

PLC

Základy programování

Tento kurz je určen účastníkům, kteří budou poprvé vytvářet řídicí programy pro programovatelné kontroléry.

Tento kurz vysvětluje programování, které je možno použít pro programovatelné kontroléry MELSEC. Jedním z hlavních programovacích jazyků je žebříkový diagram (LD). Tento kurz obsahuje základní body žebříkového programování včetně jeho hlavních povelů.

Některé kapitoly v tomto kurzu jsou založeny na základních kurzech programovatelných kontrolérů MELSEC. Před zahájením tohoto kurzu je doporučováno absolvovat základní kurz.

Obsah tohoto kurzu je následující.

Kapitola 1 – Programování PLC

Tato kapitola vysvětluje základní body žebříkového programování.

Kapitola 2 – Instrukce bitových proměnných

Tato kapitola vysvětluje instrukce obsahující bitové proměnné (ON/OFF).

Kapitola 3 – Instrukce datových proměnných

Tato kapitola vysvětluje instrukce obsahující datové (numerické) proměnné.

Kapitola 4 – Instrukce s rozvětvenými programy

Tato kapitola vysvětluje instrukce, které vytvářejí rozvětvené programy.

Závěrečný test

Známka pro úspěšné absolvování: Musí být 60% nebo vyšší.

Úvod Používání tohoto elektronického výukového nástroje



Přechod na následující stránku		Přejdete na následující stránku.
Zpět na předchozí stránku		Zpět na předchozí stránku.
Přesunutí na požadovanou stránku		Zobrazí se „Obsah“, pomocí kterého můžete přejít na požadovanou stránku.
Ukončit výuku		Ukončíte výuku.

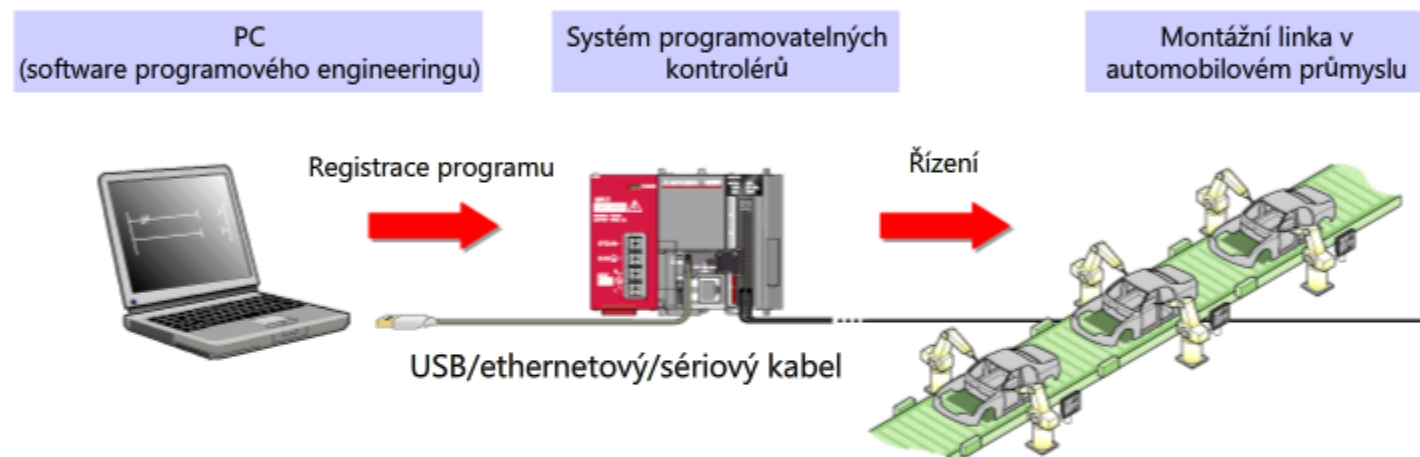
Bezpečnostní opatření

Pokud se učíte za použití skutečných produktů, přečtěte si prosím pečlivě bezpečnostní upozornění v příslušných návodech.

Kapitola 1 Řídicí program

Operace prováděné programovatelným kontrolérem se zapisují jako řídicí programy. Tyto programy jsou zaregistrovány do CPU modulu a ovládají různé vstupní a výstupní (I/O) signály. K programovacím jazykům, které se používají pro programovatelné kontroléry, patří žebřík, seznam instrukcí (IL) a sekvenční funkční diagram (SFC).

Tento kurz obsahuje základní body žebříkového programování včetně jeho hlavních instrukcí.



V tomto kurzu se k vytváření programů používá engineeringový software pro programovatelné kontroléry GX Works2 nebo GX Works3.

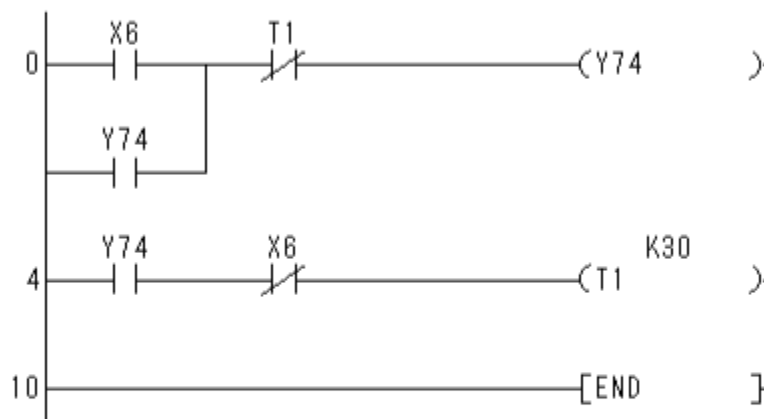
Abyste získali znalosti o používání engineeringového softwaru, absolvujte kurz „GX Works2 Basics“ (GX Works2 – základy) nebo „Engineering Software MELSOFT GX Works3 (Ladder)“ (Engineeringový software MELSOFT GX Works3 (žebříkový)).

GX Works2 podporuje kontroléry řady MELSEC-Q/L/F.

GX Works3 podporuje kontroléry řady MELSEC iQ-R/iQ-F.

K programovacím jazykům, které se používají pro programovatelné kontroléry, patří žebřík, seznam instrukcí (IL) a sekvenční funkční diagram (SFC).

Žebříkový program je grafický logický diagram založený na elektrickém obvodu. V žebříkových programech jsou symboly zastupující instrukce spojeny s vedeními, podobně jako u okruhového schématu, a toky operací jsou snadno rozpoznatelné. Navíc žebříkové programování nevyžaduje speciální znalosti programování, jako je tomu u jazyků C a BASIC, a je snadno pochopitelné i pro ty, kteří mají zkušenosti z práce s elektrickými obvody a relé.



V níže uvedené tabulce je zobrazen stejný program v IL.

IL vyžaduje pro vyjádření operací v podobě instrukcí určité znalosti programování.

Č. kroku	Instrukce	Proměnná
0	LD	X6
1	OR	Y74
2	ANI	T1
3	OUT	Y74
4	LD	Y74
5	ANI	X6
6	OUT	T1 K30
10	END	

Programy pro programovatelné kontroléry jsou schopny pracovat se dvěma typy hodnot.

Bit	Bit je vyjadřován ve dvou typech elektrických signálů, ON a OFF. Ty mohou být vyjádřeny i jako „1“ (ON) a „0“ (OFF). Bitové hodnoty se často používají pro zobrazení stavů I/O proměnných, např. vypínačů a kontrolék.
Slovo	Číslo a znaky. Slovní hodnoty se často používají pro zobrazení množství a času. *V tomto kurzu budou vysvětlena pouze čísla. Podrobnosti ohledně znaků používaných jako slovní hodnoty najdete v návodu k příslušnému produktu.

K zobrazení hodnot se používají následující numerické formáty.

- Decimální
- Binární
- Hexadecimální
- Oktální

1.2.1**Decimální zápis**

V decimálním zápisu je velikost (kvantita) čísla znázorněna za použití základního formátu 10 od „0 do 9“.

U programovatelných kontrolérů MELSEC předchází před decimálními čísly písmeno „K“.
Například „K153“ zastupuje decimální číslo „153“.

1.2.2

Binární zápis

Zatímco decimální zápis se obvykle používá k vyjadřování množství a času, používají programovatelné kontroléry a osobní počítače binární data, což jsou kombinace „0“ a „1“.

V níže uvedené tabulce je znázorněno, jak si odpovídají decimální a binární hodnoty, až do decimálního čísla „8“.

Decimální	Binární
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
⋮	⋮

Kdekoli je v programu použita instrukce tvořená 1 slovem, uložil by ho programovatelný kontrolér jako 16bitový binární údaj. 16bitový binární údaj je synonymem pro „1 slovo“.

Například decimální číslo „157“ se v binární soustavě vyjádří jako „0000000010011101“.

V decimálním zápisu se bity píšou zprava. (Bit úplně vpravo je počáteční.)

b15	~	b8	b7	b6	b5	b4	b3	b2	b1	b0	
0	...	0	1	0	0	1	1	1	0	1	← Pozice bitu
2^{16}		2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	← Binární
32768		256	128	64	32	16	8	4	2	1	← Mocnina 2
											← Váha pro decimální hodnoty

Pro konvertování binárních hodnot na decimální vynásobte každý bitový stav („0“ nebo „1“) příslušnou váhou a všechny výsledky sečtěte.

$$\begin{aligned}
 &= \underline{1 \times 128} + 0 \times 64 + 0 \times 32 + \underline{1 \times 16} + 1 \times 8 + \underline{1 \times 4} + 0 \times 2 + \underline{1 \times 1} \\
 &= 128 + 16 + 8 + 4 + 1 \\
 &= 157
 \end{aligned}$$

Směrem shora je možno považovat binární hodnoty jako formát číslování založený na váhách.

1.2.3 Hexadecimální zápis

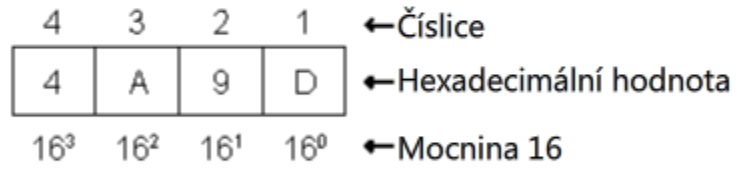
U hexadecimálního zápisu je velikost (kvantita) znázorněna za použití základního formátu 16, nebo je vyjádřena za použití 16 alfanumerických znaků: 0 až 9 a A až F. Každá číslice v hexadecimálním zápisu se zvyšuje jako 0, 1...9, A...E a pak F. Když hodnota dosáhne radixu „F“, je přenesena doleva a stává se z ní „10“.

U programovatelných kontrolérů MELSEC předchází před hexadecimálními čísly písmeno „H“. Například „H4A9D“ zastupuje hexadecimální číslo „4A9D“.

Binární zápis může být dlouhý a v programech a při znázornění na monitorech se obtížně používá.

V takovém případě je užitečný hexadecimální zápis. Jedna číslice hexadecimální hodnoty může vyjádřit 4 bity (4 číslice) binárních hodnot.

Na následujícím obrázku je vidět, jak je možno vyjádřit hexadecimální hodnotu jako decimální.



$$= 4 \times 16^3 + A \times 16^2 + 9 \times 16^1 + D \times 16^0$$

(4096) (10) (256) (16) (13) (1)

$$= 19101$$

*Jedna číslice hexadecimální hodnoty může vyjádřit 4 bity binárních hodnot.

Decimální	Binární	Hexadecimální
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

16	10000	10
17	10001	11
18	10010	12
⋮	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮
19101	0100 1010 1001 1101	4 A 9 D

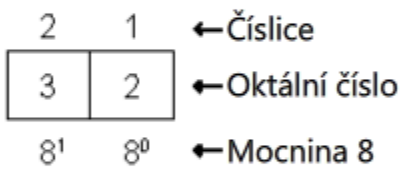
1.2.4 Oktální zápis

V oktálním zápisu je velikost (kvantita) čísla znázorněna za použití základního formátu 8 („0 až 7“).

Když se hodnota zvýší od „0“, „1“, „2“ do „7“, je přenesena doleva a stává se z ní „10“.

Tento oktální zápis se používá pro čísla I/O u řady MELSEC iQ-F/F.

Na následujícím obrázku je vidět, jak je možno vyjádřit oktální hodnotu jako decimální.



$$= 3 \times 8^1 + 2 \times 8^0$$

(8) (1)

$$= 26$$

*Jedna číslice oktálové hodnoty může vyjádřit 3 bity binárních hodnot.

Decimální	Binární	Hexadecimální	Oktalový
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
<hr style="border-top: 1px dashed black;"/>			
8	1000	8	10
9	1001	9	11
10	1010	A	12
11	1011	B	13
12	1100	C	14
16	1101	D	15
14	1110	E	16
15	1111	F	17
<hr style="border-top: 1px dashed black;"/>			
16	1000	10	20
17	10001	11	21
18	10010	12	22
.	.	.	.
.	.	.	.
.	.	.	.
26	011 010	1A	32

↑ ↑ ↑

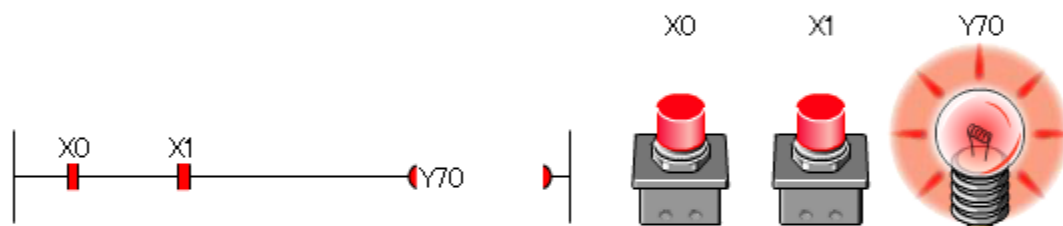
1.3 Základy programování PLC

V sekvenčním řízení je série operací prováděna na základě signálů ON/OFF přicházejících od zařízení připojeného k modulu vstupů a poté jsou výsledky operace vysílány k zařízení připojenému jako výstupní modul.

Pro provádění takového řízení musí mít řídicí program vstupní podmínky a výstupy, které se provádějí za uspokojivých vstupních podmínek.

