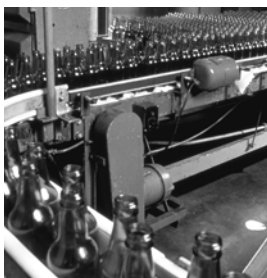


Procesory Logix5000: vstupní a výstupní data a data tagů



Programovací příručka

**Katalogová čísla 1756 ControlLogix,
1769 CompactLogix, 1789 SoftLogix,
1794 FlexLogix, PowerFlex 700S
s DriveLogix**

Důležité informace pro uživatele

Tranzistorové prvky mají provozní vlastnosti, které se liší od provozních vlastností elektromechanických prvků. Některé důležité rozdíly mezi tranzistorovými prvky a elektromechanickým vybavením popisuje dokument Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (Bezpečnostní předpisy pro použití, instalaci a údržbu tranzistorových řídicích prvků, publikace SGI-1.1, k dispozici v místní prodejní kanceláři společnosti Rockwell Automation nebo online na adrese <http://literature.rockwellautomation.com>). V důsledku tohoto rozdílu a také v důsledku široké škály aplikací, ve kterých jsou tranzistorové prvky používány, se musí všichni pracovníci, kteří tyto prvky používají, přesvědčit o tom, že takové použití těchto prvků je přijatelné.





Společnost Rockwell Automation, Inc. není v žádném případě odpovědná za nepřímé nebo následné škody vzniklé použitím takových prvků.

Příklady a diagramy v tomto manuálu jsou poskytovány pouze jako ilustrace. Protože každá konkrétní instalace obsahuje mnoho proměnných a rozdílných požadavků, společnost Rockwell Automation, Inc. nemůže být odpovědná za konkrétní použití těchto prvků na základě těchto příkladů a diagramů.

Společnost Rockwell Automation, Inc. nepředpokládá v souvislosti s používáním informací, obvodů, vybavení nebo softwaru popsanych v tomto dokumentu žádnou patentovou odpovědnost.

Reprodukce obsahu tohoto dokumentu, částečná ani úplná, není dovolena bez písemného souhlasu společnosti Rockwell Automation, Inc.

V tomto dokumentu používáme pro upozornění na bezpečnost následující poznámky.

VAROVÁNÍ 	Informace o postupech nebo okolnostech, které mohou v nebezpečném prostředí vyvolat explozi vedoucí ke zranění či smrti, škodě na majetku nebo ekonomické ztrátě.
DŮLEŽITÉ UPOZORNĚNÍ	Informace, které jsou zásadně důležité pro úspěšnou aplikaci a použití produktu.
POZOR 	Informace o postupech nebo okolnostech, které mohou vést ke zranění či smrti, škodě na majetku nebo ekonomické ztrátě. Tyto informace vám pomohou rozpoznat nebezpečí, vyhnout se mu a identifikovat následky.
NAPĚTÍ 	Štítky na strojích a vybavení nebo uvnitř nich, například štítky na pohonu nebo motoru – přítomnost nebezpečného napětí.
RIZIKO POŽÁRU 	Štítky na strojích a vybavení nebo uvnitř nich, například štítky na pohonu nebo motoru – teplota povrchu může dosáhnout nebezpečných hodnot.

Allen-Bradley, Rockwell Automation a TechConnect jsou registrované ochranné známky společnosti Rockwell Automation, Inc.

Ochranné známky jiných subjektů než společnosti Rockwell Automation jsou majetkem příslušných společností.

Předmluva	Účel tohoto dokumentu	5
	Jak používat tuto příručku	5
Komunikace s moduly I/O	Kapitola 1	
	Úvod	7
	RPI – Requested Packet Interval	8
	Komunikační formát	9
	Přímé spojení nebo spojení typu rack-optimized	9
	Vlastnictví	10
	Electronic Keying (Elektronická identifikace)	12
	Adresy dat I/O	13
	Kopírování dat I/O (Buffering I/O)	14
	Kopírování dat I/O	14
Organizace tagů	Kapitola 2	
	Úvod	17
	Typy tagů	18
	Datový typ	19
	Rozsah	21
	Použití tagů	22
	Vytváření tagů	25
	Vytvoření pole	26
	Vytvoření pole	29
	Vytvoření uživatelem definovaného datového typu	30
	Pravidla pro používání uživatelem definovaných datových typů ...	32
	Vytvoření uživatelem definovaného datového typu	32
	Popis uživatelem definovaného datového typu	34
	Zapnutí a vypnutí předaných a doplněných popisů	35
	Vložení předaného popisu	35
	Adresování dat tagů	36
	Přidělování Alias tagů	37
	Zobrazení informací o aliasu	38
	Přidělení aliasu	39
	Přidělení nepřímé adresy	40
	Výrazy	42
	Spodní index pole mimo rozsah	43
	Dokumentace tagů	43
	Dokumentace základních tagů	44
	Dokumentace alias tagů	44
	Dokumentace vytvořených tagů	44
	Dokumentace přijatých tagů	44
	Přepínání jazyků	44

**Pevné nastavení vstupní a
výstupní hodnoty uživatelem
(FORCE I/O)****Kapitola 3**

Úvod	47
Upozornění	47
Povolení nastavení pevných hodnot uživatelem (forces).....	48
Zákaz nebo odstranění pevného nastavení (force)	48
Kontrola stavu pevných nastavení hodnot (forces)	49
FORCE LED	50
Instrukce GSV	50
Kdy použít pevné nastavení hodnot I/O	51
Pevné nastavení vstupní hodnoty	51
Pevné nastavení výstupní hodnoty	52
Přidání pevného nastavení hodnot I/O	52
Odstranění nebo vypnutí pevných nastavení hodnot	53
Odstranění jednoho pevného nastavení	53
Zákaz všech I/O Forces	54
Odstranění všech I/O Forces	54
Podpora společnosti Rockwell Automation	58
Pomoc s instalací	58
Vracení nových produktů	58

Účel tohoto dokumentu

Tento dokument popisuje přístup ke vstupním a výstupním datům a datům tagů v procesorech Logix5000. Tento dokument je jedním z několika souvisejících dokumentů, které obsahují běžné postupy programování a provozu procesorů Logix5000. Seznam všech návodů k programování je k dispozici v dokumentu Logix 5000 Controllers Common Procedures Programming Manual, publikace č. 1756-PM001.

Termín procesor Logix5000 odkazuje na jakýkoli procesor založený na operačním systému Logix5000, jnapříklad:

- procesory CompactLogix,
- procesory ControlLogix,
- procesory DriveLogix,
- procesory FlexLogix,
- Procesory SoftLogix5800.

Jak používat tuto příručku

Některé části textu mají odlišné formátování.

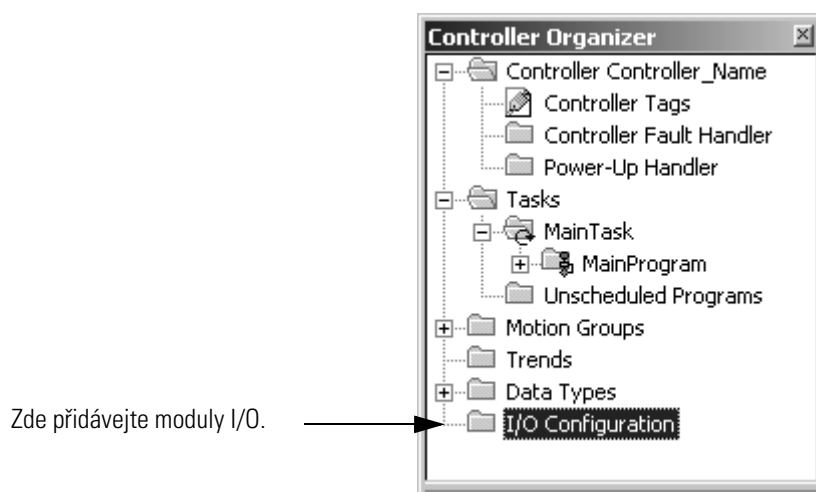
Formátování textu	Znamená	Například	Význam
<i>Proloženě</i>	konkrétní jméno položky, kterou uvidíte na obrazovce nebo v příkladu	Klepněte pravým tlačítkem myši na položku <i>User-Defined ...</i>	Klepněte pravým tlačítkem myši na položku User-Defined.
Písmo <i>Courier</i>	informace, které závisejí na konkrétní aplikaci (proměnné)	Klepněte pravým tlačítkem myši na <i>název_programu ...</i>	Je nutno identifikovat příslušný program v aplikaci. Většinou jde o název proměnné, kterou jste sami definovali.
v hranatých závorkách	klávesa na klávesnici	Stiskněte klávesu [Enter].	Stiskněte klávesu Enter.

Poznámky:

Komunikace s moduly I/O

Úvod

Pro komunikaci s modulem I/O v systému je nutné takový modul přidat do adresáře I/O Configuration procesoru.



Při přidání modulu proběhne i jeho konfigurace. Možnosti konfigurace závisí na druhu modulu, nicméně existují vlastnosti, které se většinou nastavují u všech modulů:

- RPI – Requested Packet Interval
- Komunikační formát
- Electronic Keying (Elektronická identifikace)

RPI – Requested Packet Interval

Procesor Logix5000 používá k přenosu a příjmu dat I/O různá spojení.

Termín	Definice
Spojení (Connection)	<p>Komunikační vazba mezi dvěma prvky, například mezi procesorem a modulem I/O, terminálem PanelView nebo jiným procesorem.</p> <p>Spojení představují alokaci zdrojů, které umožňují spolehlivější komunikaci mezi prvky než nepropojená hlášení. Počet spojení jednoho procesoru je omezen.</p> <p>Počet spojení, které procesor používá, nepřímo definujeme konfigurací komunikace procesoru s ostatními prvky v systému. U spojení se používají následující typy komunikace:</p> <ul style="list-style-type: none"> • moduly I/O, • vytvořené a přijaté tagy, • některé druhy MSG instrukcí (ne všechny).
Requested Packet Interval (Požadovaný interval paketů RPI)	<p>Hodnota RPI představuje periodu, ve které se přes spojení obnovují data. Například vstupní modul posílá data procesoru v intervalu nastaveném na tomto modulu.</p> <ul style="list-style-type: none"> • RPI se většinou nastavuje v milisekundách (ms) a v rozsahu 0,2 ms (200 mikrosekund) až 750 ms. • Pokud je k propojení prvků použita síť ControlNet, RPI rezervuje pozici v proudu dat v síti. Timing této pozice nemusí přesně odpovídat hodnotě RPI, nicméně řídicí systém zaručuje přesun dat alespoň v intervalu RPI.

U procesorů Logix5000 se hodnoty modulů I/O aktualizují v intervalu, který se nastavuje v projektovém adresáři konfigurace I/O. Hodnoty se aktualizují asynchronně k vykonávání logiky. Ve specifikovaném intervalu procesor hodnotu aktualizuje nezávisle na stavu vykonávání logiky.

POZOR



Po celou dobu vykonávání úlohy musí datová paměť obsahovat příslušné hodnoty. Na začátku každého skenu můžete data duplikovat nebo uložit.

- Programy v rámci úlohy přistupují ke vstupním a výstupním datům přímo z paměti omezené na procesor.
- Logika jakékoli úlohy může měnit data omezená na procesor.
- Data a hodnoty I/O jsou navzájem asynchronní a mohou se během vykonávání úlohy měnit.
- Vstupní hodnota, na kterou odkazuje začátek úlohy, může být jiná než hodnota na jejím konci.
- K prevenci změny vstupní hodnoty během skenu ji lze zkopírovat do jiného tagu a odkazovat na něj (tento proces se nazývá uložení hodnoty).

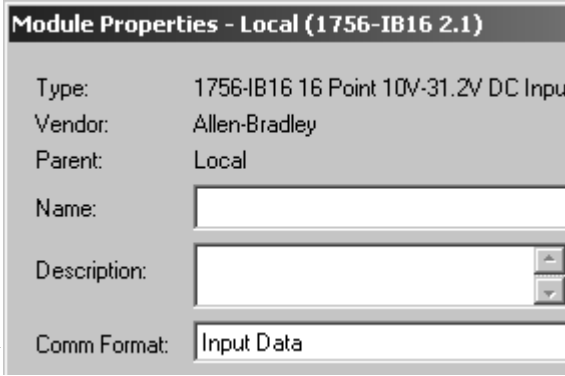
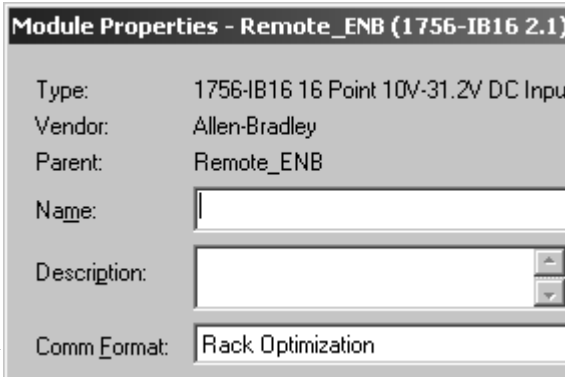
Komunikační formát

Komunikační formát rozhoduje o datové struktuře tagů spojených s modulem. Většina modulů I/O podporuje více formátů. Každý formát má jinou datovou strukturu. Výběr komunikačního formátu rozhoduje také o následujících hodnotách:

- Přímé spojení nebo spojení typu rack-optimized
- Vlastnictví

Přímé spojení nebo spojení typu rack-optimized

Procesory Logix5000 používají k přenosu a příjmu dat I/O různá spojení. Tato spojení mohou být přímá nebo typu Rack-Optimized.

