TAG
PID	NO TAG
PLS	NO TAG
R	NO TAG
RCV	NO TAG
RI	NO TAG
RLB	NO TAG
RLD	NO TAG

STL	Strana
RLW	NO TAG
ROUND	NO TAG
RRB	NO TAG
RRD	NO TAG
RRW	NO TAG
RTA	NO TAG
RTS	NO TAG
S	NO TAG
SCAT	NO TAG
SCPY	NO TAG
SCRE	NO TAG
SCRT	NO TAG
SEG	NO TAG
SFND	187
SHRB	NO TAG
SI	NO TAG
SIN	NO TAG
SLB	NO TAG
SLD	NO TAG
SLEN	NO TAG
SLW	NO TAG
SPA	NO TAG
SQRT	NO TAG
SRB	NO TAG
SRD	NO TAG
SRW	NO TAG
SSCPY	186
STD	NO TAG
STI	NO TAG
STOP	NO TAG
STR	NO TAG
SWAP	NO TAG
TAN	NO TAG
TODR	NO TAG
TODRX	NO TAG
TODW	NO TAG
TODWX	NO TAG
TOF	NO TAG
TON	NO TAG
TONR	NO TAG
TRUNC	NO TAG
WDR	NO TAG
XMT	NO TAG
XORB	NO TAG
XORD	NO TAG
XORW	NO TAG