Níže uvedený program dává povely pro tyto operace:

- Když jsou tlačítka připojená ke svorkám X1 a X2 obě ON, zapněte (ON) svorku Y70
- Výsledek operace je vyslán jako výstup ke svorce Y70, aby se rozsvítila (ON) připojená kontrolka



Současným stisknutím vypínačů X0 a X1 se zapne (ON) kontrolka Y70.

Programy popsané v kapitole 1.3 používaly k identifikaci I/O proměnných alfanumerické symboly jako X0, X1 a Y70. Tyto alfanumerické znaky se nazývají čísla I/O.

Tato kapitola obsahuje vysvětlení k číslům a proměnným I/O, která jsou zapotřebí pro vytváření řídicích programů.

Řady MELSEC iQ-R/Q/L/iQ-F a řada MELSEC-F používají pro vyjadřování čísel proměnných různé formáty.

V následující tabulce je uveden přehled rozdílů.

Řada MELSEC	Bit			Slovo
	X (číslo vstupu)	Y (číslo výstupu)	M (interní relé)	D (datový registr)
Řady iQ-R/Q/L	Hexadecimální	Hexadecimální	Decimální	Decimální
Řady iQ-F/F	Oktální	Oktální	Decimální	Decimální

1.4.1

Čísla I/O a signály I/O (Řady MELSEC iQ-R/Q/L)

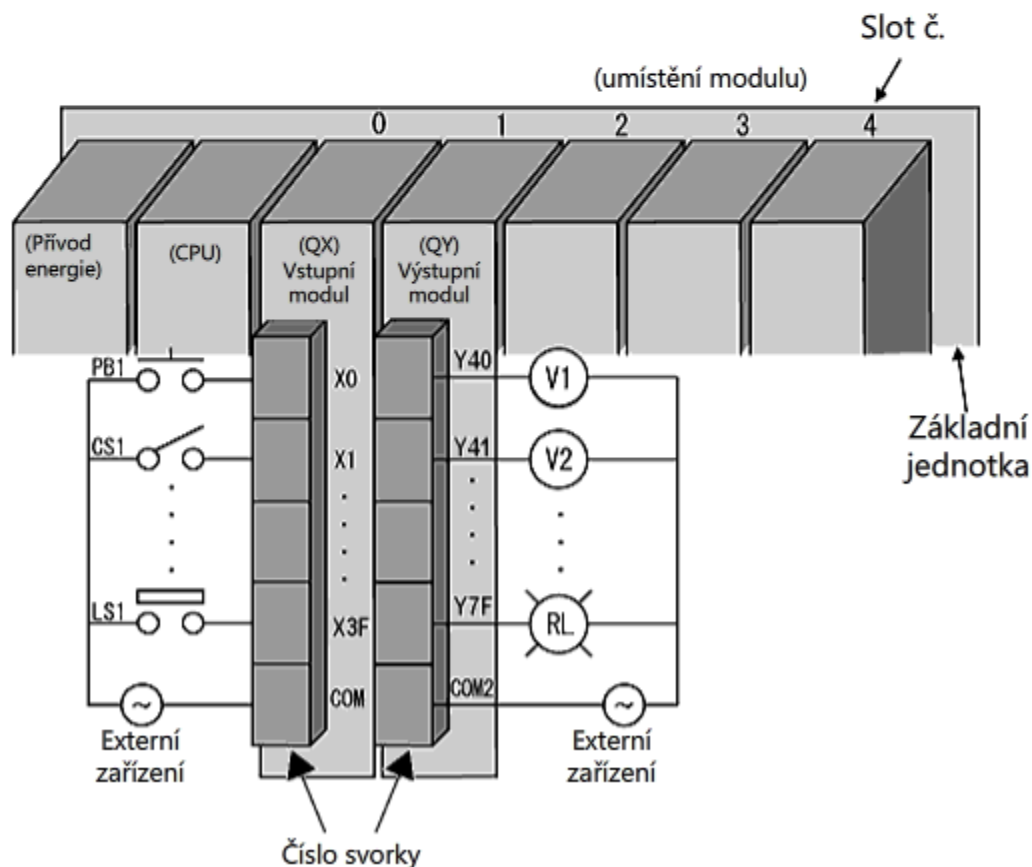
Řady MELSEC iQ-R/Q/L

Číslo I/O je tvořeno písmenem, které znamená vstup (X) nebo výstup (Y) a hexadecimální hodnotou, která znamená číslo svorky.

Číslo I/O je nejprve určováno na základě instalační pozice modulu.

Poté je určen rozsah čísel I/O na základě počtu obsazených I/O bodů na modulu.
(Počet obsazených I/O bodů je proporcionální k počtu svorek I/O modulu.)

Na následujícím obrázku je vidět, jak jsou čísla I/O přiřazována k 64bodovému vstupnímu modulu a 64bodovému výstupnímu modulu, které jsou nainstalovány na slotu č. 0 a č. 1.



1.4.2

Čísla I/O a signály I/O (Řady MELSEC iQ-F/F)

Řady MELSEC iQ-F/F

Číslo I/O je tvořeno písmenem, které znamená vstup (X) nebo výstup (Y) a oktální hodnotou, která znamená číslo svorky.

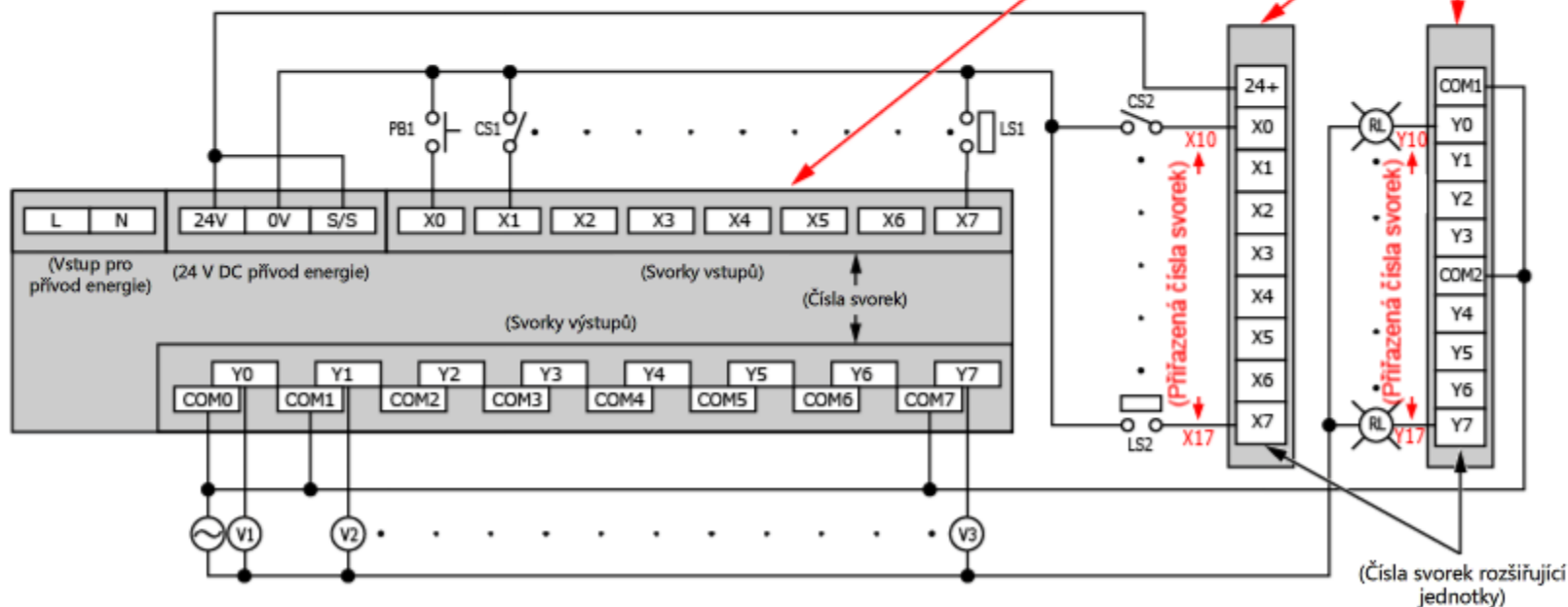
Nejprve se určí číslo I/O na základě koncového čísla I/O hlavní jednotky nebo předcházejícího rozšiřujícího bloku I/O. Poté je určen rozsah čísel I/O na základě počtu obsazených I/O bodů na jednotce.

(Počet obsazených I/O bodů je proporcionální k počtu I/O bodů obsazených rozšiřující jednotkou I/O.)

První číslice čísla I/O pro novou rozšiřující jednotku vždy začíná od „0“.

Například pokud číslo I/O předcházející jednotky končí na X7, bude číslo I/O následující jednotky začínat od X10.

Na následujícím obrázku je vidět, jak jsou čísla I/O přiřazována k 8bodové rozšiřující jednotce vstupů a 8bodové rozšiřující jednotce výstupů, které jsou přidány k hlavní jednotce řady MELSEC-F.



1.4.3 Čísla I/O a proměnné

Stavy vybavení připojeného k jednotce se ukládají do paměťové oblasti programovatelného kontroléru zvané „proměnná“.

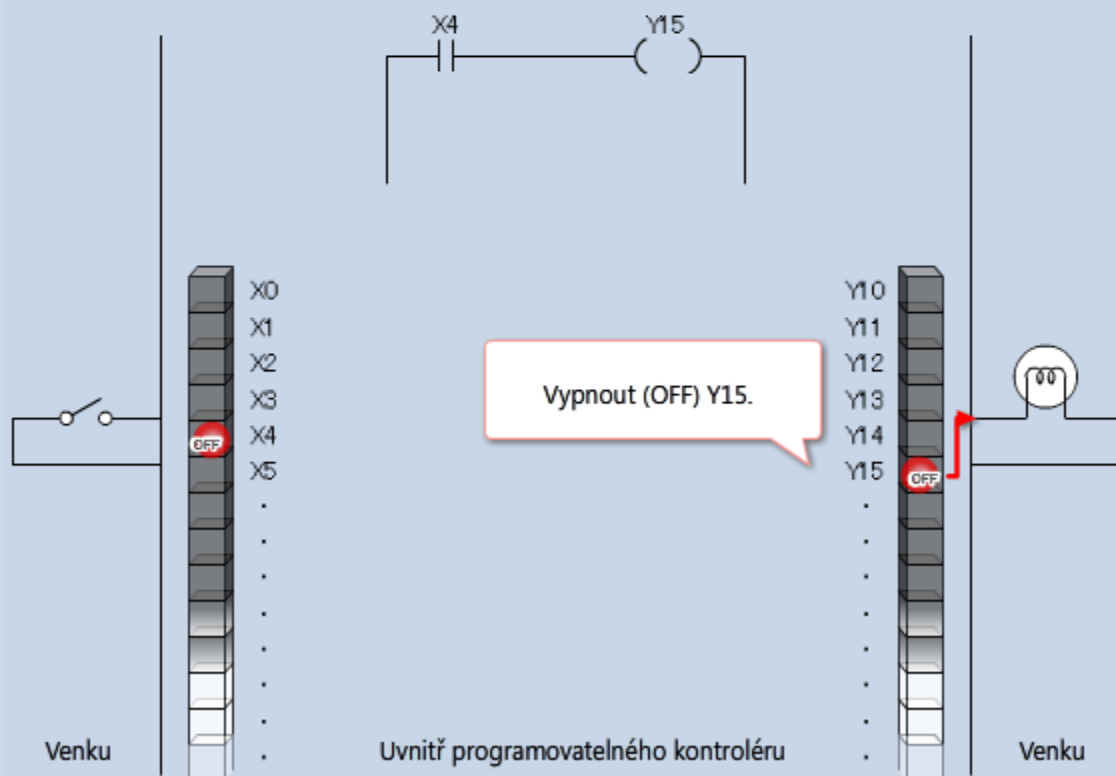
Podobně je tomu i u výstupu, výstupní vybavení pracuje v souladu se stavy těchto proměnných. Jak je vysvětleno výše, jsou řídicí programy často prováděny na základě stavů proměnných.

Proměnné, v nichž jsou uloženy bitové informace (ON/OFF), například vstupy (X) a výstupy (Y), se nazývají „bitové proměnné“.

Čísla proměnných odpovídají číslům I/O.

Například stav svorky, která má přiřazené číslo I/O X0, je uložen v proměnné X0.

Stejně tak stav proměnné Y10 odpovídá svorce, která má přiřazené číslo I/O Y10.



1.4.4

Interní relé

Teď již víme, že bitové proměnné, jako X (vstup) a Y (výstup), odpovídají číslům přiřazeným fyzickým svorkám modulu I/O.

Existuje ale i další skupina bitových proměnných, které nemají žádný vztah se svorkami modulu I/O, a jedna z nich se nazývají „interní relé (M)“.

Interní relé (M) se vyjadřují v decimálním formátu, i když vstupní (X) a výstupní (Y) proměnné se u řady MELSEC iQ-R/Q/L vyjadřují v hexadecimálním formátu a u řady MELSEC iQ-F/F v oktálním formátu.

Interní relé (M) se používají převážně k ukládání dočasných bitových dat.

Interní relé (M) se mohou používat například k ukládání výsledků výpočtů z určité operace, aby bylo možno je použít i na jiné příčce žebříku.

1.4.5 Datové proměnné

Víme, že proměnné, v nichž jsou uloženy bitové informace (ON/OFF), se nazývají „bitové proměnné“ a že proměnné, v nichž jsou uloženy slovní informace, se nazývají „datové proměnné“.