Termín	Definice
Přímé spojení	<p>Přímé spojení je přenos dat mezi procesorem a modulem I/O v reálném čase. Procesor udržuje a monitoruje spojení s modulem I/O. Jakékoli přerušení spojení, například chyba modulu nebo jeho odstranění, sepne chybové bity v datové oblasti s ním spojené.</p> <p>Přímé spojení je jakékoli spojení, které nepoužívá formát Rack Optimization Come. →</p> 
Spojení typu rack-optimized	<p>Pro digitální moduly I/O se obvykle používá spojení typu rack-optimized spojení. Spojení typu rack-optimized konsoliduje spojení mezi řídicím systémem a všemi digitálními moduly I/O v rámu (nebo na liště DIN). Místo individuálních přímých spojení s každým modulem I/O je navázáno jedno spojení pro celý rám (nebo lištu DIN).</p> <p>Spojení typu rack-optimized →</p> 

Vlastnictví

V systému Logix5000 generují moduly data typu multicast. To znamená, že více prvků přijímá ve stejný okamžik stejná data od jednoho prvku.

Při výběru komunikačního formátu si musíte vybrat, zda chcete navázat s modulem spojení typu owner nebo typu listen-only.

Procesor typu owner

Procesor, který je odpovědný za primární konfiguraci a komunikaci s modulem. Procesor typu owner vytváří konfigurační data a může vytvářet spojení s modulem.

Spojení typu owner je jakékoli spojení, které ve svém komunikačním formátu nezahrnuje funkci listen-only.

Module Properties - Local (1756-IB16 2.1)

Type: 1756-IB16 16 Point 10V-31.2V DC Input

Vendor: Allen-Bradley

Parent: Local

Name:

Description:

Comm Format: Input Data

Spojení typu listen-only

Spojení I/O, u kterého konfigurační data pro modul I/O vlastní/poskytuje jiný procesor. Procesor používající spojení typu listen-only monitoruje modul. Nevytváří konfigurační data a udržuje spojení s modulem I/O, pouze pokud procesor typu owner modul I/O aktivně řídí.

Spojení typu listen-only

Module Properties - Local (1756-IB16 2.1)

Type: 1756-IB16 16 Point 10V-31.2V DC Input

Vendor: Allen-Bradley

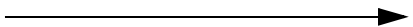
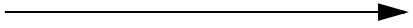
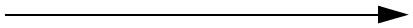
Parent: Local

Name:

Description:

Comm Format: Listen Only - Input Data

Typy vlastnictví modulu jsou popsány v následující tabulce:

Pokud je modulem	a jiný procesor	a pokud chcete	použijte tento typ spojení:
vstupní modul	modul nevlastní		owner (nikoli listen-only)
	modul vlastní	udržovat s modulem spojení, i když je komunikace s jiným procesorem přerušena,	owner (nikoli listen-only) použijte stejnou konfiguraci jako jiný procesor.
		přerušit s modulem spojení i když je komunikace s jiným procesorem přerušena.	listen-only
výstupní modul	modul nevlastní		owner (tedy nikoli listen-only)
	modul vlastní		listen-only

Mezi řízením vstupních a výstupních modulů existují některé rozdíly.

K řízení	použijte vlastnictví typu	Popis
vstupních modulů	owner	Vstupní moduly jsou konfigurovány prvním procesorem, který s nimi naváže spojení jako vlastník (owner). Poté co je vstupní modul nakonfigurován (a vlastněn procesorem), mohou s ním navazovat spojení typu owner i jiné procesory. Pokud pak původní vlastník přeruší spojení s modulem, ostatní vlastníci z modulu dále dostávají data multicast. Všichni ostatní vlastníci musí mít stejná konfigurační data a stejný komunikační formát jako původní vlastník, jinak je pokus o spojení odmítnut.
	listen-only	Poté co je vstupní modul nakonfigurován (a vlastněn procesorem), mohou s ním navazovat spojení typu listen-only i jiné procesory. Takové procesory pak dostávají data multicast, i když modul vlastní jiný procesor. Pokud se vstupním modulem přeruší spojení všichni vlastníci, všechna spojení typu listen-only nadále nebudou dostávat data multicast.
výstupních modulů	owner	Výstupní moduly jsou konfigurovány prvním procesorem, který s nimi naváže spojení jako vlastník (owner). Pro výstupní moduly je dovoleno pouze jedno spojení typu owner. Pokud se takové spojení pokusí navázat jiný procesor, je pokus o spojení odmítnut.
	listen-only	Poté co je výstupní modul nakonfigurován (a vlastněn jedním procesorem), mohou s ním navazovat spojení typu listen-only i jiné procesory. Takové procesory pak dostávají data multicast, i když modul vlastní jiný procesor. Pokud se výstupním modulem přeruší spojení jeho vlastník, všechna spojení typu listen-only nadále nebudou dostávat data multicast.

Electronic Keying (Elektronická identifikace)

POZOR


Při vypnutí elektronické identifikace postupujte opatrně. Tato funkce může při nesprávném použití vést ke zranění či smrti, škodám na majetku a ekonomickým ztrátám.

Při konfiguraci modulu specifikujete číslo jeho pozice. Na stejnou pozici lze však umístit i jiný modul, buď úmyslně, nebo v důsledku chyby.

Funkce Electronic keying váš systém chrání před nechtěným umístěním nesprávného modulu do určité pozice. Jednotlivá nastavení této funkce rozhodují o tom, jak přísně musí modul splňovat nastavení pozice.

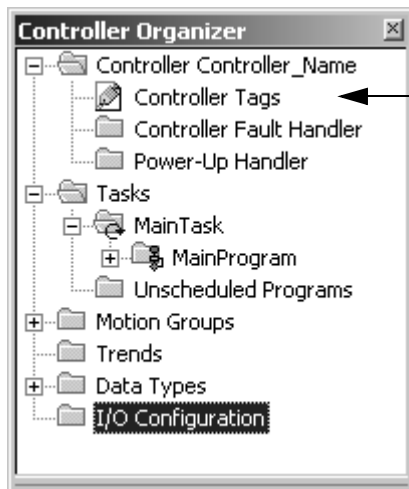
Pokud chcete, aby...	Pak vyberte možnost:
odpovídaly všechny informace: <ul style="list-style-type: none"> • typ • katalogové číslo • dodavatel • číslo major a minor revision 	Exact Match
... všechny informace kromě čísla minor revision	Compatible Module
... neodpovídaly žádné informace	Disable Keying

Adresy dat I/O

Informace I/O jsou prezentovány jako skupina tagů.

- Každý tag má nějakou datovou strukturu. Tato struktura závisí na specifických vlastnostech modulu I/O.
- Názvy tagů závisí na pozici modulu I/O v systému.

Po přidání modulu do složky I/O Configuration...



...software pro modul automaticky vytvoří tagy omezené na procesor.

Adresa I/O má následující formát:

Umístění :Pozice :Typ .Člen .Podčlen .Bit

☐ = volitelně

Kde	Je
Location (<i>Umístění</i>)	umístění v síti LOCAL = stejný rám nebo lišta DIN jako procesor ADAPTER_NAME = identifikuje vzdálený komunikační adaptér nebo modul bridge
Slot (<i>Pozice</i>)	číslo pozice modulu I/O v příslušném rámu nebo na liště DIN
Type (<i>Typ</i>)	datový typ I = vstup (input) O = výstup (output) C = konfigurace (configuration) S = stav (status)
Member (<i>Člen</i>)	specifická data z modulu I/O; závisí na tom, jaký typ dat může modul uchovávat. <ul style="list-style-type: none"> • U digitálních modulů datový člen (data member) obvykle uchovává vstupní nebo výstupní hodnoty. • U analogových modulů kanálový člen (CH#, channel member) obvykle uchovává data pro daný kanál.
SubMember (<i>Podčlen</i>)	specifická data týkající se členu
Bit	specifický bod digitálního modulu I/O; závisí na velikosti modulu I/O (0–31 pro 32bodový modul)

Kopírování dat I/O (Buffering I/O)

Buffering je technika, při které logika přímo neodkazuje na a nemanipuluje s tagy skutečných prvků I/O. Místo toho používá kopie dat I/O. Tuto techniku je vhodné používat v následujících situacích:

- Pokud je nutné zabránit změně vstupní nebo výstupní hodnoty během vykonávání programu. (data modulu I/O jsou aktualizována vzhledem k provádění logiky asynchronně)
- Pokud potřebujete zkopírovat vstupní nebo výstupní tag do člena struktury nebo prvku pole.

Kopírování dat I/O

Ke zkopírování dat I/O proveďte následující kroky:

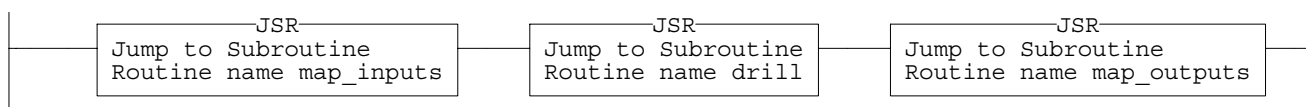
1. Na řádku před logikou funkce zkopírujte nebo přesuňte data z požadovaných vstupních tagů do odpovídajících tagů pro kopírování.
2. V logice funkce odkažte na tagy kopírování.
3. Na řádku po samotné funkci zkopírujte data z tagů kopírování do odpovídajících výstupních tagů.

V následujícím příkladu provedeme zkopírování vstupů a výstupů do tagů struktury vrtacího stroje.

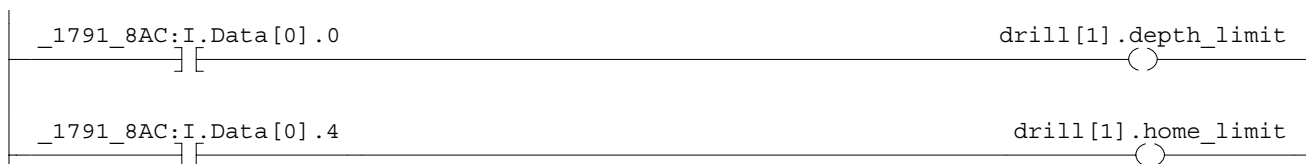
PŘÍKLAD

Kopírování dat I/O

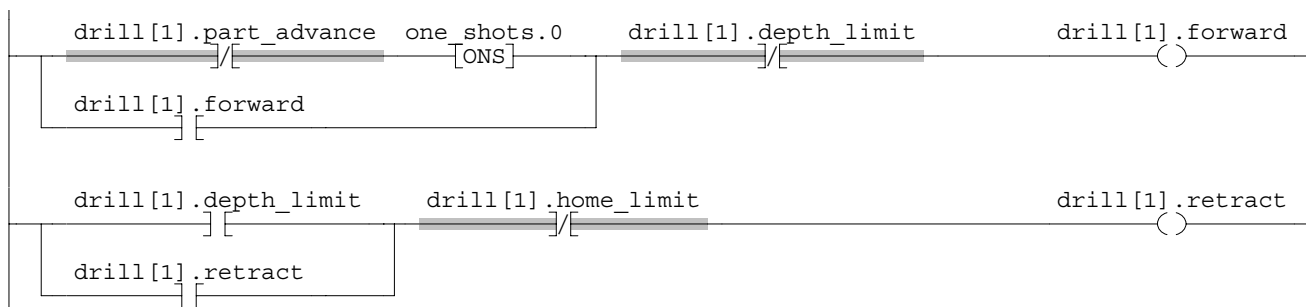
Hlavní rutina programu vykoná následující dílčí rutiny.