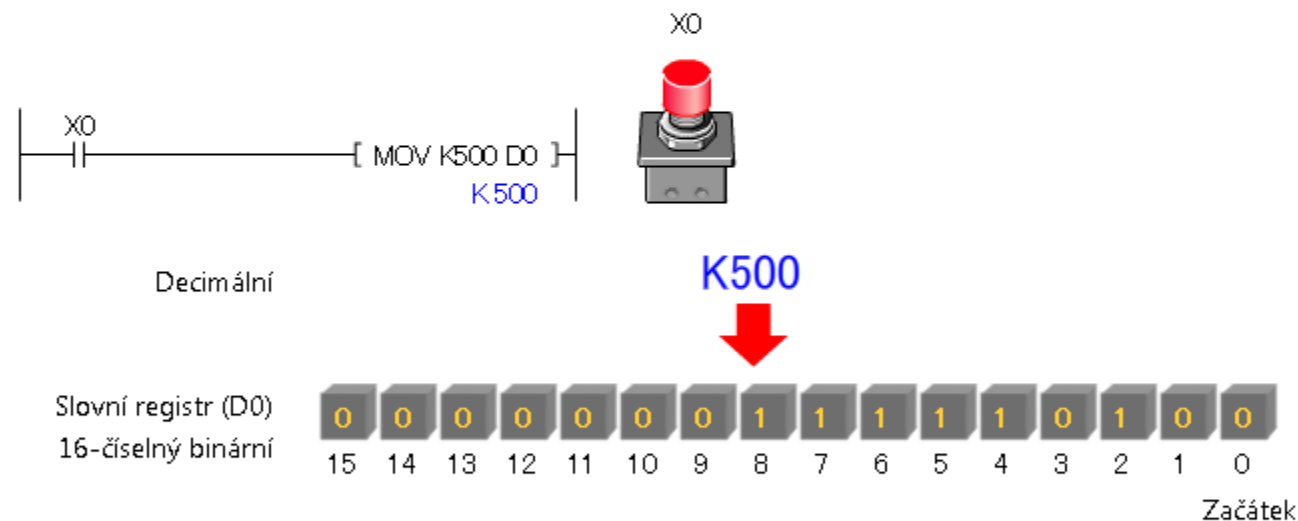
K typicky používaným datovým proměnným patří „datové registry“ (D). Do jednoho datového registru (D) je možno uložit 1 slovo (16 bitů) dat.

Následující animace znázorňuje, jak používat datové registry (D).

V programu se při zapnutí (ON) X0 uloží „K500“ (decimální hodnota) do D0.

Instrukce MOV zkopíruje hodnotu do specifikované proměnné. (Další podrobnosti budou vysvětleny v kapitole 3.1.)

V této kapitole jsou do datových registrů ukládána čísla.



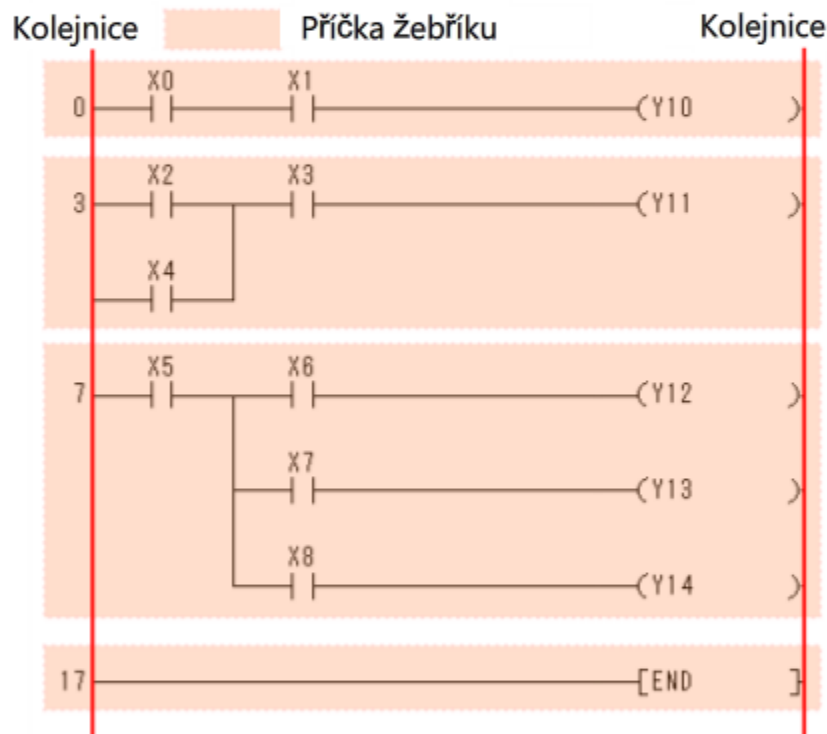
I když je tlačítko puštěno a X0 se vypne (OFF), „K500“ uložená v datovém registru D0 se zachová.

1.5 Vytváření řídicích programů

Řídicí program je tvořen kolejnicemi na levém a pravém okraji a symboly instrukcí spojenými čarami.

Oblast od povelu -| |- spojeného s levou kolejnicí po povel -()- nebo -[]- spojený s pravou kolejnicí se nazývá příčka žebříku.

Několik těchto příček žebříku tvoří ovládací program, který končí povellem -[END]- nebo -[FEND]- .



■ Rozdíl mezi engineeringovým softwarem

Povely -()- a -[]- mezi engineeringovým softwarem, GX Works2 a GX Works3 se liší. V tomto kurzu jsou popisovány povely GX Works2.

	GX Works2	GX Works3
-()-	(Y10)	Y10
-[]-	{MOV K500 D0 }	

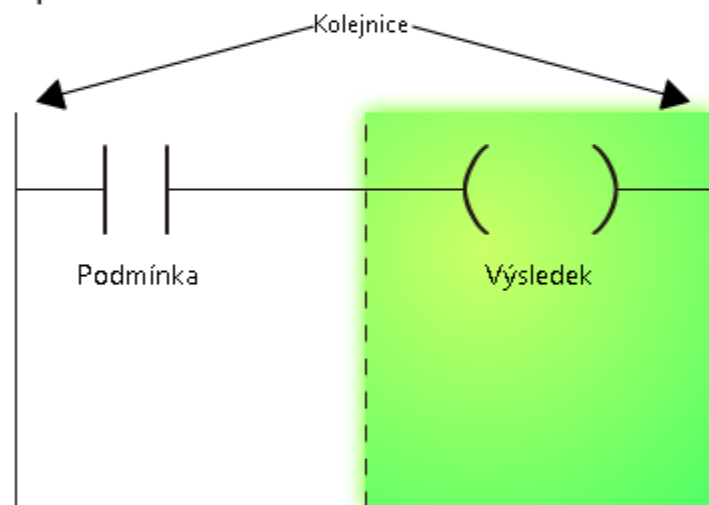
1.5.1 Symboly instrukcí

Jak je vysvětleno v kapitole 1.3, musí programovatelný kontrolér dostávat povely k provádění určitých operací, když je splněna předem specifikovaná podmínka vstupu. Pro takové povely se používají symboly instrukcí, které vyjadřují podmínky vstupů a podrobnosti výstupů.

Symbol instrukce často obsahuje číslo proměnné.

Číslo proměnné specifikuje oblast (proměnnou), kde je uložen určitý stav, který je použit pro posouzení podmínky nebo jako výstupní výsledek.

Příčka žebříku obsahuje podmínky a výstupní výsledky. Podmínky jsou umístěny vlevo a výstupní výsledky vpravo. Výstupní výsledek může být jednoduchý signál ON/OFF nebo dedikovaný povel jako například operace výpočtu nebo kopírování.



V žebříkovém diagramu jsou dvě rovnoběžné kolejnice.
Vlevo jsou napsané podmínky.
Vpravo jsou napsané výsledky.

1.6

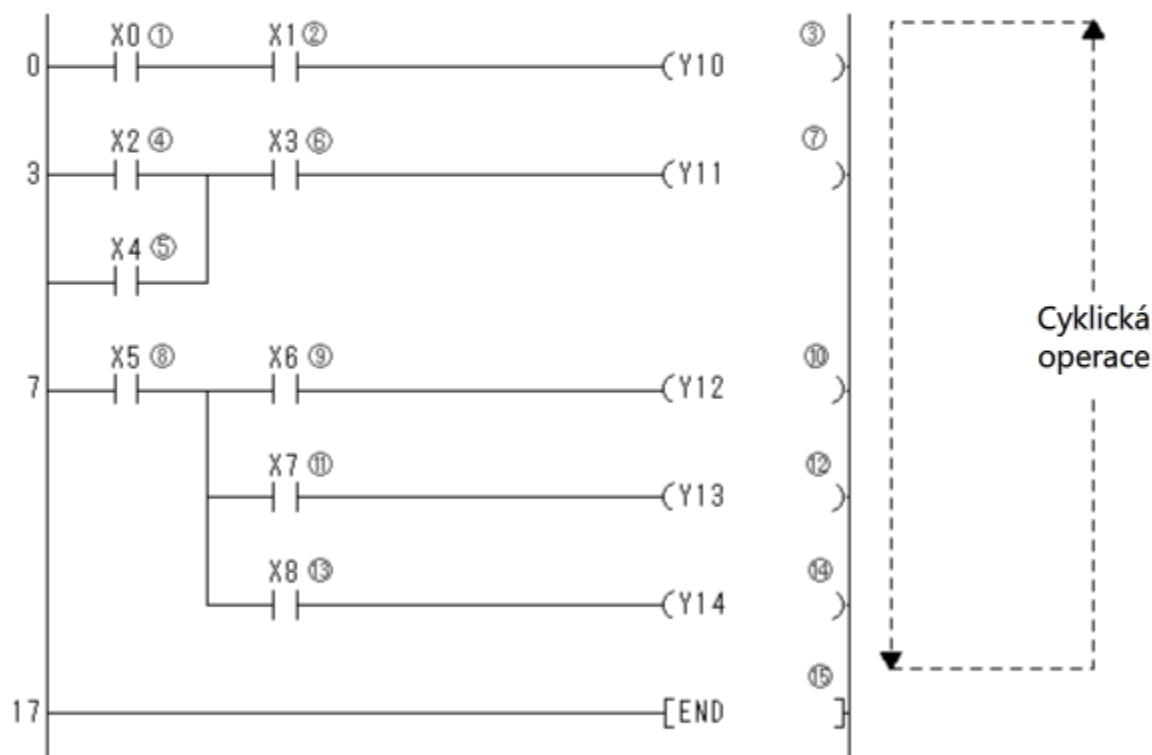
Postup provádění programu

Program začíná instrukcí k zahájení vlevo a končí instrukcí k ukončení -[END]-.
Po dosažení instrukce -[END]- začne provádění programu znovu od instrukce k zahájení.
Toto opakované provádění se nazývá „cyklická operace“.

Jeden cyklus této cyklické operace se nazývá „sken“ a období, které trvá zpracování jednoho skenu, se nazývá „čas skenování“.

Na následujícím obrázku je znázorněn postup provádění instrukcí.

Instrukce jsou prováděny zleva doprava na každé příčce žebříku a potom od horní příčky žebříku k dolní (č. 1, 2,...15 -> 1...).



1.7 Doba obnovy

Jak již bylo vysvětleno, čas skenování je období, které trvá provedení jedné série programů. Čas skenování je možno vyjádřit i takto:

$$\text{Čas skenování} = \text{čas obnovy} + \text{čas provádění programu} + \text{čas zpracování konce (END)}$$

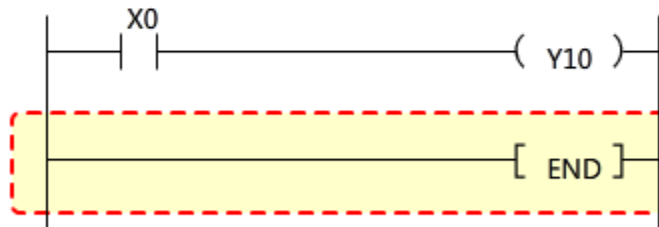
Čas obnovy je období, které trvá načtení dat z modulu vstupů do vstupních proměnných (X) plus období, které trvá zápis dat do výstupního modulu z výstupních proměnných (Y).



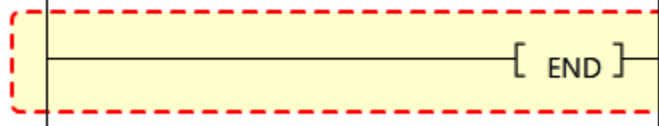
(1) Obnovení I/O

- Vysílá stav ON/OFF výstupních proměnných k připojenému výstupnímu vybavení
- Ukládá stav ON/OFF přijatý od připojeného vstupního vybavení do vstupních proměnných

(2) Provedení programu



(3) END zpracování



END zpracování instrukcí END
(Podrobnosti jsou zde vynechány.)

Upozorňujeme, že stavy ON/OFF vypínače jsou načteny najednou a uloženy do vstupních proměnných (X) a nová data přepíše během obnovy dosavadní hodnoty. Stejně tak data ve výstupních zařízeních (Y) jsou načtena najednou do výstupního modulu při provádění instrukce.

To znamená, že pokud se signál přepne z OFF do ON a pak znovu do OFF, není tento signál rozpoznán jako ON. Avšak čas skenování je velmi krátký ve srovnání s délkou signálu. Je neobvyklé, aby programovatelný kontrolér zmeškal změnu stavu signálu.

Kapitola 2 Instrukce bitových proměnných

Tato kapitola vysvětluje instrukce obsahující bitové proměnné (ON/OFF).

Operace, které používají bitové proměnné, jsou v řídicích programech ty nezákladnější operace. Vstupy ze vstupního vybavení se používají jako podmínky pro ovládání výstupního vybavení.

2.1

Vstupní podmínky a výstupy

Jako vstupní podmínky se používají normálně otevřené - spínací (NO) a normálně zavřené – rozpínací (NC) kontakty.

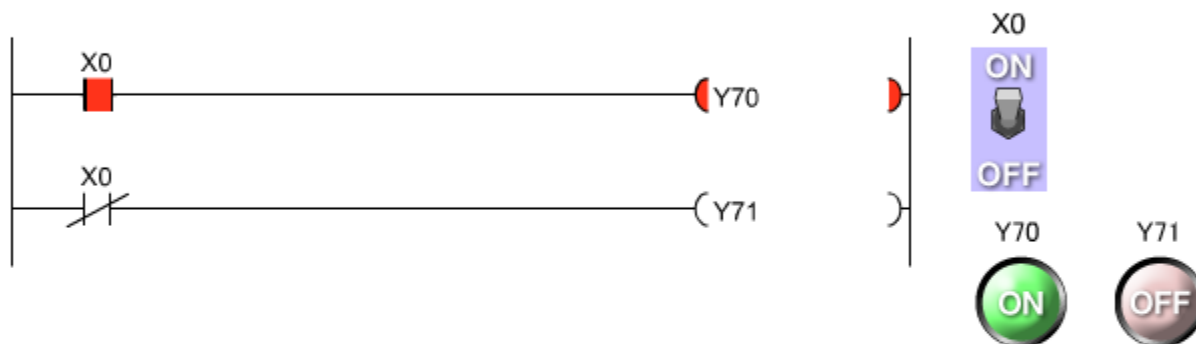
Když jsou vstupní podmínky splněny, je vydána instrukce výstupu cívky (instrukce OUT).

Když vstupní podmínky splněny nejsou, není instrukce výstupu cívky vydána.

Instrukce kontaktu NO/NC a instrukce OUT jsou hlavní kombinace instrukcí používané v řídicích programech.

■ Žebříkový program a operace

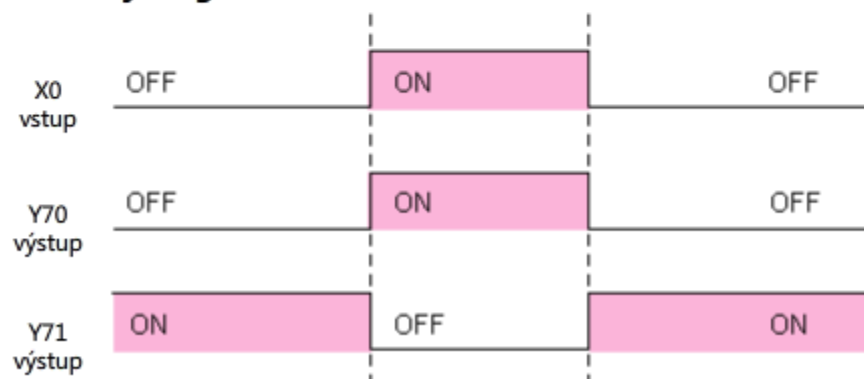
Simulujte operaci s instrukcemi NO, NC a OUT kliknutím na vstupní vypínač vpravo.



■ Instrukční kódy a funkce

Symbol	Funkce
	Kontakt NO Proveden, když je stav proměnné ON.
	Kontakt NC Proveden, když je stav proměnné OFF (opak kontaktu NO).
	Výstup cívky (OUT) Když je splněna předchozí vstupní podmínka, jsou vydána data v předem nastavené proměnné.
	Koncová instrukce (END) Označuje konec programu. Program vyžaduje koncovou instrukci END.

■ Časový diagram



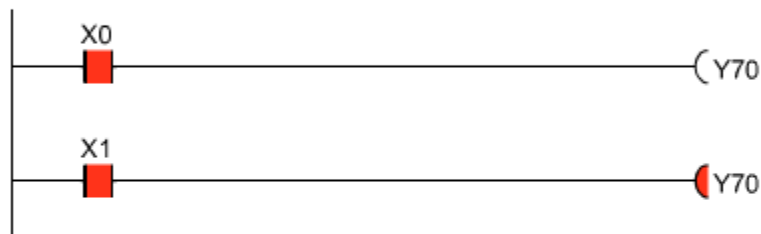
2.1.1

Použití stejného čísla proměnné pro instrukce

Na přičce žebříku může být s jedním číslem proměnné použit pouze jedna instrukce OUT. Pokud je použito více instrukcí OUT se stejným číslem proměnné, bude platná pouze poslední instrukce OUT, takže první instrukce OUT bude neplatná.

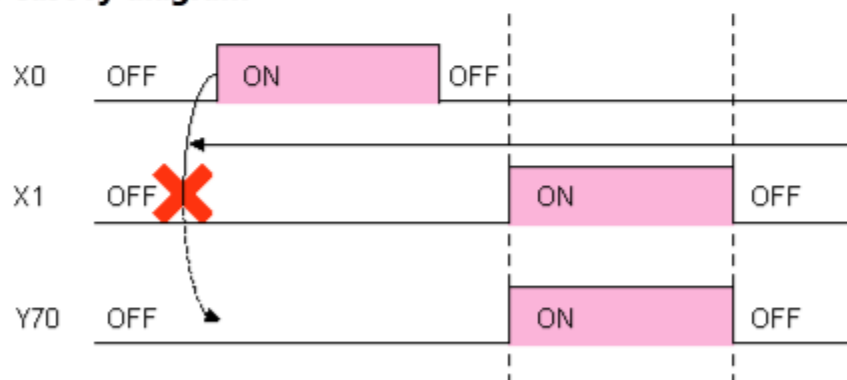
■ Žebříkový program

Simulujte operaci se dvěma instrukcemi se stejným číslem proměnné kliknutím na vstupní vypínač vpravo. Tento typ použití (používající OUT Y70 pro dvě instrukce) se nazývá „duplicitní cívka“.



Když X1=ON, Y70=ON.

■ Časový diagram



První vstupní podmínka X0 je ignorována, protože prioritu má poslední vstupní podmínka.

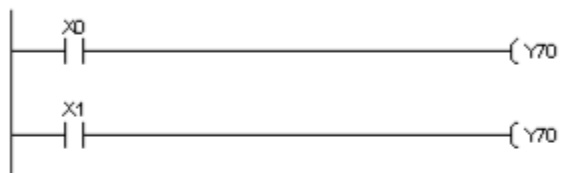
2.1.1

Použití stejného čísla proměnné pro instrukce

■ Příklad opravy

V tomto příkladu má vyšší prioritu vstupní podmínka „X1“ a „X0“ je ignorována.

Pokud opravíme příčku žebříku do podoby znázorněné na obr. B, proměnná Y70 se zapne (ON), když bude splněna kterákoli z obou vstupních podmínek, takže mezi těmito dvěma instrukcemi OUT nevznikne rozpor.



(obr. A)



(obr. B)

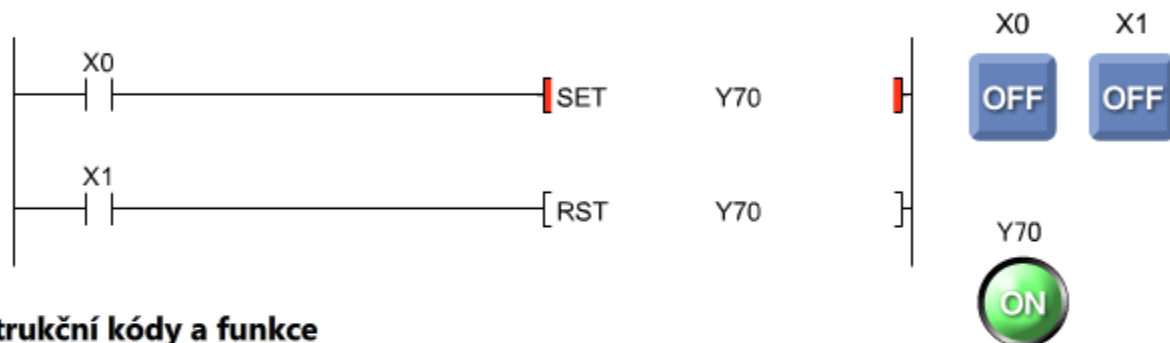
2.2

Setování / resetování výstupů

Na rozdíl od instrukce OUT udržuje instrukce k zachování operace (instrukce SET) výstupní stav, i když vstupní podmínka není splněna. Pro zrušení výstupu (OFF) může být provedena instrukce k zachování operace (RST).

■ Žebříkový program a operace

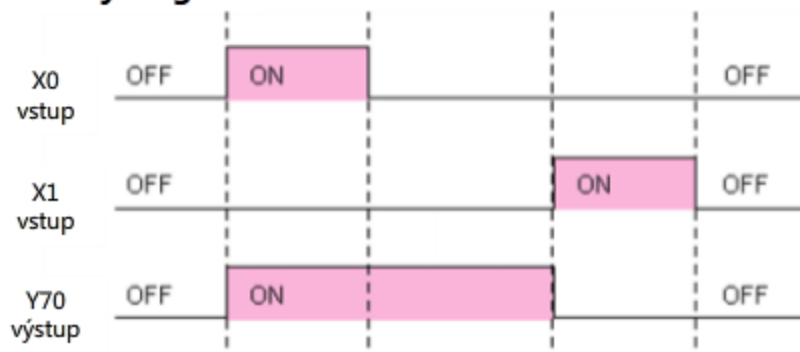
Simulujte operaci s instrukcemi SET a RST kliknutím na vstupní vypínač vpravo.



■ Instrukční kódy a funkce

Symbol	Funkce
	Instrukce k zachování operace (SET) Zapne proměnnou (ON) a zachová stav ON (výstupní). Výstup je zachován, i když vstupní podmínka není splněna.
	Instrukce ke zrušení zachování operace (RST) Zruší stav ON a zruší výstup na specifikované proměnné.

■ Časový diagram



2.2.1 Rozdíly mezi instrukcemi OUT a SET

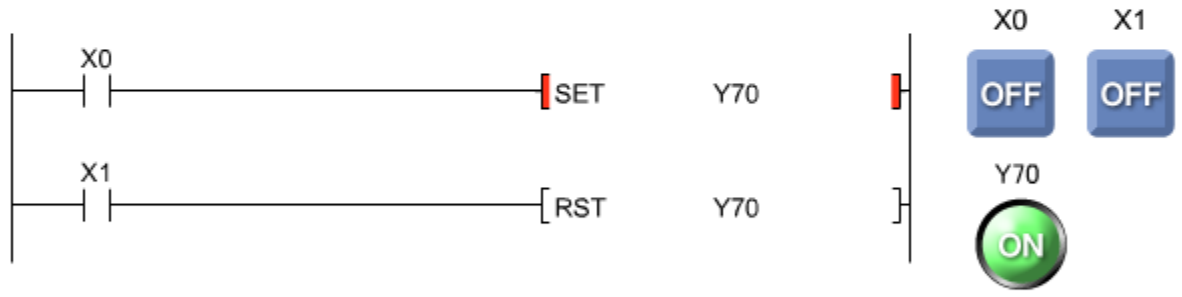
Simulujte rozdíly v operacích instrukcí OUT a SET kliknutím na vstupní vypínače vpravo.

■ Instrukce OUT



Y70 je ON, když je splněna vstupní podmínka.

■ Instrukce SET/RST



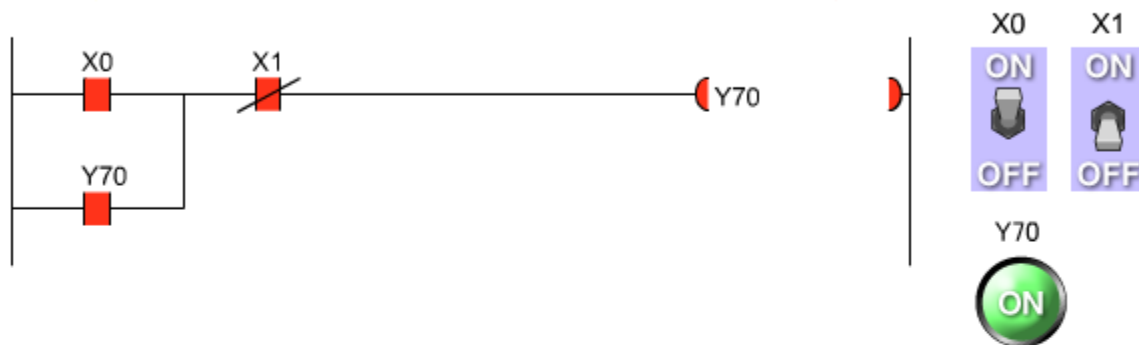
Jakmile je splněna vstupní podmínka, je Y70 ON, dokud není provedena instrukce RST.

2.2.2

Náhrada přídržných žebříků instrukcí SET

Žebříkový program a operace

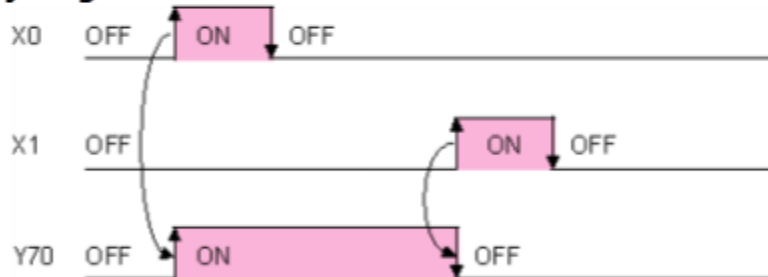
Simulujte operaci přídržného žebříku kliknutím na vstupní vypínače vpravo.



Když X0=ON a X1=OFF, Y70=ON.

Y70 = ON (zachování), dokud není X1=ON.

■ Časový diagram



I poté, co se X1 vypne (OFF), Y70 (cívka) zůstane ON (zachování)

■ Náhrada instrukcí SET

Program přídržného žebříku může být přepsán jako žebříkový program instrukcí SET.

Použitím instrukce SET může být žebříkový program zjednodušen.



2.3

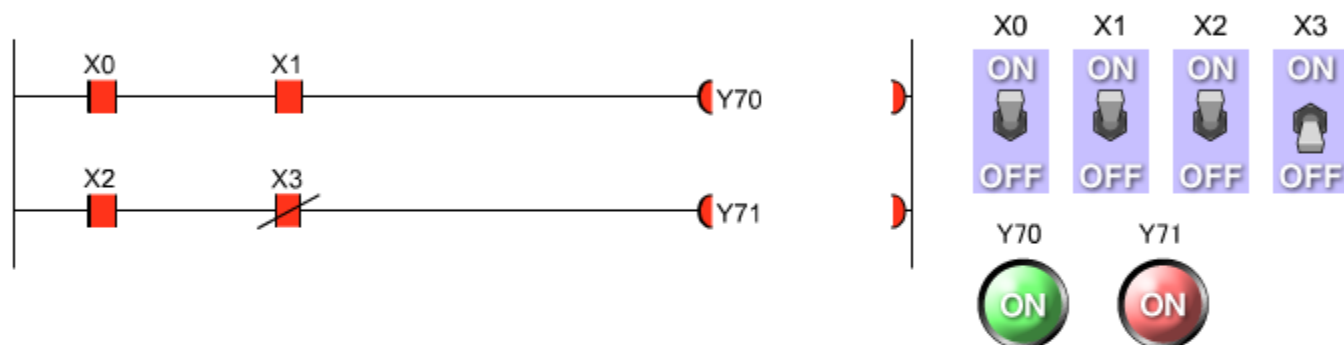
Přidávání podmínek (logika AND)

Aby vznikla logika AND (A), jsou kontakty NO/NC umístovány do sérií.

V logice AND je podmínka splněna, když je ON více kontaktů NO/NC, které jsou spojeny do série.

■ Žebříkový program a operace

Simulujte operaci s logikou AND kliknutím na vstupní vypínač vpravo.



Když X0 a X1 jsou ON, Y70 je ON.