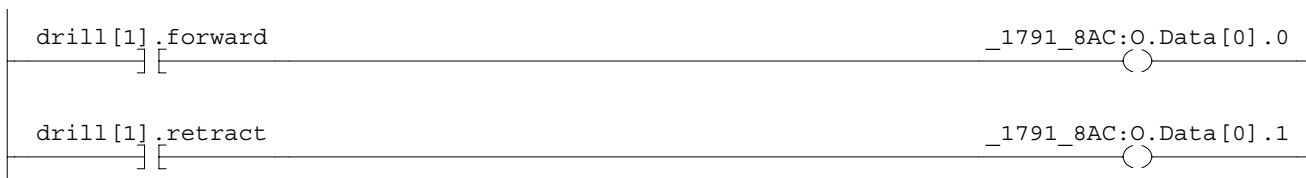
Rutina map_inputs zkopíruje hodnoty vstupních prvků do jejich tagů které jsou použity v rutině drill.



Rutina drill vykoná logiku operace vrtacího stroje.



Rutina map_outputs zkopíruje hodnoty výstupních tagů v rutině drill do odpovídajících výstupních prvků.



42369

V následujícím příkladě je ke zkopírování datového pole, které představuje vstupní prvky sítě DeviceNet použita instrukce CPS.

PŘÍKLAD

Kopírování I/O dat

Local:0:I.Data uchovává vstupní data sítě DeviceNet připojené k modulu 1756-DNB v pozici 0. Aby proběhla synchronizace vstupů s aplikací, CPS instrukce zkopíruje vstupní data do tagu `input_buffer`.

- Během kopírování dat CPS instrukcí se data nemohou změnit.
- Aplikace použije vstupní data z tagu `input_buffer`.



42578

Organizace tagů

Úvod

U procesorů Logix5000 se tagy (s alfanumerickými jmény) používají k adresování dat (proměnných).

Termín	Definice
Tag	<p>Textové jméno pro část paměti procesoru, kde jsou uložena data.</p> <ul style="list-style-type: none"> • Tagy jsou základním mechanismem pro alokaci paměti, odkazování na data z logiky a monitorování dat. • Minimální alokace paměti na tag je 4 byty. • Pokud vytvoříte tag, který obsahuje data vyžadující méně než čtyři byty, procesor mu přidělí čtyři byty, ale data zaplní pouze tolik bytů, kolik potřebují.

Procesor používá jména tagů interně a nepotřebuje odkazovat na fyzickou adresu.

- U běžných programovatelných řídicích jednotek identifikuje každý datový prvek fyzická adresa.
 - Adresy jsou založeny na fixním numerickém formátu, který závisí na typu dat, například N7:8, F8:3 atd.
 - Pro interpretaci logiky je nutné používat symboly.
- U procesorů Logix5000 nemají tagy fixní numerický formát. Data identifikuje sám název tagu. Díky tomu lze:
 - organizovat data podle rozložení strojů,
 - dokumentovat (pomocí názvů tagů) vaši aplikaci během jejího vývoje.

PŘÍKLAD

Tagy

Analogový prvek I/O →

Celočíselná hodnota →

Ukládací bit →

Čítač →

Časovač →

Digitální I/O prvek →

Program Tags - MainProgram				
Scope:	MainProgram	Show:	Show All	Sort: Tag Name
	Tag Name	Alias For	Base Tag	Type
	north_tank_mix			BOOL
	north_tank_pressure			REAL
	north_tank_temp			REAL
	+one_shots			DINT
	+recipe			TANK[3]
	+recipe_number			DINT
	replace_bit			BOOL
	+running_hours			COUNTER
	+running_seconds			TIMER
	start			BOOL
	stop			BOOL
Monitor Tags Edit Tags				

Typy tagů

Typ tagů definuje chování tagu v projektu.

Pokud chcete, aby tag	Použijte tento typ
Ukládal hodnoty, používané logikou projektu	Base
Reprezentoval jiný tag	Alias
Zasílal data jinému procesoru	Produced
Přijímal data od jiného procesoru	Consumed

Pokud chcete používat produced/vytvořené a consumed/přijaté tagy, musíte se řídit i dalšími pravidly.

Datový typ

Termín	Definice
Data type/ Datový typ	Datový typ definuje typ dat, které tag ukládá, například bit, celé číslo, hodnotu s pohyblivou řádovou čárkou, řetězec atd.
Structure/ Struktura	<p>Datový typ který je kombinací jiných typů.</p> <ul style="list-style-type: none"> • Struktura vytváří unikátní datový typ, který odpovídá specifickým potřebám. • V rámci struktury je každý individuální datový typ nazýván členem. • Podobně jako tagy mají členy jména a datové typy. • Procesory Logix5000 obsahují množinu předdefinovaných struktur (datových typů) pro použití se specifickými instrukcemi jako jsou časovače, čítače, funkční bloky, atd. • Uživatelé mohou kromě toho vytvářet vlastní struktury, nazývané uživatelem definované datové typy.

Následující tabulka obsahuje seznam nejběžnějších datových typů a jejich použití.

Pro	Vyberte
Analogový prvek v režimu s pohyblivou řádovou čárkou	REAL
Analogový prvek v režimu celé číslo (pro rychlé vzorkování)	INT
ASCII znaky	String
Bit	BOOL
Čítač	COUNTER
Digitální I/O prvek	BOOL
Číslo s pohyblivou řádovou čárkou	REAL
Celé číslo	DINT
Řadič	CONTROL
Časovač	TIMER

Minimální alokace paměti na tag je 4 byty. Pokud vytvoříte tag, který obsahuje data, vyžadující méně než čtyři byty, procesor mu přidělí čtyři byty, ale data zaplní pouze tolik bytů, kolik potřebují.

Datový typ	Bity				
	31	16	15	8	7 1 0
BOOL	nepoužito				0 nebo 1
SINT	nepoužito				-128 až +127
INT	nepoužito				-32 768 až +32 767
DINT	-2 147 483 648 až +2 147 483 647				
REAL	-3,40282347E ³⁸ až -1,17549435E ⁻³⁸ (negativní hodnoty)				
	0				
	1,17549435E ⁻³⁸ až 3,40282347E ³⁸ (pozitivní hodnoty)				

Nejčastěji používané datové typy jsou COUNTER a TIMER.

Pro rozbalení struktury a zobrazení jejích členů klepněte na symbol +.

Pro sbalení struktury a schování jejích členů klepněte na symbol –.

Členové running_seconds

Program Tags - MainProgram				
Scope:	MainProgram	Show:	Show All	Sort: T.
Tag Name	Alias For	Base Tag	Type	
+ -running_hours			COUNTER	← COUNTER
- -running_seconds			TIMER	← TIMER
+ -running_seconds.PRE			DINT	Datové typy členů
+ -running_seconds.ACC			DINT	
- -running_seconds.EN			BOOL	
- -running_seconds.TT			BOOL	
- -running_seconds.DN			BOOL	
- -running_seconds.FS			BOOL	
- -running_seconds.LS			BOOL	
- -running_seconds.OV			BOOL	
- -running_seconds.ER			BOOL	

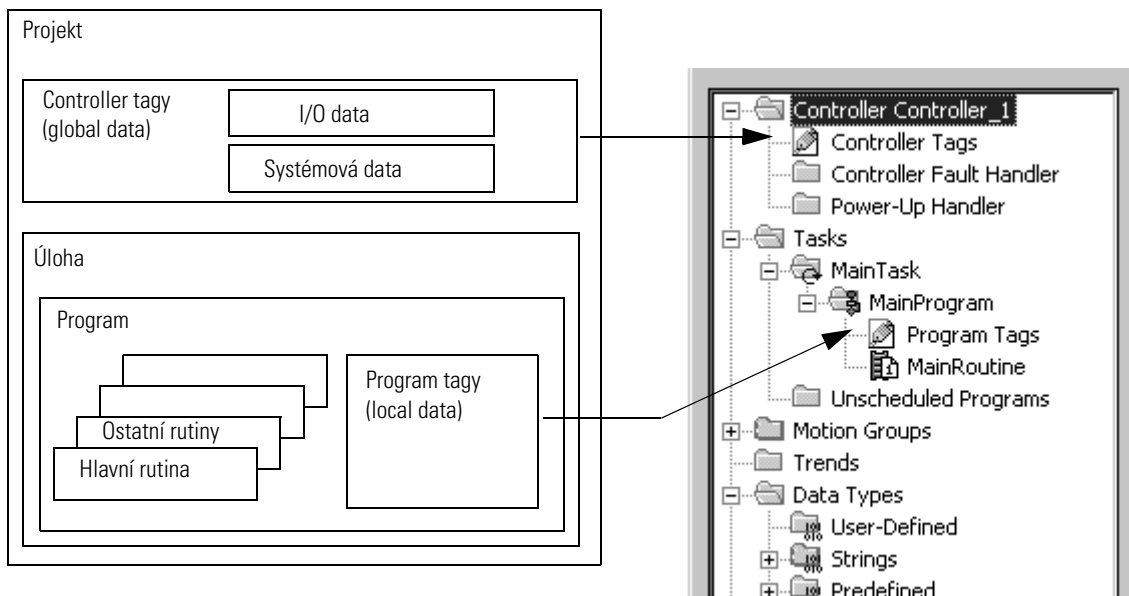
42365

Pro zkopírování dat do struktury použijte instrukci COP.

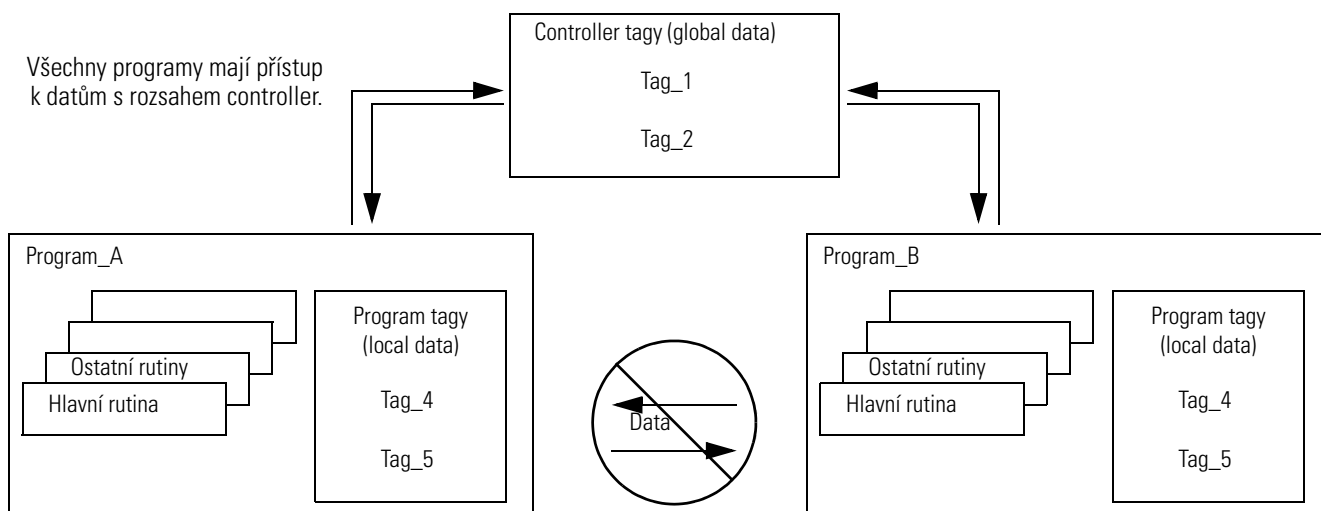
Viz. Logix5000 Controllers General Instruction Set Reference Manual, publikace 1756-RM003.

Rozsah

Při vytvoření tagu jej definujeme buď jako controller tag (globální data), nebo program tag (lokální data).



Procesor Logix5000 umožňuje rozdělení aplikace na více programů z nichž každý má svá vlastní data. Není pak nutné zabývat se konfliktními názvy tagů a lze znovupoužívat kód ve více programech.



Data v rozsahu program jsou od ostatních programů oddělena.

- K datům jiného programu v rozsahu program nemají rutiny přístup.
- Jméno tagu, který je omezen na jeden program, tedy můžete použít ve více programech.

Například jak Program_A tak Program_B mohou obsahovat tag, pojmenovaný Tag_4.

Vyhňte se použití stejného jména pro tagy v rozsahu procesoru a programu. V rámci jednoho programu nelze odkazovat na controller tag pokud existuje i program tag se stejným jménem.

Některé tagy musí být rozsahu procesor (controller tag).