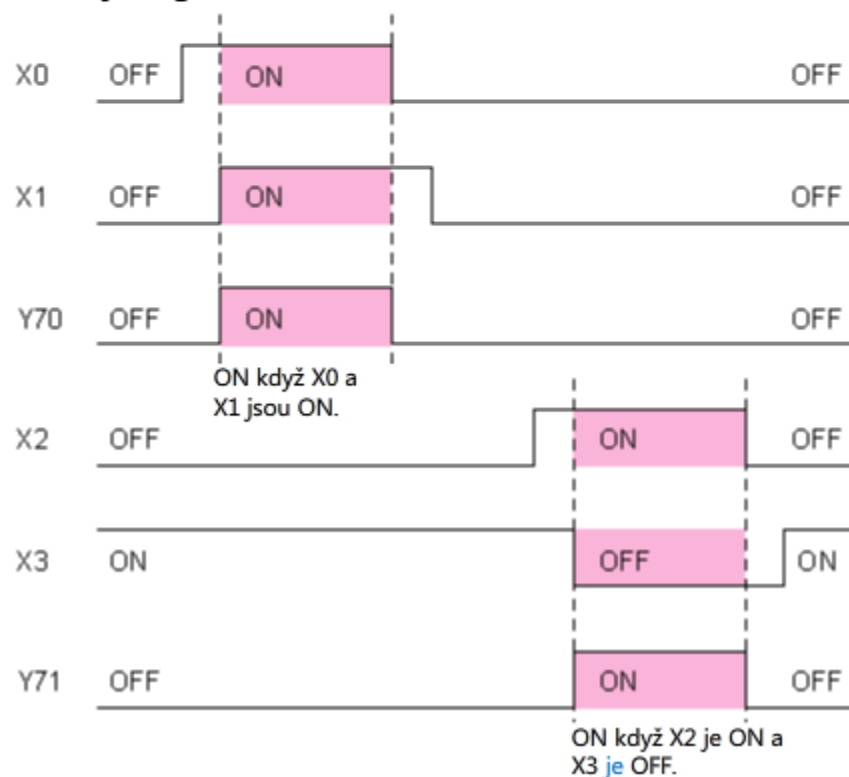
Když X2 je ON a X3 je OFF, Y71 je ON.

2.3 Přidávání podmínek (logika AND)

■ Instrukční kódy a funkce

Symbol	Funkce
	Sériové spojení kontaktu NO Kontakt NO je spojen v sérii (horizontálně).
	Sériové spojení kontaktu NC Kontakt NC je spojen v sérii (horizontálně).

■ Časový diagram



2.4

Přidávání podmínek (logika OR)

Aby vznikla logika OR (NEBO), jsou kontakty NO/NC umístěny paralelně.

V logice OR je podmínka splněna, když je ON jeden z kontaktů NO/NC, které jsou spojeny paralelně.

■ Žebříkový program a operace

Simulujte operaci s logikou OR kliknutím na vstupní vypínače vpravo.



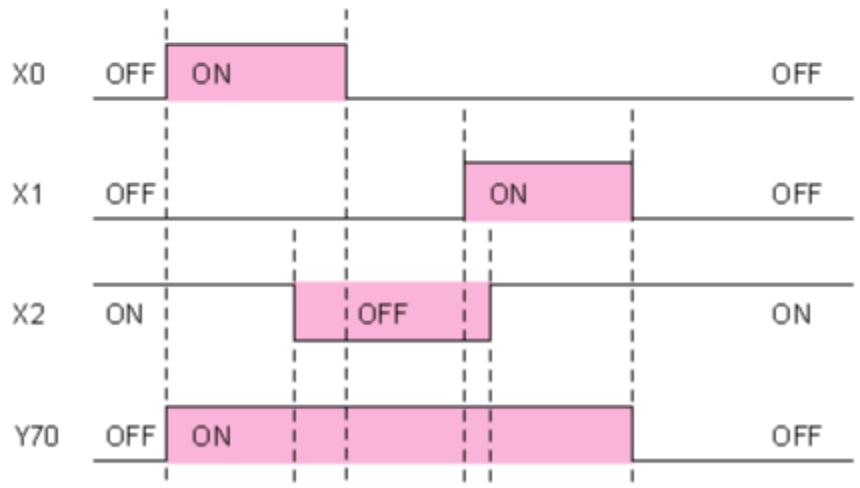
Y70 je ON, když je splněna kterákoli z těchto podmínek: X0 je ON, X1 je ON, nebo X2 je OFF.

2.4 Přidávání podmínek (logika OR)

■ Instrukční kódy a funkce

Symbol	Funkce
	<p>Paralelní spojení kontaktu NO Kontakt NO je spojen paralelně (vertikálně).</p>
	<p>Paralelní spojení kontaktu NC Kontakt NC je spojen paralelně (vertikálně).</p>

■ Časový diagram



2.5

Výstup v podobě pulsů

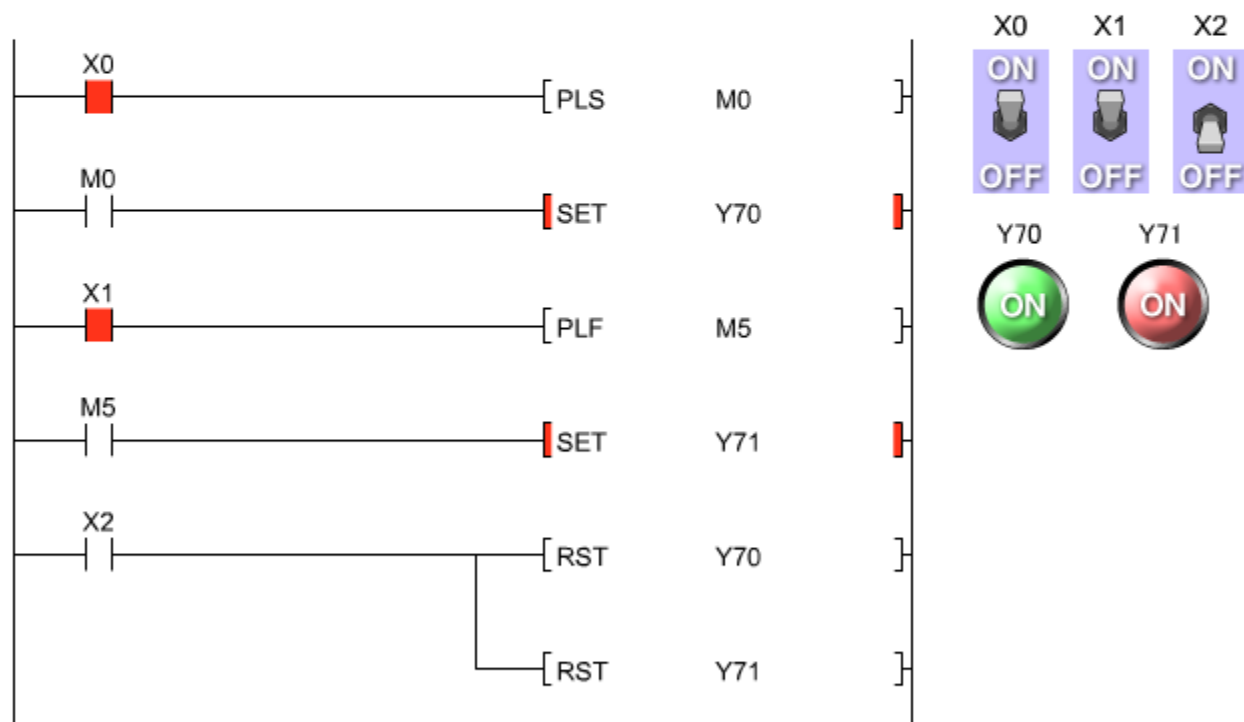
Na rozdíl od instrukce OUT způsobí instrukce na náběžné hraně (povel PLS) zapnutí cívky (ON) na jeden sken po splnění vstupní podmínky.

Na rozdíl od instrukce PLS způsobí instrukce na sestupné hraně (povel PLF) zapnutí cívky (ON) na jeden sken po nesplnění vstupní podmínky.

Cívka zapnutá (ON) povel PLS/PLF se vrátí do stavu OFF po jednom skenu.

■ Žebříkový program a operace

Simulujte operaci s instrukcemi PLS a PLF kliknutím na vstupní vypínač vpravo.



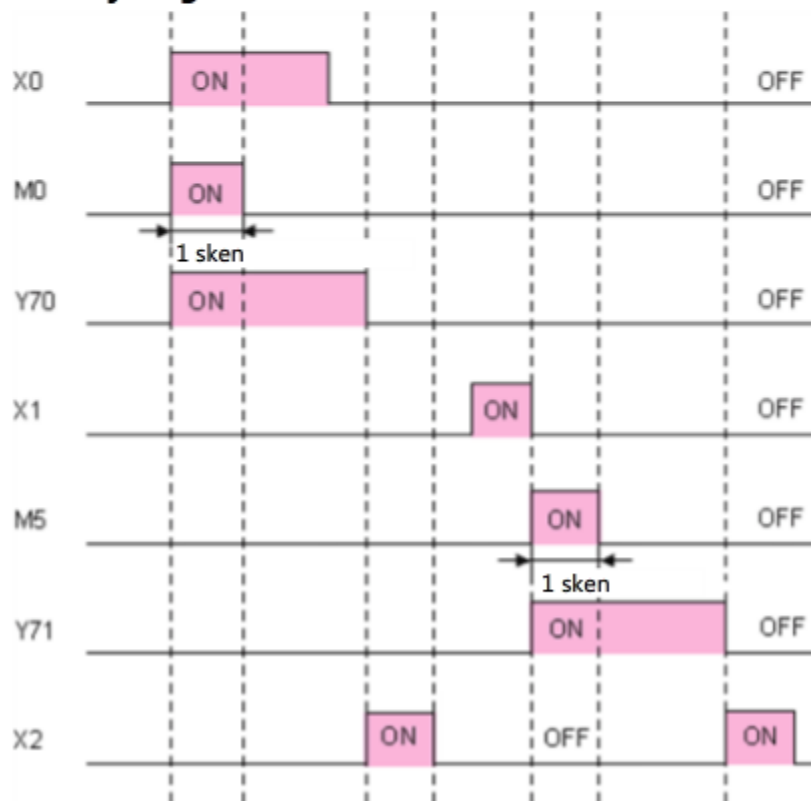
Na náběžné hraně X0 (OFF na ON) se M0 zapne (ON) na 1 sken
 Na sestupné hraně X0 (ON na OFF) se M5 zapne (ON) na 1 sken

2.5 Výstup v podobě pulsů

■ Instrukční kódy a funkce

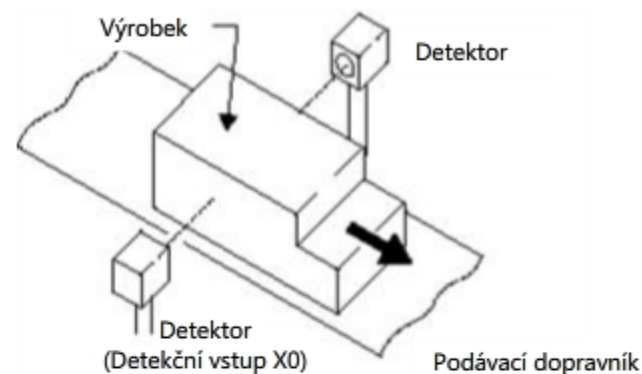
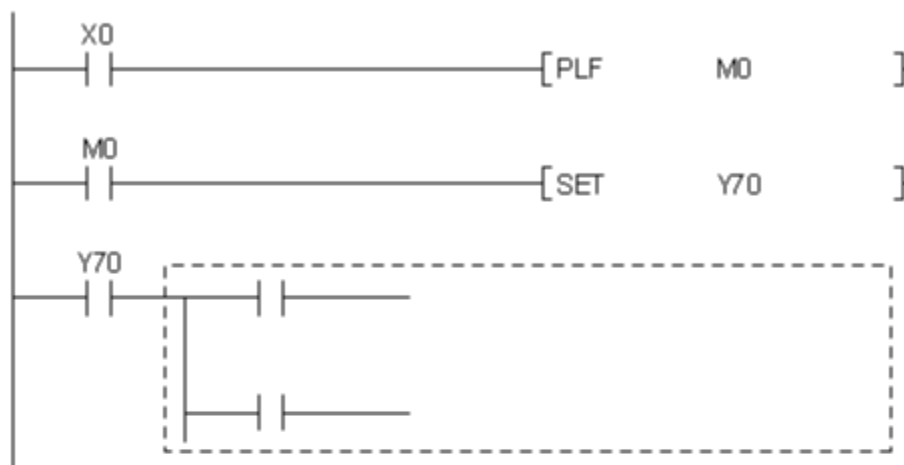
Symbol	Funkce
	Výstup na náběžné hraně (PLS) Data jsou vydána na specifikované proměnné při 1. skenu po splnění vstupní podmínky.
	Výstup na sestupné hraně (PLF) Data jsou vydána na specifikované proměnné při 1. skenu po nesplnění vstupní podmínky.

■ Časový diagram



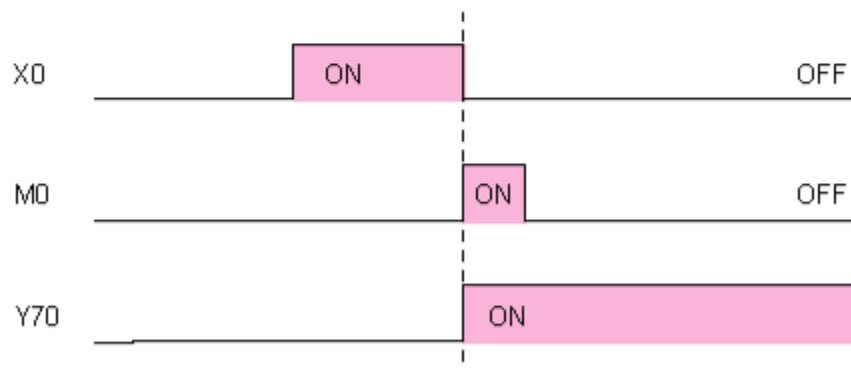
2.5.1 Příklad použití pulsních výstupů

■ Žebříkový program



Pulsní výstup je použit pro detekování průchodu pohybujících se objektů. Když je detekován průchod výrobků, je iniciován následný proces.