Pokud chcete tag použít	zvolte rozsah
Ve více než jednom programu	Procesor (controller tag)
V instrukci MSG	
K vytvoření nebo přijímání dat	
Ke komunikaci s terminálem PanelView	
K žádné z těchto možností	Program (program tag)

Použití tagů

Při vytváření tagů do projektů Logix5000 se řiďte těmito pravidly:

Pravidlo	Detaily
<input type="checkbox"/> 1. Vytvářejte uživatelem definované datové typy.	<p>Uživatelem definované datové typy (struktury) dovolují organizaci dat podle stroje nebo procesu. Uživatelem definované datové typy mají tyto výhody:</p> <ul style="list-style-type: none"> Jeden tag obsahuje všechna data týkající se specifického aspektu systému. Související data jsou tak u sebe a lze je snadno nalézt bez ohledu na jejich typ. Každý datový prvek (člen) má popisné jméno. Tím je automaticky vytvořena základní úroveň dokumentace logiky. Tento datový typ lze používat k vytváření více tagů se stejným rozvržením dat. <p>Uživatelem definovaný datový typ lze například použít pro uložení všech parametrů nádrže včetně teplot, tlaků, pozic ventilů a předem nastavených hodnot. Poté lze pro každou nádrž vytvořit tag založený na tomto datovém typu.</p>
<input type="checkbox"/> 2. Pro rychlé vytváření skupin podobných tagů používejte pole/arrays.	<p>Pomocí pole lze vytvářet více instancí datového typu pod společným názvem tagu.</p> <ul style="list-style-type: none"> Pomocí polí lze sestavovat bloky tagů které používají stejné datové typy a mají podobnou funkci. Data lze používat v jednom, dvou nebo třech rozměrech. <p>Dvourozměrné pole lze například použít k organizaci dat pro skupinu nádrží. Každý prvek pole představuje jednu nádrž. Umístění prvku v poli představuje geografickou polohu nádrže.</p> <p>Důležité: Pole typu BOOL omezte na minimum. Pro BOOL pole jsou mnohé instrukce neplatné. Tím je inicializace pole BOOL dat značně ztížena.</p> <ul style="list-style-type: none"> BOOL pole lze většinou použít pro objekty PanelView obrazovky na úrovni bitů. V ostatních případech používejte jednotlivé bity DINT tagů nebo pole DINT tagů.

Pravidlo	Detaily									
<div><input type="checkbox"/> 3. Používejte tagy s rozsahem program.</div>	<p>Pokud chcete použít více tagů se stejným jménem, definujte každý tag na úrovni odpovídajícího programu. Díky tomu můžete v různých programech používat stejnou logiku i názvy tagů.</p> <p>Vyhnete se použití stejného jména pro tagy v rozsahu procesoru a programu. V rámci jednoho programu nelze odkazovat na controller tag pokud existuje i program tag se stejným jménem.</p> <p>Některé tagy musí být rozsahu procesor (controller tag).</p>									
	<table><tr><th>Pokud chcete tag použít</th><th>zvolte rozsah</th></tr><tr><td>Ve více než jednom programu</td><td rowspan="4">Procesor (controller tag)</td></tr><tr><td>V MSG instrukci</td></tr><tr><td>K vytvoření nebo přijímání dat</td></tr><tr><td>Ke komunikaci s terminálem PanelView</td></tr><tr><td>K žádné z těchto možností</td><td>Program (program tag)</td></tr></table>	Pokud chcete tag použít	zvolte rozsah	Ve více než jednom programu	Procesor (controller tag)	V MSG instrukci	K vytvoření nebo přijímání dat	Ke komunikaci s terminálem PanelView	K žádné z těchto možností	Program (program tag)
	Pokud chcete tag použít	zvolte rozsah								
	Ve více než jednom programu	Procesor (controller tag)								
	V MSG instrukci									
	K vytvoření nebo přijímání dat									
	Ke komunikaci s terminálem PanelView									
K žádné z těchto možností	Program (program tag)									
<div><input type="checkbox"/> 4. Pro celá čísla používejte datový typ DINT.</div>	<p>Pro zvýšení efektivity logiky omezte použití SINT a INT data typů na minimum. Pro celá čísla používejte kdykoli to je možné datový typ DINT.</p> <ul style="list-style-type: none">• Procesory Logix5000 většinou porovnávají nebo používají hodnoty v 32bitovém formátu (DINT nebo REAL).• SINT nebo INT hodnoty jsou procesorem před použitím většinou konvertovány na DINT nebo REAL hodnoty.• Pokud je cílem SINT nebo INT tag, procesor většinou hodnotu konvertuje zpět na SINT nebo INT.• Konverze na nebo z SINT a INT je automatická a nevyžaduje žádné programování. Vyžaduje nicméně čas na provedení a paměťovou kapacitu.									

Pravidlo	Detaily										
<div><div></div><div>5. Omezte jména tagů na 40 znaků.</div></div>	<div>Pravidla pro sestavování názvů tagů:</div> <div><div><div></div><div>Používejte pouze písmena (A–Z nebo a–z), čísla (0–9) a podtržítka (_)</div></div><div><div></div><div>názvy musí začínat písmenem nebo podtržítkem</div></div><div><div></div><div>názvy nesmí být delší než 40 znaků</div></div><div><div></div><div>podtržítka nesmí být vícenásobná nebo na konci (_)</div></div><div><div></div><div>názvy nerozlišují mezi malými a velkými písmeny</div></div></div>										
<div><div></div><div>6. Používejte velká a malá písmena.</div></div>	<div>Ačkoli tagy nerozlišují mezi malými a velkými písmeny (velké <i>A</i> je stejné jako malé <i>a</i>), použití velkých a malých písmen usnadňuje čtení.</div> <div><table><tr><th>Tyto názvy se čtou lépe</th><th>než tyto</th></tr><tr><td>Tank_1</td><td>TANK_1</td></tr><tr><td>Tank1</td><td>TANK1</td></tr><tr><td></td><td>Tank_1</td></tr><tr><td></td><td>Tank1</td></tr></table></div>	Tyto názvy se čtou lépe	než tyto	Tank_1	TANK_1	Tank1	TANK1		Tank_1		Tank1
Tyto názvy se čtou lépe	než tyto										
Tank_1	TANK_1										
Tank1	TANK1										
	Tank_1										
	Tank1										
<div><div></div><div>7. Všímejte si abecedního pořadí tagů.</div></div>	<div>Software RSLogix 5000 zobrazuje tagy stejného rozsahu v abecedním pořadí. Pro snadnější sledování souvisejících tagů použijte pro tagy které chcete zobrazit společně podobné začáteční znaky.</div> <div><div><div><div>Jestliže použijeme v začátku názvu každého tagu, souvisejícího s nádrží (tank) slovo Tank, tagy zůstanou u sebe.</div><div><table><tr><th>Název tagu</th></tr><tr><td>Tank_North</td></tr><tr><td>Tank_South</td></tr><tr><td>...</td></tr></table></div></div><div><div><div>Pokud bychom to neudělali, mohou být tagy odděleny a na různých místech.</div><div><table><tr><th>Název tagu</th></tr><tr><td>North_Tank</td></tr><tr><td>...</td></tr><tr><td>...</td></tr><tr><td>...</td></tr><tr><td>South_Tank</td></tr></table></div><div><div>Jiné tagy, začínající na písmena <i>o</i>, <i>p</i>, <i>q</i>, atd.</div><div></div></div></div></div></div></div>	Název tagu	Tank_North	Tank_South	...	Název tagu	North_Tank	South_Tank
Název tagu											
Tank_North											
Tank_South											
...											
Název tagu											
North_Tank											
...											
...											
...											
South_Tank											

Vytváření tagů

DŮLEŽITÉ
UPOZORNĚNÍ

Software RSLogix 5000 automaticky vytváří tagy pokud:

- přidáte prvek do (SFC – Sequential Function Chart, postupový funkční graf).
- přidáte instrukci v diagramu funkčních bloků do diagramu funkčních bloků.

Okno Tags umožňuje vytváření a editaci tagů pomocí tabulkového zobrazení.

1. Z nabídky Logic vyberte položku Edit Tags.
2. Vyberte pro tag rozsah (scope).



Pokud chcete tag použít	Vyberte
Ve více než jednom programu v rámci projektu	<i>name_of_controller(procesor)</i>
K vytváření nebo přijímání dat	
V MSG instrukci	
Jen v jednom programu v rámci projektu	Program, který bude tag používat

3. Vložte jméno, datový typ a popis (volitelně) tagu.

Vytvoření pole

Procesory Logix5000 dovolují pro uspořádání dat používat i pole.

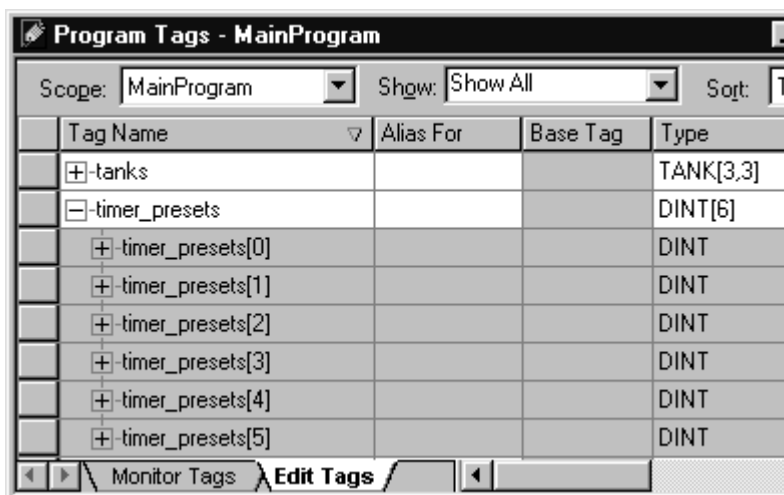
Termín	Definice
Array/pole	<p>Tag, který obsahuje blok více dat.</p> <ul style="list-style-type: none"> Pole je podobné souboru. V rámci pole se každý individuální datový segment nazývá prvkem. Každý prvek používá stejný datový typ. Tag typu array v procesoru obsadí sousedící bloky paměti. K manipulaci s nebo seřazení prvků v poli lze použít instrukce typu array a sequencer. Data lze sestavovat do bloků s jedním, dvěma nebo třemi rozměry.

Každý prvek pole je identifikován spodním indexem. Ten začíná na nule a končí na počtu prvků mínus 1.

Pro rozbalení pole a zobrazení jeho prvků klepněte na symbol +.

Pro sbalení pole a schování jeho prvků klepněte na symbol –.

Prvky pole timer_presets



Toto pole obsahuje šest prvků typu DINT.

Šest DINT prvků

42367

Následující příklad obsahuje porovnání struktury a pole:

Toto je tag, který používá strukturu typu Timer.

Název tagu	Datový typ
<input type="checkbox"/> Timer_1	TIMER
<input checked="" type="checkbox"/> Timer_1.PRE	DINT
<input checked="" type="checkbox"/> Timer_1.ACC	DINT
Timer_1.EN	BOOL
Timer_1.TT	BOOL
Timer_1.DN	BOOL

Toto je tag, který používá pole typu Timer.

Název tagu	Datový typ
<input type="checkbox"/> Timers	TIMER[3]
<input checked="" type="checkbox"/> Timer[0]	TIMER
<input checked="" type="checkbox"/> Timer[1]	TIMER
<input checked="" type="checkbox"/> Timer[2]	TIMER

PŘÍKLAD

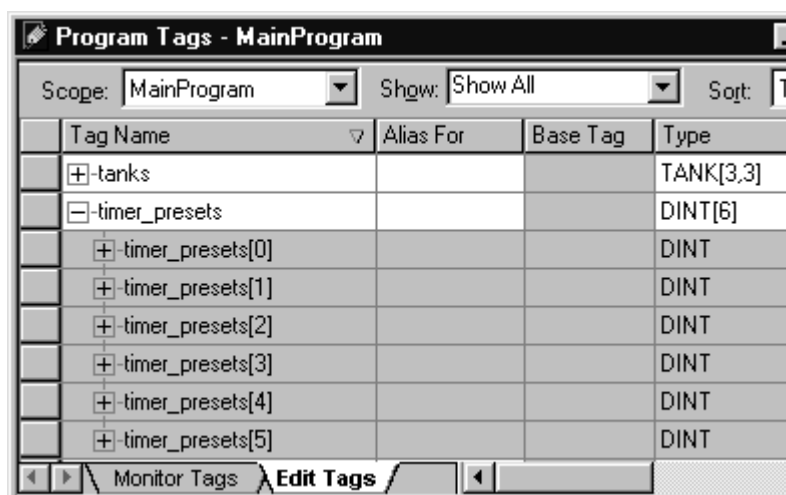
Jednorozměrné pole

V tomto příkladu sleduje jedna instrukce časovače čas provedení několika kroků. Každý krok vyžaduje jinou přednastavenou hodnotu. Protože všechny tyto hodnoty jsou stejného datového typu (DINT), je vhodné použít pole.

Pro rozbalení pole a zobrazení jeho prvků klepněte na symbol +.

Pro sbalení pole a schování jeho prvků klepněte na symbol –.

Prvky pole timer_presets



← Toto pole obsahuje šest prvků typu DINT.