■ Časový diagram



2.6

Měření času

K měření času se používá povel OUT a proměnná zvaná časovač (T).

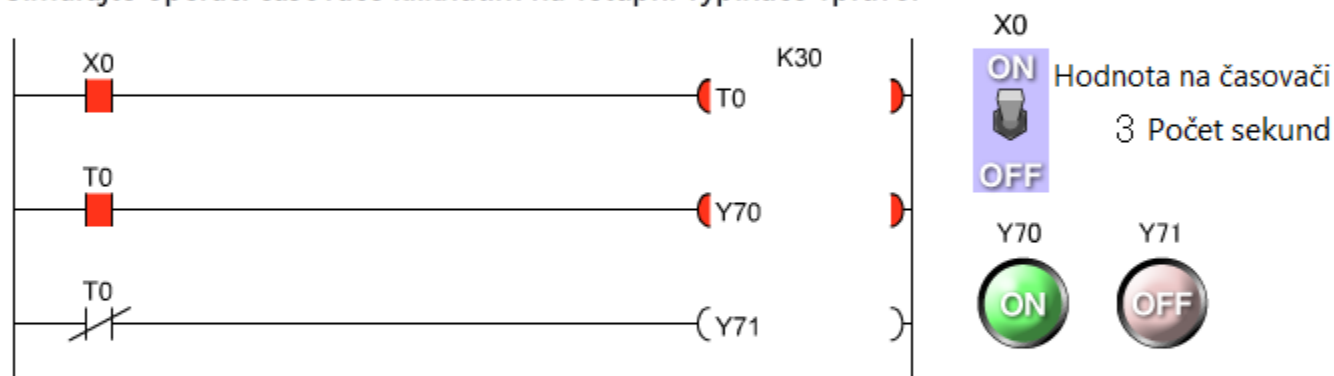
Když je splněna vstupní podmínka (ON), začne měření času. Když časové období dosáhne specifikované hodnoty, časovač (T) se zapne (ON).

Pokud vstupní podmínka není splněna (OFF), nebo je časovač (T) resetován повеlem RST, je inicializován uplynulý čas i výstup.

Stav časovače (T) je možno použít jako vstupní podmínku v jiných částech programu.

■ Žebříkový program a operace

Simulujte operaci časovače kliknutím na vstupní vypínače vpravo.

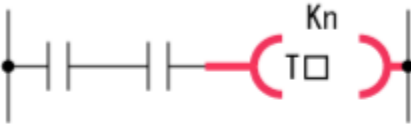


X0 se zapne (ON), pak po 3 sekundách se zapne (ON) Y70 a Y71 se vypne (OFF).

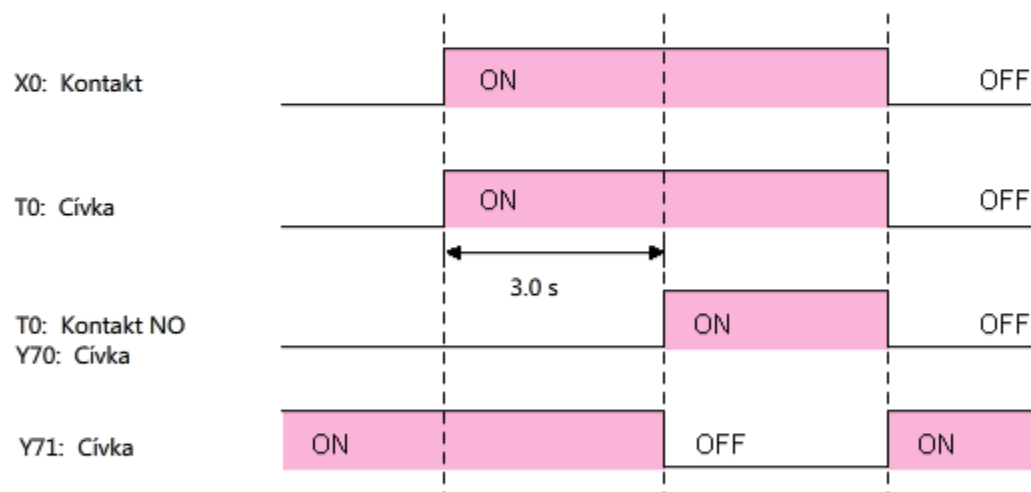
2.6

Měření času

■ Instrukční kód a funkce

Symbol	Funkce
 <p>□ : Číslo zařízení</p>	<p>Operace časovače</p> <p>Časovač (T) se používá s výstupem cívky (OUT) pro měření, jak dlouho je plněna podmínka (ON). Po specifikovaném časovém období nastane timeout.</p> <p>Souběžně s timeoutem se časovač (T0) zapne (ON). Nastavená hodnota časovače je označena „Kn“ (n: decimální).</p> <p>Časovače se často používají při prodlevě ke specifikaci času po splnění určité podmínky.</p>

■ Časový diagram



K čítání se používá instrukce OUT a proměnná zvaná čítač (C).

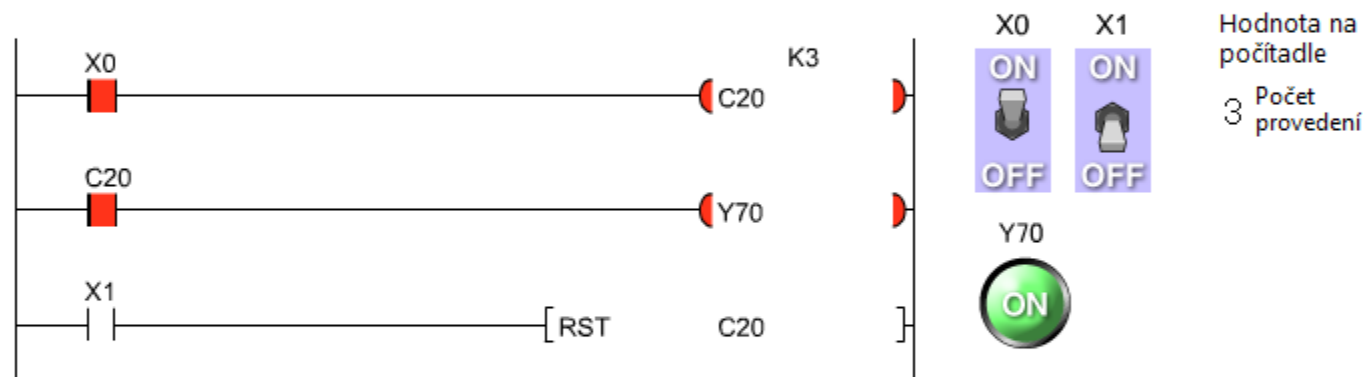
Když je splněna vstupní podmínka, počet se zvýší, a když počet dosáhne specifikované hodnoty, zapne se (ON) specifikovaný čítač (C).

Pokud je čítač (C) resetován instrukcí RST, počet i stav proměnné se inicializují.

Stav čítače (C) je možno použít jako vstupní podmínku v jiných částech programu.

■ Žebříkový program a operace

Simulujte operaci čítače kliknutím na vstupní vypínače vpravo.



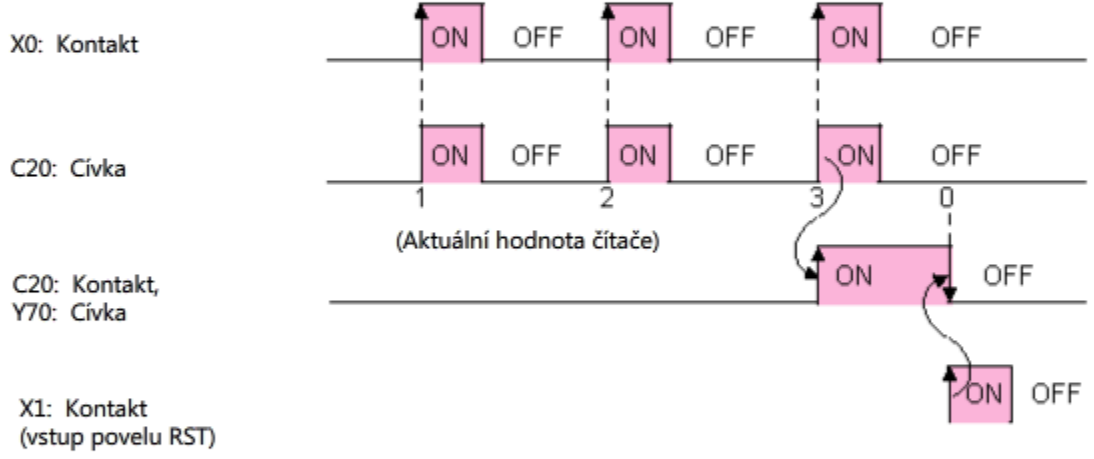
Hodnota v C20 naroste pokaždé, když se X0 zapne (ON). Když počet dosáhne 3, zapne se Y70 (ON).

2.7 Čítání

Instrukční kód a funkce

Symbol	Funkce
	<p>Čítač V kombinaci s výstupem cívky (OUT) čítač počítá (po jednom), kolikrát je splněna podmínka. K sepnutí dojde, když počet dosáhne specifikovaného čísla, a kontakt čítače se zapne (ON). Nastavená hodnota čítače je označena „Kn“ (n: decimální).</p>

Časový diagram



Kapitola 3 Instrukce datových proměnných

Tato kapitola vysvětluje instrukce obsahující datové proměnné.

Datové proměnné jsou užitečné při řízení času, počtu a hodnotových vstupů z externího vybavení.

Datové proměnné mohou zajistit větší citlivost řídicích programů na reálnou operaci.

- Simulujte základní programové operace za účelem pochopení operace s hlavními instrukcemi.
- Simulace umožní pochopit role instrukcí a zpracování prováděné v programovatelném kontroléru.

3.1

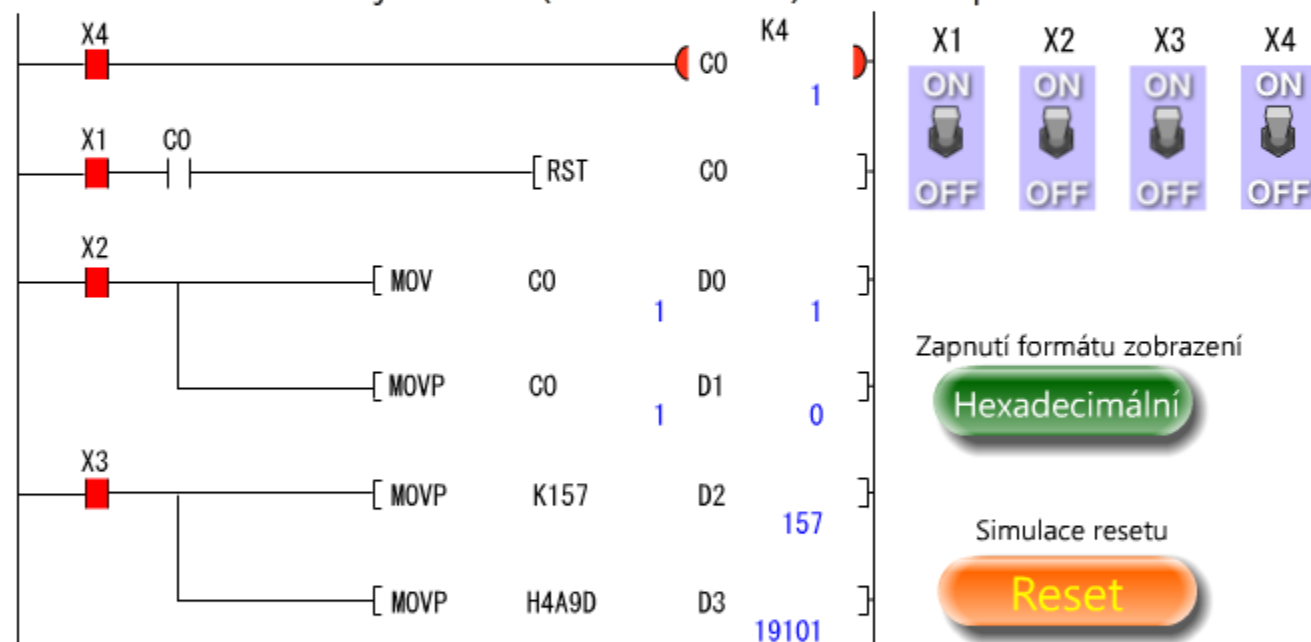
Přesun dat do datových proměnných

16bitová instrukce k přesunu dat (MOV) přemístí (zkopíruje) 1-slovní (16bitovou) datovou jednotku na specifikovanou datovou proměnnou. Přemístitelná data mohou být tvořena buď hodnotou v proměnné, nebo mohou být specifikována. Formát přemístitelných dat může být buď decimální nebo hexadecimální.

■ Žebříkový program a operace

Simulujte operaci s následujícími instrukcemi kliknutím na vstupní vypínače vpravo.

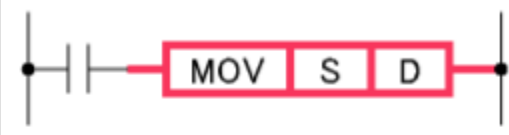
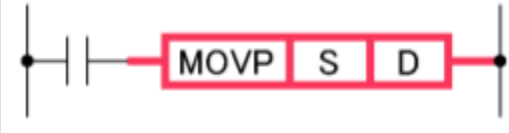
Každé modré číslo označuje hodnotu (aktuální hodnotu) uloženou v proměnné.



Když opakovaně zapnete/vypnete (ON/OFF) X4, aktuální hodnota C0 se zvýší. (0, 1...4->0...).

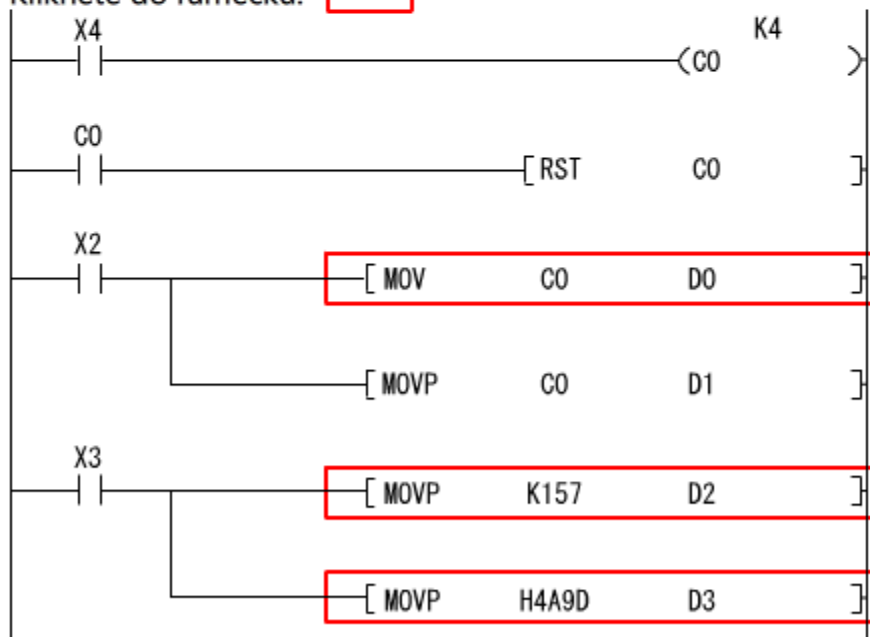
3.1 Přesun dat do datových proměnných

■ Instrukční kódy a funkce

Symbol	Funkce
	<p>Přesun 16bitových dat (MOV) Když je splněna vstupní podmínka, jsou data specifikovaná ve zdroji (S) přenesena (zkopírována) na proměnnou specifikovanou v destinaci (D).</p>
	<p>Přesun 16bitových dat (pulsový) (MOV P) Na náběžné hraně podmínky (OFF na ON) jsou data specifikovaná ve zdroji (S) přenesena (zkopírována) na proměnnou specifikovanou v destinaci (D).</p>

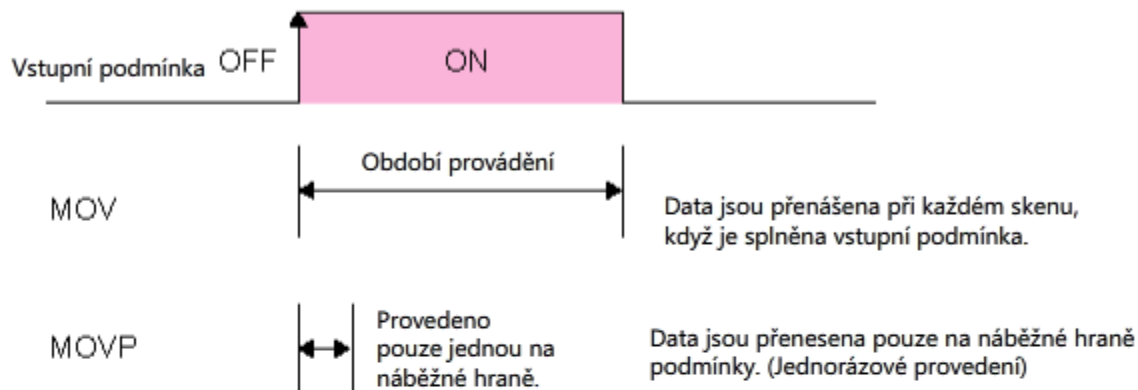
■ Žebříkový program

Klikněte do rámečku.

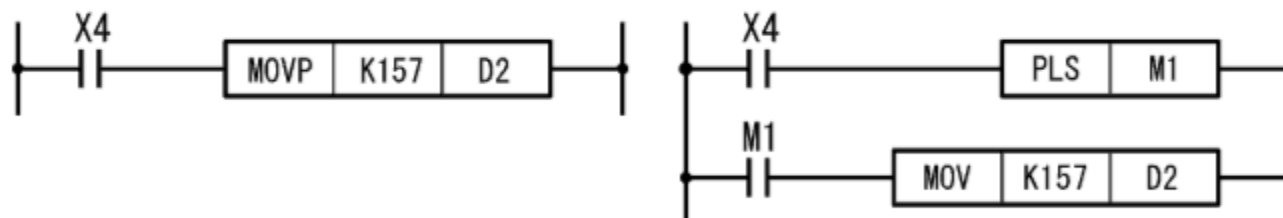


3.1.1 Rozdíl mezi MOV a MOVP

Instrukce MOV se používá k nepřetržitému čtení měnicích se dat. Instrukce MOVP se používá k jednorázovému přenosu dat, například k nastavení dat nebo k načtení dat při výskytu chyby.



Na níže uvedených obrázcích jsou znázorněny dva programy, jejichž výsledkem bude stejná operace, s instrukcemi MOV a MOVP. Na příčkách obou žebříků je přenos dat proveden, jakmile se zapne (ON) X4. Pomocí povelu MOVP může být operace provedena bez použití povelu PLS, který specifikuje provedení operace na náběžné hraně.



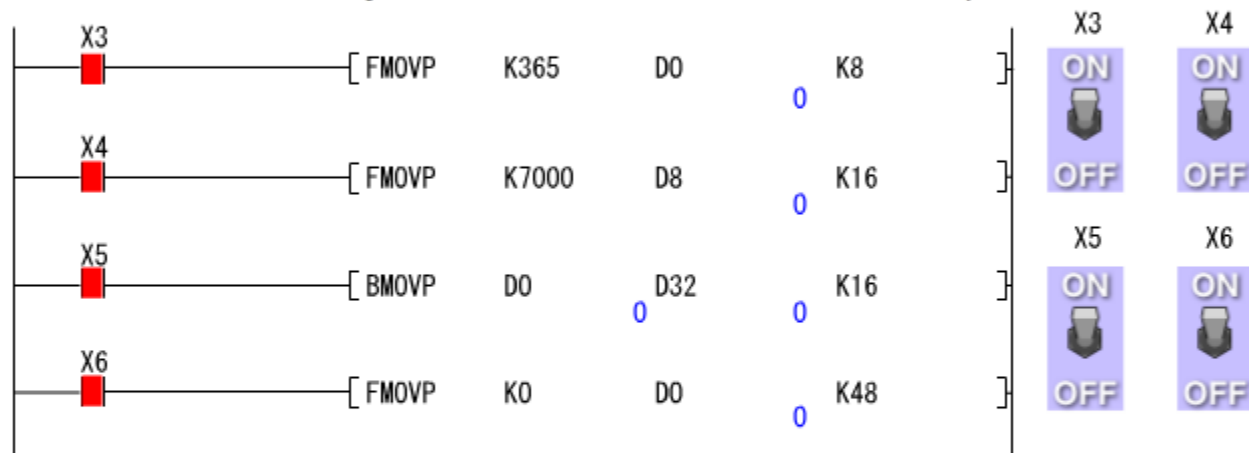
3.1.2

Přesun dat na několik datových proměnných najednou

Instrukce MOV/MOVP se používají k přenosu dat na proměnnou. Pro přenos dat na několik proměnných se souvislými čísly je možno použít instrukci FMOV („instrukce k přenosu dávky identických dat“) nebo povel BMOV („instrukce k přenosu bloku identických dat“).

■ Žebříkový program a operace

Simulujte operaci s následujícími instrukcemi kliknutím na vstupní vypínače vpravo. Každé modré číslo označuje hodnotu (aktuální hodnotu) uloženou v proměnné.



Monitor proměnných

D0	0	D8	0	D32	0
D1	0	D9	0	D33	0
D7	0	D23	0	D47	0

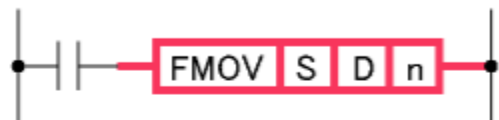
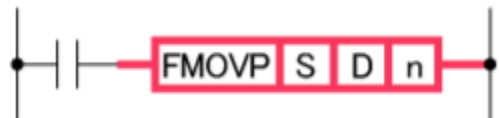
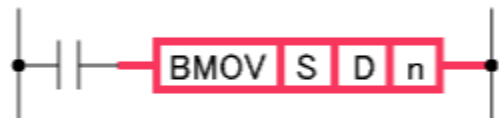
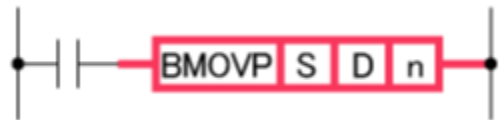
Když se zapne (ON) každý vstupní signál, jsou specifikovaná data přenesena najednou.

POZN.: Na třetí příčce žebříku začínající X5 jsou data přenášena повеlem BMOV.

3.1.2

Přesun dat na několik datových proměnných najednou

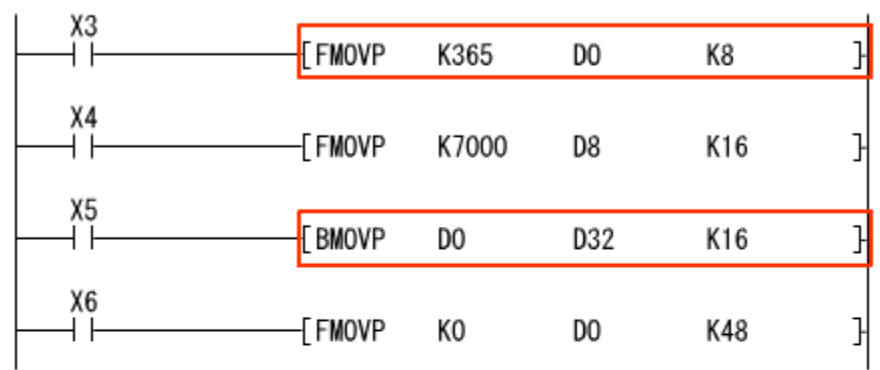
■ Instrukční kódy a funkce

Symbol	Funkce
	<p>Přenos dávky identických dat (FMOV)</p> <p>Když je splněna vstupní podmínka, jsou data specifikovaná ve zdroji (S) přenesena (zkopírována) na proměnnou specifikovanou destinací (D) a na „n“ počet proměnných, která následují po D.</p>
	<p>Přenos dávky identických dat (pulsový) (FMOVP)</p> <p>Na náběžné hraně podmínky jsou data specifikovaná ve zdroji (S) přenesena (zkopírována) na proměnnou specifikovanou destinací (D) a na „n“ počet proměnných, která následují po D.</p>
	<p>Dávkový přenos blokových dat (BMOV)</p> <p>Když je splněna vstupní podmínka, jsou data v proměnné specifikované zdrojem (S) a následném „n“ počtu proměnných přenesena na proměnnou specifikovanou destinací (D) a na následný „n“ počet proměnných.</p>
	<p>Dávkový přenos blokových dat (pulsový) (BMOVP)</p> <p>Na náběžné hraně podmínky jsou data v v proměnné specifikované zdrojem (S) a následném „n“ počtu proměnných přenesena na proměnnou specifikovanou destinací (D) a na následný „n“ počet proměnných.</p>

3.1.2 Přesun dat na několik datových proměnných najednou

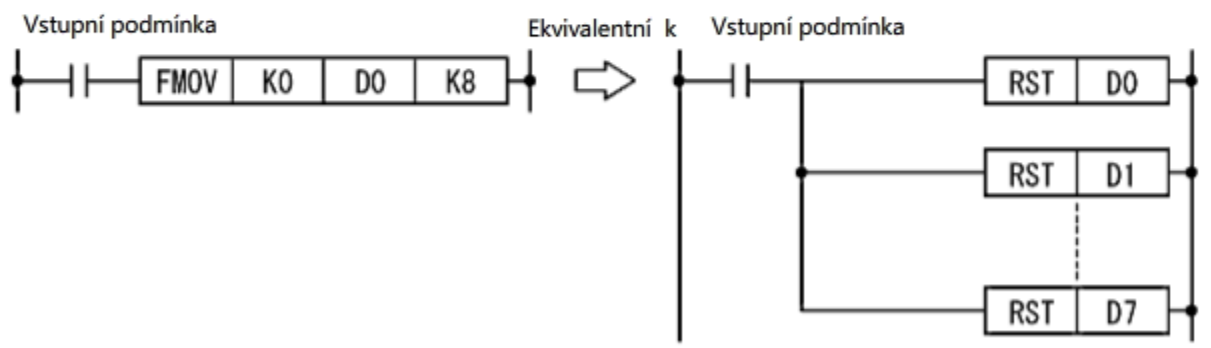
■ Žebříkový program a operace

Klikněte do rámečku .



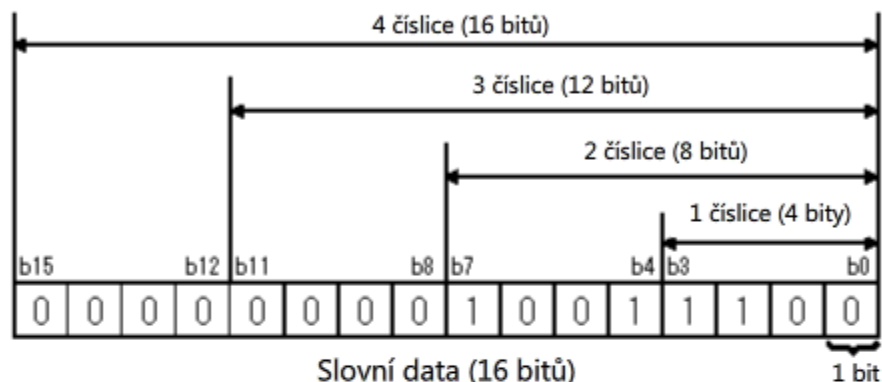
■ Použití povelů FMOV a BMOV

Instrukce FMOV je vhodná pro vymazání velkého objemu dat najednou.



3.1.3 Shluk bitových proměnných

Čtyři bitové proměnné jsou seskupeny do jednoho shluku bitových proměnných pro kontrolu bitových informací určitého rozsahu (přenosu dat atd.).



■ Jak specifikovat shluk bitových proměnných

Číslice bitových proměnných je vyjádřena jako „počet číslic“ + „číslo počáteční proměnné“. Počet číslice je násobkem 4. V níže uvedené tabulce je uvedeno několik příkladů.

V následujících příkladech je číslo počáteční proměnné „M0“.

Bitový rozsah	Metoda specifikace
16bitová data	K4M0 (16 bitů, M0 až M15)
32bitová data	K8M0 (32 bitů, M0 až M31)

Číslice bitové proměnné (počet bitů) určuje rozsah použitelných číselných hodnot.