— Šest DINT prvků

42367

PŘÍKLAD

Dvourozměrné pole

Vrtací stroj může do knihy vrtat jednu až šest děr. Pozice každé díry je dána vzdáleností od hrany knihy. Pro organizaci hodnot do konfigurace použijeme dvourozměrné pole. První spodní index označuje díru, ke které se hodnota vztahuje a druhý spodní index označuje počet děr, které budou vyvrtány (jednu až pět).

		Spodní index druhého rozměru						popis
		0	1	2	3	4	5	
Spodní index prvního rozměru	0							
	1		1.5	2.5	1.25	1.25	1.25	Pozice první díry – vzdálenost od hrany knihy
	2			8.0	5.5	3.5	3.5	Pozice druhé díry – vzdálenost od hrany knihy
	3				9.75	7.5	5.5	Pozice třetí díry – vzdálenost od hrany knihy
	4					9.75	7.5	Pozice čtvrté díry – vzdálenost od hrany knihy
	5						9.75	Pozice páté díry – vzdálenost od hrany knihy

V okně Tags jsou prvky zobrazeny v tomto pořadí.

Program Tags - MainProgram				
Scope: MainProgram		Show: Show All		Sort: T
Tag Name	Alias For	Base Tag	Type	
[-hole_position			REAL[6,6]	<p>← Toto pole obsahuje dvourozměrnou síť prvků (6x6 prvků).</p>
[-hole_position[0,0]			REAL	
[-hole_position[0,1]			REAL	
[-hole_position[0,2]			REAL	
[-hole_position[0,3]			REAL	
[-hole_position[0,4]			REAL	
[-hole_position[0,5]			REAL	
[-hole_position[1,0]			REAL	
[-hole_position[1,1]			REAL	
[-hole_position[1,2]			REAL	
[-hole_position[1,3]			REAL	

42367

→ Rozměr nejvíce vpravo se zvětší na svou maximální hodnotu a poté se vrátí na začátek.

→ Po návratu rozměru nejvíce vpravo na začátek se rozměr vlevo zvětší o jednu.

Vytvoření pole

Při vytváření pole vytvoříme tag a datovému typu přidělíme rozměry:

- 1. Z nabídky Logic vyberte položku Edit Tags.



42350

- 2. Vložte název tagu a vyberte jeho rozsah:

Pokud chcete tag použít	Vyberte možnost
Ve více než jednom programu v rámci projektu	<i>name_of_controller (procesor)</i>
K vytváření nebo přijímání dat	
V MSG instrukci	
Jen v jednom programu v rámci projektu	Program který bude tag používat

- 3. Přidělte rozměry pole:

Pokud je tag	Vložte	Kde
Jednorozměrný	<i>data_type[x]</i>	<i>data_type</i> je typ dat, které tag ukládá. <i>x</i> je počet prvků v prvním rozměru. <i>y</i> je počet prvku ve druhém rozměru. <i>z</i> je počet prvku ve třetím rozměru.
Dvourozměrné pole	<i>data_type[x,y]</i>	
Třírozměrné pole	<i>data_type[x,y,z]</i>	

Vytvoření uživatelem definovaného datového typu

Uživatelem definované datové typy (struktury) dovolují organizaci dat podle stroje nebo procesu.

PŘÍKLAD

Uživatelem definovaný datový typ obsahující recepturu

V systému s několika nádržemi může každá nádrž pracovat s různou recepturou. Protože receptury vyžadují kombinaci různých datových typů (REAL, DINT, BOOL, atd.) používá se uživatelem definovaný datový typ.

Jméno (datového typu): TANK

Jméno člena	Datový typ
Temp (teplota)	REAL
Deadband (pásmo necitlivosti)	REAL
Step (krok)	DINT
Step_time (čas kroku)	TIMER
Preset (přednastavení)	DINT[6]
Mix	BOOL

Pole v tomto datovém typu vypadá takto:

Pole receptur

První receptura

Členové receptury

Toto pole obsahuje tři prvky typu TANK.

42368

PŘÍKLAD

Uživatelé definovaný datový typ, ukládající data potřebná k provozu stroje

Protože tato data vyžaduje několik vrtacích stanic, je vhodné použít uživatelem definovaný datový typ

Jméno (datového typu): DRILL_STATION

Jméno člena	Datový typ
Part_advance (posun součástky)	BOOL
Hole_sequence (pořadí děr)	CONTROL
Type (typ)	DINT
Hole_position (pozice děr)	REAL
Depth (hloubka)	REAL
Total_depth (celková hloubka)	REAL

Pole v tomto datovém typu vypadá takto:

Pole vrtacích stanic

První vrtání

Data pro vrtačku

Toto pole obsahuje čtyři prvky datového typu DRILL_STATION.

42583

Pravidla pro používání uživatelem definovaných datových typů

Při vytváření uživatelem definovaných datových typů se řiďte následujícími pravidly:

- Pokud typ obsahuje členy, představující I/O prvky, je potřeba použít logiku, která zkopíruje data mezi členy ve struktuře a odpovídajícími I/O tagy. Viz. Adresy dat I/O na strana 13.
- Pokud vložíte jako člen pole, omezte jej na jeden rozměr. Vícerozměrná pole *nejdou* v uživatelem definovaných datových typech povoleny.
- Při použití datových typů BOOL, SINT, nebo INT umístěte členy, používající stejný datový typ za sebou:

Efektivnější

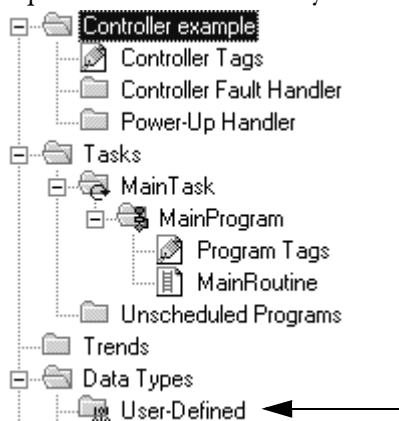
BOOL
BOOL
BOOL
DINT
DINT

Méně efektivní

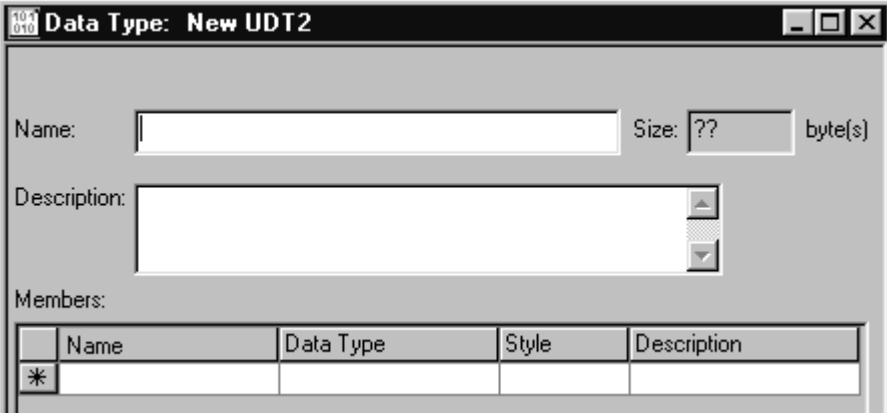
BOOL
DINT
BOOL
DINT
BOOL

Vytvoření uživatelem definovaného datového typu

1. Z adresáře User-Defined pod Data Types klepněte pravým tlačítkem myši na položku User-Defined a vyberte možnost New Data Type.



2. Vložte jméno pole a jeho popis.
3. Pro každého člena pole vložte jméno, datový typ, styl a popis (volitelně).



The screenshot shows a dialog box titled "Data Type: New UDT2". It has a "Name:" text box, a "Description:" text box with a vertical scrollbar, and a "Size:" text box followed by a unit dropdown menu currently set to "byte(s)". Below these fields is a "Members:" section containing a table with the following structure:

	Name	Data Type	Style	Description
*				

42196

Pokud vkládáte pole, omezte je jen na jeden rozměr.

Pro zobrazení hodnoty/hodnot člena v jiném stylu (radix) vyberte příslušný styl.

4. Klepněte na položku Apply.
5. Přidejte tolik členů, kolik je potřeba.

Popis uživatelem definovaného datového typu



Software RSLogix 5000 verze 13.0
nebo novější

Software RSLogix 5000 umožňuje automatické vytváření popisů pomocí popisů uživatelem definovaných datových typů. To značně redukuje množství času potřebného na dokumentaci projektu.

Při organizaci uživatelem definovaných data typů mějte na paměti následující vlastnosti softwaru RSLogix 5000:

Name	Data Type	Style	Description
Level	DINT	Decimal	Current Liters
Pressure	DINT	Decimal	Kpa
Temp	REAL	Float	Degrees C
Agitator_Speed	DINT	Decimal	RPM of Agitator
Ingredient_A	BOOL	Decimal	Add Red
Ingredient_B	BOOL	Decimal	Add Blue

Předávání popisů – Pokud je to možné, software RSLogix 5000 vyhledá stávající popis tagu, prvku, nebo členu.

- Popisy uživatelem definovaných datových typů přecházejí na tagy které příslušný datový typ používají.
- Popis tagu typu pole přechází na prvky a členy tohoto pole.

P	Tag Name	Type	Description
<input type="checkbox"/>	-Tanks	Tank[4]	Tank
<input type="checkbox"/>	-Tanks[0]	Tank	Tank
<input type="checkbox"/>	+Tanks[0].Level	DINT	Tank Current Liters
<input type="checkbox"/>	+Tanks[0].Pressure	DINT	Tank Kpa
<input type="checkbox"/>	-Tanks[0].Temp	REAL	Tank Degrees C
<input type="checkbox"/>	+Tanks[0].Agitator_Speed	DINT	Tank RPM of Agitator
<input type="checkbox"/>	-Tanks[0].Ingredient_A	BOOL	Tank Add Red
<input type="checkbox"/>	-Tanks[0].Ingredient_B	BOOL	Tank Add Blue...
<input type="checkbox"/>	-Tanks[1]	Tank	West Tank
<input type="checkbox"/>	+Tanks[1].Level	DINT	West Tank Current Liters
<input type="checkbox"/>	+Tanks[1].Pressure	DINT	West Tank Kpa
<input type="checkbox"/>	-Tanks[1].Temp	REAL	West Tank Degrees C

Doplnění popisu k základnímu tagu – software RSLogix 5000 automaticky vytváří popis pro každý člen tagu používající uživatelem definovaný datový typ. Popis obsahuje popis tagu a dále popis členu z datového typu.

Vložení předaného popisu – popis datového typu a pole lze použít pro vytvoření specifického popisu.

V následujícím případě se z nádrže (Tank) stane západní nádrž (West Tank).

Software RSLogix 5000 používá pro popisy různé barvy:

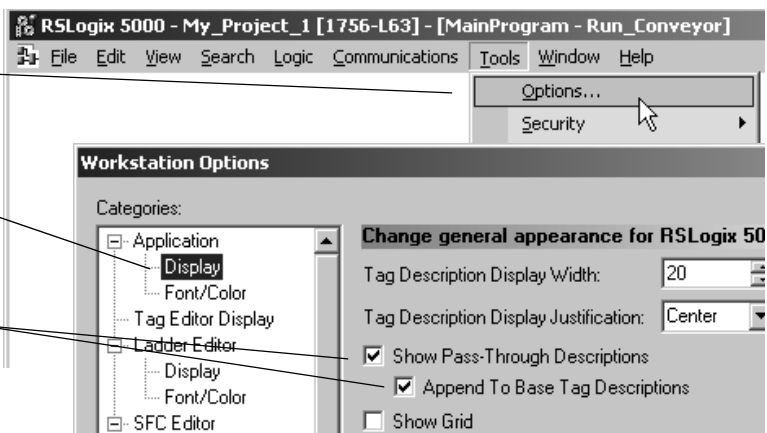
Popis v barvě	je
Šedé	předaný popis
Černé	ručně vložený popis

Zapnutí a vypnutí předaných a doplněných popisů

1. V softwaru RSLogix 5000 vyberte v nabídce Tools > položku Options.

2. Poté vyberte Application > Display.

3. Zapněte (zaškrtněte) nebo vypněte (zrušte zaškrtnutí) požadovaných možností.



Vložení předaného popisu

Pro použití předaného popisu jako základu pro specifitější popis:

1. Klepněte pravým tlačítkem myši na předaný popis a vyberte možnost Paste Pass-Through.

2. Provedte editaci popisu a poté zadejte [Ctrl] + [Enter].

P	Tag Name	Type	Description
<input type="checkbox"/>	-Tanks	Tank[4]	Tank
<input type="checkbox"/>	+Tanks[0]	Tank	Tank
<input checked="" type="checkbox"/>	+Tanks[1]	Tank	West Tank
<input type="checkbox"/>	+Tanks[2]	Tank	Tank
<input type="checkbox"/>	+Tanks[3]	Tank	Tank

Adresování dat tagů

Název tagu má následující formát:

Jméno

[Prvek]

.Člen

[Prvek]

.Bit

nebo

.[Index]

 = volitelně

Kde	Je
<i>Jméno</i>	Jméno identifikující specifický tag.
<i>Prvek</i>	<p>Spodní index nebo indexy které odkazují na specifický prvek pole.</p> <ul style="list-style-type: none"> Identifikátor prvku použijte pouze pokud je člen tagu polem. Použijte jeden spodní index pro každý rozměr pole. Například [5], [2,8], [3,2,7]. <p>Pro nepřímý (dynamický) odkaz na prvek použijte tag nebo numerický výraz který dodá číslo prvku.</p> <ul style="list-style-type: none"> Numerický výraz používá k výpočtu hodnoty kombinaci tagů, konstant, operátorů a funkcí. Například: Tag_1-Tag_2, Tag_3+4, ABS (Tag_4). Hodnotu tagu nebo numerického výrazu udržujte v rámci rozměrů pole. Například pokud rozměr pole obsahuje 10 prvků, hodnota tagu nebo numerického výrazu musí být mezi 0 a 9 (10 prvků).
<i>Člen</i>	<p>Specifický člen struktury.</p> <ul style="list-style-type: none"> Identifikátor člena používejte pouze pokud je tag strukturou. Pokud struktura obsahuje jako jeden ze svých členů další strukturu, použijte k jeho identifikaci další úrovně formátu .Member.
<i>Bit</i>	Specifický bit datového typu celé číslo (SINT, INT, nebo DINT).
<i>Index</i>	<p>Pro nepřímý (dynamický) odkaz na bit celého čísla použijte tag nebo numerický výraz který dodá číslo bitu.</p> <ul style="list-style-type: none"> Numerický výraz používá k výpočtu hodnoty kombinaci tagů, konstant, operátorů a funkcí. Například: Tag_1-Tag_2, Tag_3+4, ABS (Tag_4). Hodnotu tagu nebo numerického výrazu udržujte v rámci rozsahu bitů tagu typu integer. Například pokud je integer tag tagem typu Dint (32 bitů), hodnota indexu musí být 0 až 31 (32 bitů).

Přidělování Alias tagů

Alias tag umožňuje vytvoření tagu, reprezentujícího jiný tag.

- Oba tagy mají stejnou hodnotu.
- Když se změní hodnota jednoho tagu, druhý tag změnu reflektuje.

Alias tagy jsou užitečné v těchto situacích:

- Programování logiky před finalizací zapojení.
- Přidělení popisného jména prvku I/O.
- Zjednodušené jméno složitějšího tagu.
- Popisné jméno pro prvek v poli.

Okno Tags obsahuje i informace o aliasech.

drill_1_depth_limit je alias pro tag Local:2:I.Data.3 (digitální vstupní bod). Při zapnutí vstupu se zapne i alias tag.

drill_1_on je alias pro tag Local:0:O.Data.2 (digitální výstupní bod). Při zapnutí alias tagu se zapne i výstupní tag.

north_tank je alias pro tanks[0,1].

Tag Name	Alias For	Base Tag	Type
drill_1			DRILL_STAT
drill_1_depth_limit	Local:2:I.Data.3(C)	Local:2:I.Data.3(C)	BOOL
drill_1_forward	Local:0:O.Data.3(C)	Local:0:O.Data.3(C)	BOOL
drill_1_home_limit	Local:2:I.Data.2(C)	Local:2:I.Data.2(C)	BOOL
drill_1_on	Local:0:O.Data.2(C)	Local:0:O.Data.2(C)	BOOL
drill_1_retract	Local:0:O.Data.4(C)	Local:0:O.Data.4(C)	BOOL
hole_position			REAL[6,6]
machine_on			BOOL
north_tank	tanks[0,1]	tanks[0,1]	TANK
north_tank_drain			BOOL

42360

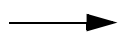
(C) znamená, že tag má rozsah procesor.

Častým důvodem pro použití alias tagů je programování logiky před samotným zapojením.

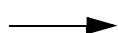
1. Pro každý I/O prvek vytvořte tag s názvem, který popisuje příslušný prvek, například conveyor pro motor dopravníku.
2. Logiku naprogramujte pomocí těchto popisných názvů. (Logiku lze dokonce otestovat bez připojení k I/O)
3. Když jsou k dispozici schémata zapojení, přidejte I/O moduly do I/O configuration procesoru.
4. Popisné tagy pak konvertujte na aliasy pro příslušné I/O body nebo kanály.

Následující logika byla naprogramována pomocí popisných názvů tagů, jako například stop a conveyor_on. Později byly tagy konvertovány na aliasy příslušných I/O prvků.

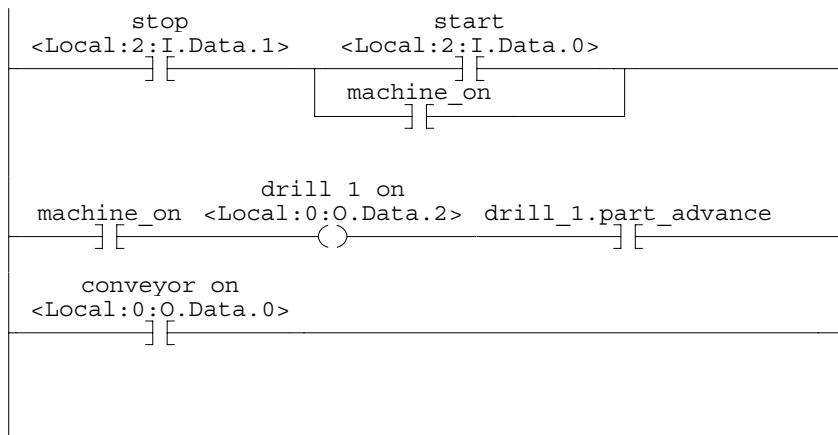
stop je alias pro Local:2:I.Data.1
(tlačítko stop na panelu obsluhy)



conveyor_on je alias pro
Local:0:O.Data.0



(spouštěcí kontakt motoru
dopravníku)



42351

Zobrazení informací o aliasu

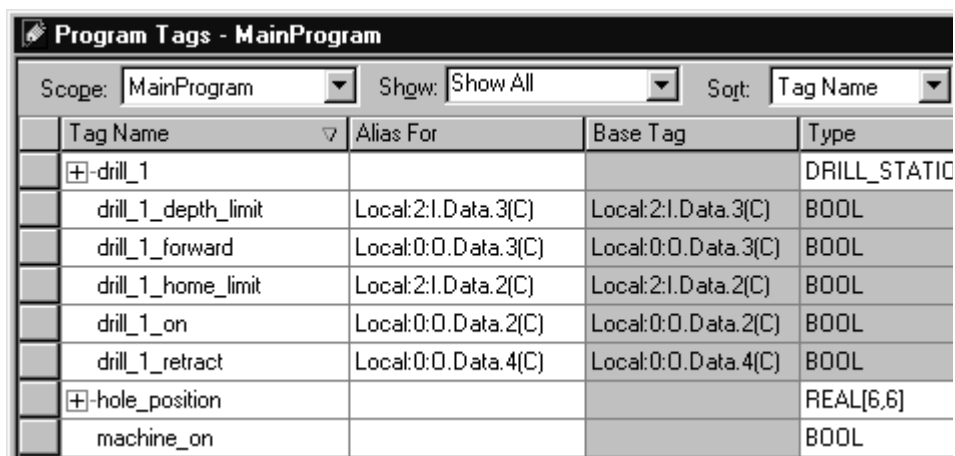
Pro zobrazení tagu na který odkazuje alias v logice proveďte následující:

1. Z nabídky Tools vyberte možnost Options.
2. Vyberte záložku Ladder Display.
3. Zaškrtněte políčko Show Tag Alias Information.
4. Klepněte na tlačítko OK.

Přidělení aliasu

Pro zadání alias tagu proveďte následující kroky:

1. Z nabídky Logic vyberte položku Edit Tags.



Tag Name	Alias For	Base Tag	Type
[-]drill_1			DRILL_STATIC
drill_1_depth_limit	Local:2:I.Data.3(C)	Local:2:I.Data.3(C)	BOOL
drill_1_forward	Local:0:0.Data.3(C)	Local:0:0.Data.3(C)	BOOL
drill_1_home_limit	Local:2:I.Data.2(C)	Local:2:I.Data.2(C)	BOOL
drill_1_on	Local:0:0.Data.2(C)	Local:0:0.Data.2(C)	BOOL
drill_1_retract	Local:0:0.Data.4(C)	Local:0:0.Data.4(C)	BOOL
[-]hole_position			REAL[6,6]
machine_on			BOOL

42360

2. Vyberte pro tag rozsah (scope).
3. Klepněte na buňku Alias For napravo od názvu tagu.

Buňka obsahuje ▼.

4. Klepněte na ▼.
5. Vyberte tag, který bude alias reprezentovat:

Pro	Proveďte
Výběr tagu	Klepněte dvakrát na název tagu.
Výběr bit čísla	A. Klepněte na název tagu. B. Vpravo od názvu tagu klepněte na ▼. C. Klepněte na požadovaný bit.

6. Zadejte [Enter] nebo klepněte na jinou buňku.

Přidělení nepřímé adresy

Pokud chcete, aby instrukce měla přístup k různým prvkům pole, použijte ve spodním indexu pole tag (nepřímé adresování). Změnou hodnoty tagu pak změníte prvek pole na který se logika odkazuje.

Pokud index = 1, array[index] ukazuje na.

array[0]	4500
array[1]	6000
array[2]	3000
array[3]	2500

Pokud index = 2, array[index] ukazuje na.

Následující tabulka obsahuje některá běžná využití nepřímého adresování:

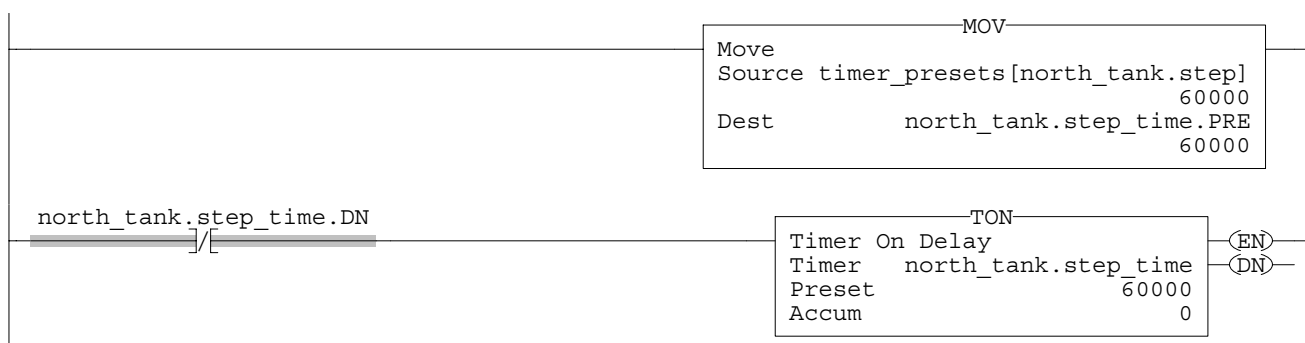
Pro	Použijte ve spodním indexu tag a
Výběr receptury z pole receptur	Vložte do tagu číslo receptury.
Načtení specifického nastavení stroje z pole možných nastavení	Vložte do tagu požadované nastavení.
Načtení parametrů nebo stavů z pole po jednotlivých prvcích	A. Proveďte požadovanou operaci na prvním prvku.
Log chybových kódů	B. Použijte ADD instrukci pro zvětšení hodnoty tagu a odkaz na další prvek pole.
Provedení několika operací na prvku pole a přechod na další prvek	

Následující příklad načte do časovače sérii přednastavených hodnot (jednu po druhé).

PŘÍKLAD

Pohyb polem

Pole timer_presets ukládá skupinu přednastavených hodnot pro časovač na dalším řádku. Tag north_tank.step ukazuje, který prvek pole použít. Například pokud north_tank.step = 0, instrukce načte do časovače 60 000 ms (timer_presets[0]).



Po ukončení tagu north_tank.step_time řádek zvýší north_tank.step na další číslo a příslušný prvek pole timer_presets se načte do časovače.



Když north_tank.step přesáhne velikost pole, řádek provede reset tagu pro začátek na prvním prvku pole. (Pole obsahuje prvky 0 až 3.)



42358

Výrazy

Výraz lze rovněž použít pro specifikaci dolního indexu pole.

- Výraz používá k výpočtu hodnot operátory jako například + nebo –.
- Procesor vypočítá výsledek výrazu a použije ji jako spodní index pole.

Ke specifikaci spodního indexu pole lze použít tyto operátory:

Operátor	Popis	Operátor	Popis
+	Součet	MOD	Modulo
–	Odečtení/negace	NOT	Doplňěk
*	Násobení	OR	OR
/	Dělení	SQR	Druhá odmocnina
ABS	Absolutní hodnotu	TOD	celé číslo na BCD
AND	AND	TRN	Zaokrouhlení
FRD	BCD na celé číslo	XOR	Výlučné OR

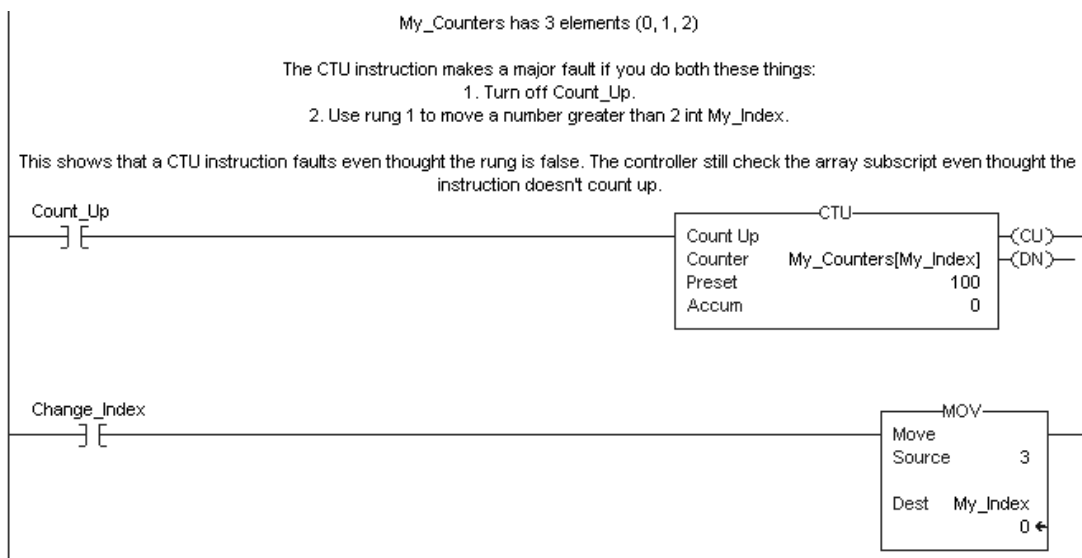
Výrazy formátujte takto:

Pokud operátor vyžaduje	Použijte tento formát	Příklady
Jednu hodnotu (tag nebo výraz)	<i>operator(value)</i>	ABS(tag_a)
Dvě hodnoty (tagy, konstanty, nebo výrazy)	<i>value_a operator value_b</i>	<ul style="list-style-type: none"> • tag_b + 5 • tag_c AND tag_d • (tag_e ** 2) MOD (tag_f / tag_g)

Spodní index pole mimo rozsah

Pokud je spodní index pole mimo rozsah každá instrukce oznámí závažnou chybu. Přejížděcí instrukce také oznámí závažnou chybu, a to i pokud je řádek nepravdivý. Procesor zkontroluje pomocí těchto instrukcí spodní index pole i když je řádek nepravdivý.

PŘÍKLAD



Více informací o řešení závažných chyb viz. Major and Minor Faults Programming Manual, publikace 1756-PM014.

Dokumentace tagů

Protože lze vytvářet čtyři různé druhy tagů, popisy jednotlivých tagů se liší. Tagy mohou být těchto typů:

- Base (základní)
- Alias
- Produced (vytvořený)
- Consumed (přijatý)

DŮLEŽITÉ UPOZORNĚNÍ

Programovací software RSLogix 5000 automaticky přiděluje předávané (pass-through) popisy vytvořených tagů. Přidělování těchto popisů lze povolit nebo vypnout.

Dokumentace základních tagů

Při vytvoření tagu bez specifikace jeho typu RSLogix 5000 mu automaticky přidělí implicitní typ Base (základní).

Protože base tagy umožňují vytváření vlastního interního úložiště dat, v popisu tagu lze dokumentovat druh ukládaných dat.

Dokumentace alias tagů

Vytvořením alias tagu přidělíte vlastní jméno stávajícímu tagu, struktuře, členu tagu, nebo bitu. V popisu alias tagu je vhodné popsat tag na který alias odkazuje.

Dokumentace vytvořených tagů

Produced (vytvořený) tag odkazuje na tag, který je přijímán jiným procesorem. V popisu vytvořeného tagu popište vzdálené procesory kterým má být tag k dispozici pomocí procesu controller-to-controller messaging.

Dokumentace přijatých tagů

Consumed (přijatý) tag odkazuje na tag, vytvořený jiným procesorem a obsahující data která chceme použít v našem procesoru. V popisu přijatého tagu popisujte způsoby kterými chcete používat jeho data nebo procesor, který tag vytváří.

Přepínání jazyků

V software RSLogix 5000 verze 17 máte možnost zobrazit projektovou dokumentaci, například popisy tagů a komentáře k řádkům ve kterémkoli podporovaném lokalizačním jazyce. Projektová dokumentace pro více jazyků se ukládá do jednoho souboru, nikoli do souborů podle jednotlivých jazyků. Můžete definovat všechny jazyky, které bude projekt podporovat a nastavit aktuální, implicitní a volitelný jazyk. Pokud je obsah v aktuálním jazyce pro určitý prvek prázdný (není k dispozici), software použije implicitní jazyk. Pro úpravu dokumentace pro specifické uživatele lze použít volitelný jazyk.

Vložte do svého projektu RSLogix 5000 lokalizované popisy, buď přímo během programování nebo pomocí funkce import/export, překladu obsahu offline a jeho zpětného importu do projektu. Pokud je v projektu RSLogix 5000 povoleno přepínání jazyků, lze během používání softwaru dynamicky přepínat mezi jazyky.

Projektová dokumentace podporující v rámci projektu různé překlady zahrnuje:

- popis komponentů tagů, rutin, programů, uživatelem definovaných datových typů a Add-on instrukcí,
- fáze vybavení,
- trendy,
- procesory,
- výstražná hlášení (v konfiguraci ALARM_ANALOG a ALARM_DIGITAL),
- úlohy,
- popisy vlastností modulů v nástroji Controller Organizer,
- komentáře k řádkům, textové rámečky SFC a textové rámečky FBD,

více informací o překladech projektové dokumentace je k dispozici v online nápovědě.

Poznámky:

Pevné nastavení vstupní a výstupní hodnoty uživatelem (FORCE I/O)

Úvod

Pevná nastavení hodnot (force) se používají k přepsání dat použitých nebo vyprodukovaných logikou. Lze je například používat pro:

- testování logiky,
- kontroly kabeláže výstupního prvku,
- dočasné udržení funkce procesu po odpojení vstupního prvku.

Pevné nastavení hodnoty používejte vždy pouze dočasně. Není vhodné pro trvalé použití.

Upozornění

Při použití pevných nastavení hodnot dodržujte tato bezpečnostní opatření.

POZOR



Použití pevně nastavených hodnot může zapříčinit neočekávaný pohyb strojů a vybavení, které může zranit obsluhu. Před použitím této techniky zjistěte, jak budou stroje a procesy reagovat, a ujistěte se, že se v jejich blízkosti nenacházejí lidé.

- Povolení nastavení pevných hodnot I/O zapříčiní změnu vstupních, výstupních, produkovaných nebo použitých hodnot.
- Povolení nastavení pevných hodnot SFC způsobí změnu stavu nebo fáze stroje nebo procesu.
- I po odstranění pevných nastavení mohou některá z nich být přítomna.
- Pokud jsou pevná nastavení povolena a nějaké takové nastavení provedete, jeho efekt bude okamžitý.

Povolení nastavení pevných hodnot uživatelem (forces)

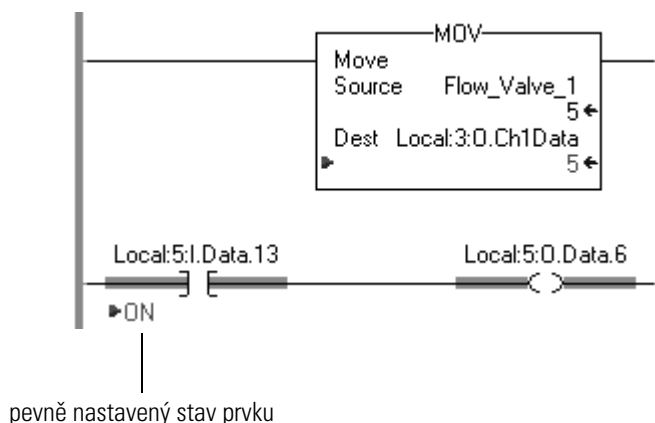
Aby pevné nastavení hodnoty uživatelem) proběhlo, je potřeba je povolit. Lze je povolit nebo zakázat pouze na úrovni procesoru.

- Pevná nastavení hodnot I/O a SFC uživatelem lze povolit nebo zakázat odděleně.
- Pevná nastavení hodnot uživatelem nelze povolit nebo zakázat pro určitý modul, soubor tagů nebo prvek tagu.

DŮLEŽITÉ UPOZORNĚNÍ

Pokud provedete stažení projektu, ve kterém jsou pevná nastavení uživatelem povolena, programovací software se dotáže, zda chcete po ukončení stahování pevná nastavení povolit nebo zakázat.

Pokud jsou pevná nastavení povolena, vedle prvku, kterého se týkají, se objeví symbol ►.



Zákaz nebo odstranění pevného nastavení (force)

Pro odstranění efektu pevného nastavení hodnot (force) a vykonání programu se skutečnými hodnotami je nutno pevné nastavení zakázat nebo odstranit.

- Pevná nastavení I/O a SFC (forces) mohou být zakázána a odstraněna odděleně.
- Odstranění pevného nastavení hodnoty na tagu aliasu odstraní pevné nastavení hodnoty i na příslušném základním tagu.

POZOR



Jakákoli změna pevných nastavení může zapříčinit neočekávaný pohyb strojů a vybavení, které může zranit obsluhu. Před zákazem nebo odstraněním pevných nastavení zjistěte, jak budou stroje a procesy reagovat, a ujistěte se, že se v jejich blízkosti nenacházejí lidé.

Kontrola stavu pevných nastavení hodnot (forces)

Před použitím pevného nastavení hodnot (force) vždy zjistěte stav pevných nastavení hodnot příslušného procesoru. Kontrola stavu se provádí následovně:

Pro zjištění stavu:	...použijte
pevných hodnot (forces) I/O	<ul style="list-style-type: none"> • nástrojovou lištu Online • FORCE LED • instrukci GSV
pevných hodnot (forces) SFC	Nástrojovou lištu Online

Nástrojová lišta Online zobrazuje stav pevných nastavení. Stav pevných nastavení I/O a SFC je zobrazen zvlášť.



Nastavení	Význam
Enabled	<ul style="list-style-type: none"> • Pokud program obsahuje jakékoli pevné nastavení hodnoty (force) tohoto typu, takové pevné nastavení bude použito a mění logiku. • Pokud přidáte pevné nastavení tohoto typu, bude okamžitě použito.
Disabled	Pevná nastavení tohoto typu nejsou aktivní. Pokud program obsahuje jakákoli pevná nastavení tohoto typu, tato nastavení nejsou použita a nemění logiku.
Installed	V projektu existuje alespoň jedna pevně nastavená hodnota tohoto typu.
None Installed	V projektu neexistují žádné pevně nastavené hodnoty tohoto typu.

FORCE LED

Pokud má procesor kontrolku FORCE LED, lze ji použít k zjištění stavu pevně nastavených hodnot .

DŮLEŽITÉ UPOZORNĚNÍ

Kontrolka FORCE LED zobrazuje stav pevných nastavení /O. Stav pevných nastavení SFC nezobrazuje!

Pokud kontrolka FORCE LED

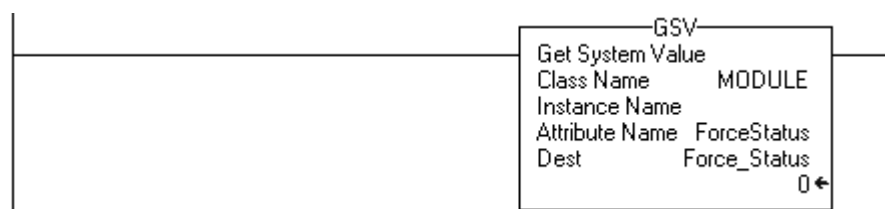
Pokud kontrolka FORCE LED	Proveďte
Nesvítí	<ul style="list-style-type: none"> Žádné tagy neobsahují pevně nastavené hodnoty. Pevná nastavení hodnot I/O nejsou aktivní (jsou zakázána).
Bliká	<ul style="list-style-type: none"> Alespoň jeden tag obsahuje pevně nastavenou hodnotu. Pevné hodnoty I/O nejsou aktivní (jsou zakázány).
Svítí	<ul style="list-style-type: none"> Pevné hodnoty I/O jsou aktivní (povoleny). Pevné hodnoty mohou, ale nemusí být použity.

Instrukce GSV

DŮLEŽITÉ UPOZORNĚNÍ

Atribut ForceStatus zobrazuje stav pevných hodnot I/O. Stav pevných hodnot SFC nezobrazuje!

Následující příklad ilustruje použití instrukce GSV k získání stavu pevně nastavených hodnot,



kde:

Force_Status je tag DINT.

Pro zjištění, zda	zkontrolujte tento bit	na hodnotu
jsou pevná nastavení hodnot nainstalována	0	1
pevná nastavení hodnot nejsou nainstalována	0	0
jsou pevná nastavení hodnot povolena	1	1
jsou pevná nastavení hodnot zakázána	1	0

Kdy použít pevné nastavení hodnot I/O

Kdy použít pevné nastavení hodnot (force) I/O:

- k přepsání vstupní hodnoty z jiného procesoru (například přijatého tagu),
- k přepsání vstupní hodnoty ze vstupního prvku,
- k potlačení logiky a specifikaci výstupní hodnoty pro jiný procesor (např. vytvořeného tagu),
- k potlačení logiky a specifikaci stavu výstupního prvku.

DŮLEŽITÉ UPOZORNĚNÍ

Použití pevných nastavení hodnot prodlužuje dobu vykonávání logiky. Čím více hodnot je pevně nastaveno, tím déle bude trvat vykonání logiky.

DŮLEŽITÉ UPOZORNĚNÍ

Pevná nastavení I/O jsou uložena na procesoru, nikoli na programovací pracovní stanici. Zůstávají v platnosti, i když je programovací stanice odpojena.

Při pevném nastavení hodnot I/O:

- Můžete nastavit všechna data I/O kromě konfiguračních dat.
- Pokud je příslušný tag polem nebo strukturou (např. tag I/O), můžete pevně nastavit prvek nebo člen BOOL, SINT, INT, DINT nebo REAL.
- Pokud je hodnota SINT, INT nebo DINT, můžete pevně nastavit celou hodnotu nebo její jednotlivé bity. Individuální bity mohou mít force status:
 - No force
 - Force on
 - Force off
- Lze také pevně nastavit alias člena struktury I/O a vytvořeného nebo přijatého tagu.
 - Alias tag má stejnou hodnotu jako jeho základní tag, takže pevné nastavení alias tagu způsobí i pevné nastavení základního tagu.
 - Odstranění pevného nastavení alias tagu odstraní pevné nastavení i na příslušném základním tagu.

Pevné nastavení vstupní hodnoty

Pevné nastavení vstupního nebo přijatého tagu:

- nastaví hodnotu bez ohledu na hodnotu fyzického modulu nebo vytvořeného tagu,
- nemá vliv na hodnotu přijatou ostatními procesory, které monitorují stejný vstup nebo vytvořený tag.

Pevné nastavení výstupní hodnoty

Pevné nastavení výstupního nebo vytvořeného tagu potlačí logiku pro příslušný fyzický modul nebo ostatní procesory. Ostatní procesory monitorující příslušný výstupní modul prostřednictvím listen-only spojení rovněž získají pevně nastavenou hodnotu.

Přidání pevného nastavení hodnot I/O

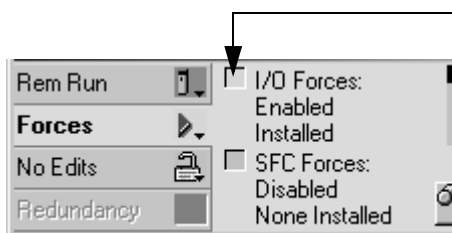
Pro pevné nastavení vstupní hodnoty, výstupní hodnoty, vytvořeného tagu nebo přijatého tagu používáme I/O force:

POZOR



Použití pevného nastavení může zapříčinit neočekávaný pohyb strojů a vybavení, které může zranit obsluhu. Před použitím této techniky zjistěte, jak budou stroje a procesy reagovat, a ujistěte se, že se v jejich blízkosti nenacházejí lidé.

- Povolení pevného nastavení zapříčiní změnu vstupních, výstupních, produkovaných nebo použitých hodnot.
- Pokud jsou pevná nastavení povolena a nějaké zadáte, jeho efekt bude okamžitý.



1. Stav indikátoru I/O Forces

Stav indikátoru	Význam
Nesvítí	V současnosti nejsou přítomna žádná pevná nastavení I/O.
Bliká	V současnosti nejsou žádná pevná nastavení I/O aktivní. Nicméně v projektu existuje alespoň jedno pevné nastavení. Při povolení pevných nastavení I/O forces budou provedeny všechny existující I/O forces.
Svítí	Pevná nastavení I/O jsou aktivní (povolena). Pokud přidáte pevné nastavení, okamžitě bude použito.

2. Otevřete rutinu obsahující tag, který chcete pevně nastavit.

3. Klikněte na tag pravým tlačítkem myši a vyberte možnost Monitor... Pokud je to nutné, tag rozbalte a zobrazte hodnotu, kterou chcete pevně nastavit (např. hodnotu BOOL tagu DINT).

4. Vložte pevnou hodnotu:

Pro vložení pevné hodnoty	Proveďte
Hodnoty BOOL	Klepněte pravým tlačítkem myši na tag a vyberte položku Force ON nebo Force OFF.
Nebooleovské hodnoty	Do sloupce Force Mask příslušného tagu vložte hodnotu, kterou chcete nastavit. Poté stiskněte klávesu Enter.

5. Jsou pevná nastavení I/O povolena? (Viz část , krok 1.)

Pokud	Proveďte
Ne	V nabídce Logic vyberte položku I/O Forcing > Enable All I/O Forces. Poté zadejte položku Yes.
Ano	Stop.

Odstranění nebo vypnutí pevných nastavení hodnot

POZOR

Jakákoli změna pevných nastavení může zapříčinit neočekávaný pohyb strojů a vybavení, které může zranit obsluhu. Před zákazem nebo odstraněním forces zjistěte, jak budou stroje a procesy reagovat, a ujistěte se, že se v jejich blízkosti nenacházejí lidé.

Pokud chcete	a zároveň	Proveďte
Zrušit určité pevné nastavení	Ponechat ostatní pevná nastavení zapnutá	Odstraňte jedno pevné nastavení
Zrušit všechna pevná nastavení I/O, ale ponechat pevná nastavení SFC	Ponechat pevná nastavení I/O v projektu Odstranit pevná nastavení I/O z projektu	Zakažte všechna pevná nastavení I/O Odstraňte všechna pevná nastavení I/O

Odstranění jednoho pevného nastavení

POZOR

Pokud odstraníte jedno pevné nastavení, ostatní zůstanou zapnutá a jakákoli další přidaná force bude hned účinná.

Před odstraněním pevného nastavení zjistěte, jak budou stroje a procesy reagovat, a ujistěte se, že se v jejich blízkosti nenacházejí lidé.

1. Otevřete rutinu obsahující pevné nastavení, které chcete odstranit.
2. V jakém jazyce je rutina napsána?

Jazyk rutiny	Postup
SFC	Přejděte na krok 4.
Logika ladder	Přejděte na krok 4.
Diagram funkčních bloků	Přejděte na krok 3.
Strukturovaný text	Přejděte na krok 3.

3. Klepněte pravým tlačítkem myši na pevně nastavený tag a vyberte možnost Monitor... Pokud je to nutné, tag rozbalte a zobrazte pevně nastavenou hodnotu, např. hodnotu BOOL tagu DINT.
4. Klepněte pravým tlačítkem myši na tag nebo prvek, pro který je nastavena force, a vyberte možnost Remove Force.

Zákaz všech I/O Forces

V nabídce Logic vyberte položku I/O Forcing > Disable All I/O Forces. Poté zadejte položku Yes.

Odstranění všech I/O Forces

V nabídce Logic vyberte položku I/O Forcing > Remove All I/O Forces. Poté zadejte položku Yes.

Jak nám to jde?

Vaše komentáře k našim technickým publikacím nám pomohou poskytovat lepší služby.
Děkujeme za váš čas.

Vyplňte prosím tento formulář a odešlete jej e-mailem na adresu
RADocumentComments@ra.rockwell.com, popřípadě faxem nebo poštou.

Název /typ publikace	Procesory Logix5000: vstupní a výstupní data a data tagů						
Kat. č.	1756 ControlLogix, 1768 CompactLogix, 1789 SoftLogix, 1794 FlexLogix, PowerFlex 700S s DriveLogix	Č. pub.	1756-PM004B-CS-P	Datum pub.	Červenec 2008	Č. dílu	None (Žádný)

Vyplňte prosím následující informace. Označte pořadí (1 = potřeba zlepšení, 2 = v pořádku, 3 = skvělé).

Celková užitečnost	1	2	3	Jak bychom mohli tuto publikaci udělat ještě užitečnější?
Kompletnost (obsahuje všechny potřebné informace)	1	2	3	Je potřeba nějaké informace doplnit?
				postup/krok ilustrace vlastnost
				příklad pravidlo ostatní
				vysvětlení definice
Technická přesnost (všechny informace jsou správné)	1	2	3	Můžeme být přesnější?
				text ilustrace
Srozumitelnost (všechny informace jsou jasné)	1	2	3	Jak můžeme zlepšit srozumitelnost?
Další komentáře				Na zadní straně tohoto formuláře můžete vložit další komentáře.

Vaše jméno _____
Vaše funkce _____
Místo/telefon _____

Přejete si, abychom vás kontaktovali?
___ Ne, nemusíte mne kontaktovat
___ Ano, kontaktujte mne
___ Ano, pošlete mi e-mail na _____
___ Ano, kontaktujte mne na adrese _____

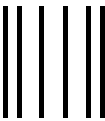
Formulář odešlete na adresu: Rockwell Automation Technical Communications, 1 Allen-Bradley Dr., Mayfield Hts., OH 44124-9705

Fax: 440-646-3525 E-mail: RADocumentComments@ra.rockwell.com

ZDE UPEVNIT (NESEŠÍVAT)

Další komentáře

ZDE PŘELOŽIT



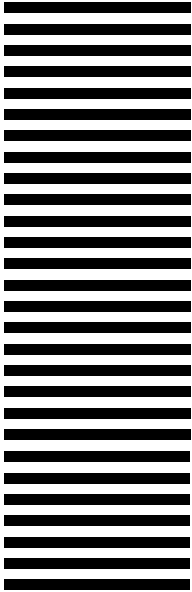
V RÁMCI
USA
BEZ
POŠTOVNÉHO

ODSTRANIT

BUSINESS REPLY MAIL
FIRST-CLASS MAIL PERMIT NO. 18235 CLEVELAND OH

**Rockwell
Automation**

**1 ALLEN-BRADLEY DR
MAYFIELD HEIGHTS OH 44124-9705**



Podpora společnosti Rockwell Automation

Společnost Rockwell Automation poskytuje na Internetu technické informace týkající se používání jejích produktů. Na adrese <http://support.rockwellautomation.com> najdete technické příručky, databázi znalostí a často kladené otázky, technické a aplikační poznámky, příklady kódů a odkazy na softwarové a servisní balíčky i funkci MySupport, kterou můžete přizpůsobit svým potřebám.

Technická telefonická podpora pro instalaci, konfiguraci a řešení problémů je nabízena v rámci programů TechConnect. Další informace vám poskytne distributor nebo zástupce společnosti Rockwell Automation; naleznete je také na adrese <http://support.rockwellautomation.com>.

Pomoc s instalací

Pokud se během prvních 24 hodin po instalaci vyskytnou problémy, projděte si informace z této příručky. Můžete se také obrátit na zákaznickou podporu.

USA	1.440.646.3434 Po–pá 8:00–17:00 EST
Mimo USA	Kontaktujte místního zástupce Rockwell Automation.

Vracení nových produktů

Společnost Rockwell Automation všechny své produkty testuje, aby se ubezpečila, že při expedici z výroby jsou plně funkční. Pokud se stane, že některý produkt není funkční a je potřeba jej vrátit, postupujte následujícím způsobem.

USA	Obraťte se na svého distributora. Budete požádáni o číslo případu zákaznické podpory (Customer Support case), které získáte na výše uvedeném tel. čísle.
Mimo USA	Obraťte se na místního zástupce společnosti Rockwell Automation.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444

Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640

Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publikace 1756-PM004B-CS-P – červenec 2008

Nahrazuje publikaci 1756-PM004A-CS-P – Červenec 2007

Copyright © 2008 Rockwell Automation, Inc. Všechna práva vyhrazena. Vytisknuto v USA.