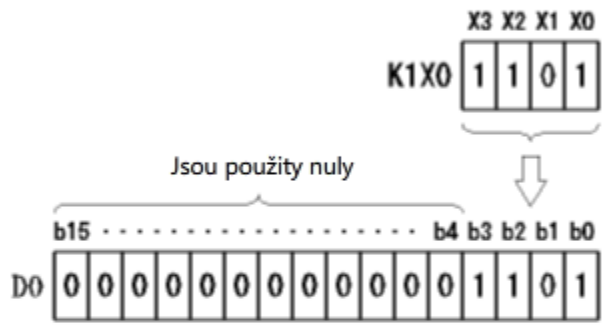
Číslice bitových proměnných	Rozsah použitelných číselných hodnot
K1 (4 bity)	0 až 15
K2 (8 bitů)	0 až 255
K3 (12 bitů)	0 až 4095
K4 (16 bitů)	-32768 až 32767 16. bit je možno použít jako kladný/záporný znak pro vyjádření záporných hodnot.

3.1.3 Příklady přenosu shluku bitových proměnných

Instrukce pro přenos dat se používají k přenosu (kopírování) čísel ze zdroje k destinačnímu zařízení. Následující příklady ukazují, jak se přenášejí specifikovaná data.

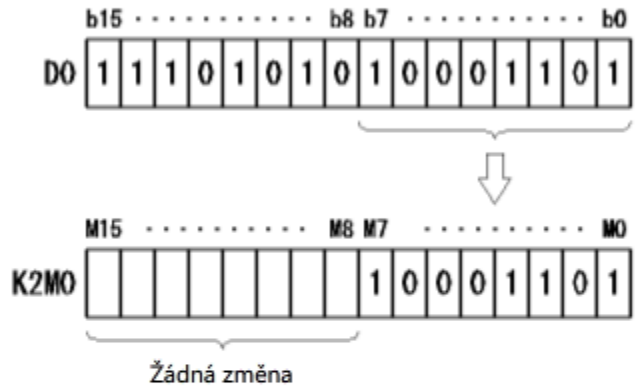
(a) Bitové proměnné se specifikovanými číslicemi → Datové proměnné

Příklad) MOV K1X0 D0



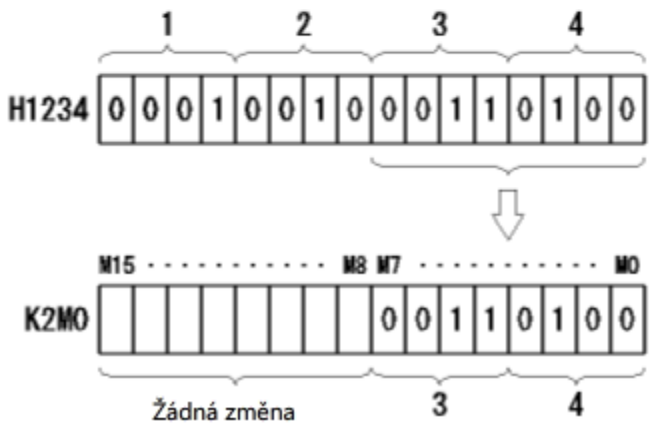
(b) Datové proměnné → Bitové proměnné se specifikovanými číslicemi

Příklad) MOV D0 K2M0



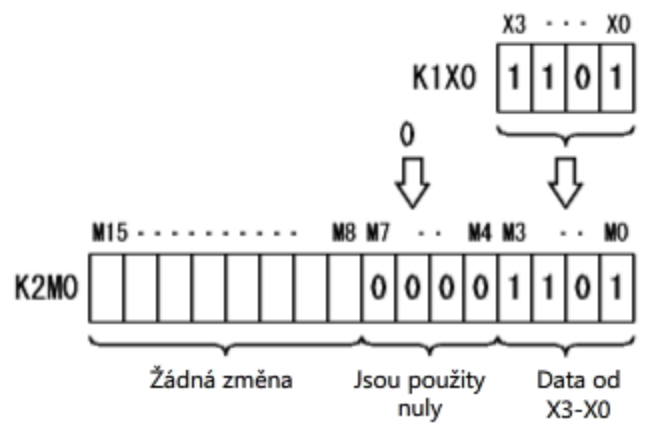
(c) Konstanty (přímo specifikovaná čísla) → Bitové proměnné se specifikovanými číslicemi

Příklad) MOV H1234 K2M0



(d) Bitové proměnné se specifikovanými číslicemi → Bitové proměnné se specifikovanými číslicemi

Příklad) MOV K1X0 K2M0



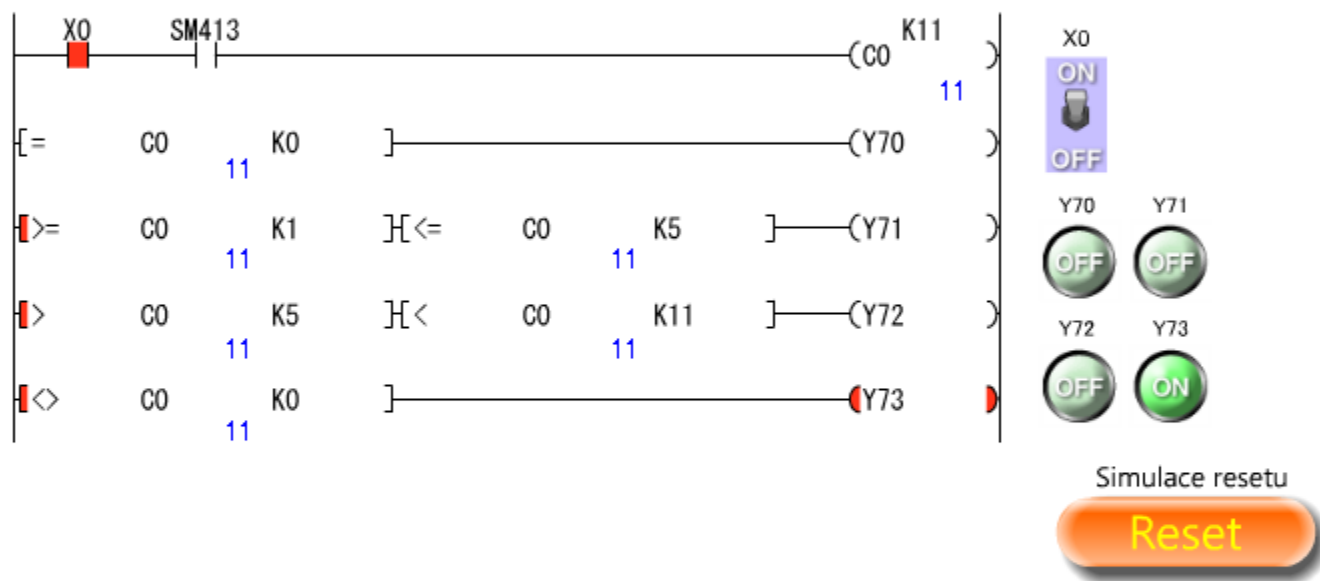
3.2

Porovnávání numerických hodnot

Instrukce pro porovnávací operace se používají k porovnávání dat datových proměnných a dat uložených v datových proměnných. Když je splněna podmínka (\rightarrow), je provedena další instrukce.

■ Žebříkový program a operace

Simulujte operaci s následujícími instrukcemi kliknutím na vstupní vypínače vpravo. Každé modré číslo označuje hodnotu (aktuální hodnotu) uloženou v proměnné.



Y70 až Y73 se zapnou/vypnou (ON/OFF) v závislosti na aktuální hodnotě C0.

SM413 je speciální relé, které je zapínáno (ON) nebo vypínáno (OFF) v 1-sekundových intervalech modulem CPU. (2-sekundové hodiny)

Když je X0 ON, čítá C0 každé 2 sekundy.

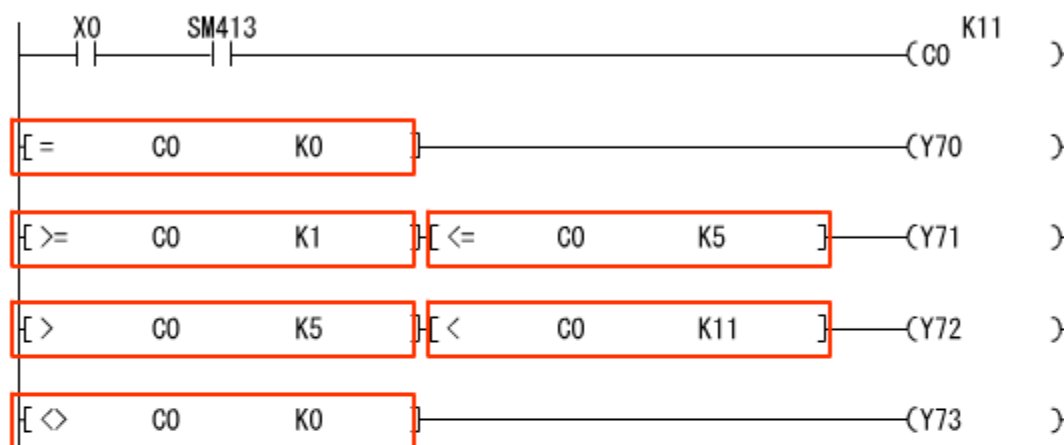
* SM413 je speciální relé, které se zapíná/vypíná (ON/OFF) v 1-sekundových intervalech (2-sekundové hodiny). SM413 je možno použít pro řady MELSEC **iQ-R/Q/L/iQ-F**. Řada MELSEC-F nemá relé s 2sekundovými hodinami, ale má M8011 (0.01 s hodiny), M8012 (0.1 s hodiny), M8013 (1 s hodiny), a M8014 (1 m hodiny).

■ Instrukční kódy a funkce

Symbol	Funkce
	<p>Porovnává 16bitová binární data. (=) Podmínka je splněna, když ZDROJ 1 rovná se ZDROJ 2.</p>
	<p>Porovnává 16bitová binární data. (<) Podmínka je splněna, když ZDROJ 1 je menší než ZDROJ 2.</p>
	<p>Porovnává 16bitová binární data. (>) Podmínka je splněna, když ZDROJ 1 je větší než ZDROJ 2.</p>
	<p>Porovnává 16bitová binární data. (<=) Podmínka je splněna, když ZDROJ 1 rovná se nebo je menší než ZDROJ 2.</p>
	<p>Porovnává 16bitová binární data. (>=) Podmínka je splněna, když ZDROJ 1 rovná se nebo je větší než ZDROJ 2.</p>
	<p>Porovnává 16bitová binární data. (<>) Podmínka je splněna, když ZDROJ 1 nerovná se ZDROJ 2.</p>

■ Žebříkový program a operace

Klikněte do rámečku



SM413 je speciální relé, které je zapínáno (ON) nebo vypínáno (OFF) v 1-sekundových intervalech modulem CPU (2-sekundové hodiny). Speciální relé (SM) jsou reléové proměnné v modulu CPU. Každé speciální relé plní určitou roli.

V této kapitole jsou vysvětleny základní aritmetické operace slovních (numerických) proměnných.

■ Sčítání a odčítání

Aritmetické operace používající symboly pro sčítání (+) a odčítání (-).

■ Násobení a dělení

Aritmetické operace používající symboly pro násobení (*) a dělení (/).

Mezi řadami MELSEC iQ-R/Q/L/iQ-F a řadou MELSEC-F se liší instrukce, ale základní pojetí je stejné.

Vysvětlení v této kapitole jsou založena na instrukcích používaných u řad MELSEC iQ-/Q/L /iQ-F.

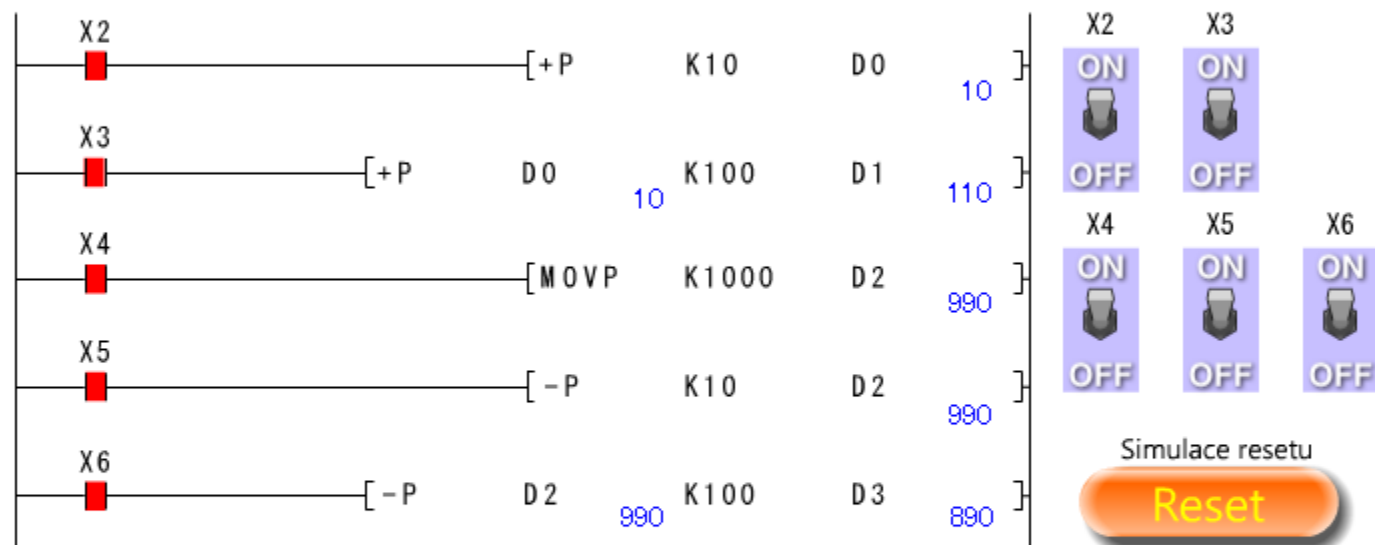
3.3.1

Sčítání a odčítání

V následujícím grafu jsou znázorněny instrukce, které provádějí sčítání a odčítání a ukládají získanou hodnotu do specifikovaných proměnných.

■ Žebříkový program a operace

Simulujte operaci s následujícími instrukcemi kliknutím na vstupní vypínače vpravo. Každé modré číslo označuje hodnotu (aktuální hodnotu) uloženou v proměnné.



Když se každý vstupní signál zapne (ON), je provedena aritmetická operace.

- Příklad je založený na řadách MELSEC iQ-R/Q/L/iQ-F.

3.3.1 Sčítání a odčítání

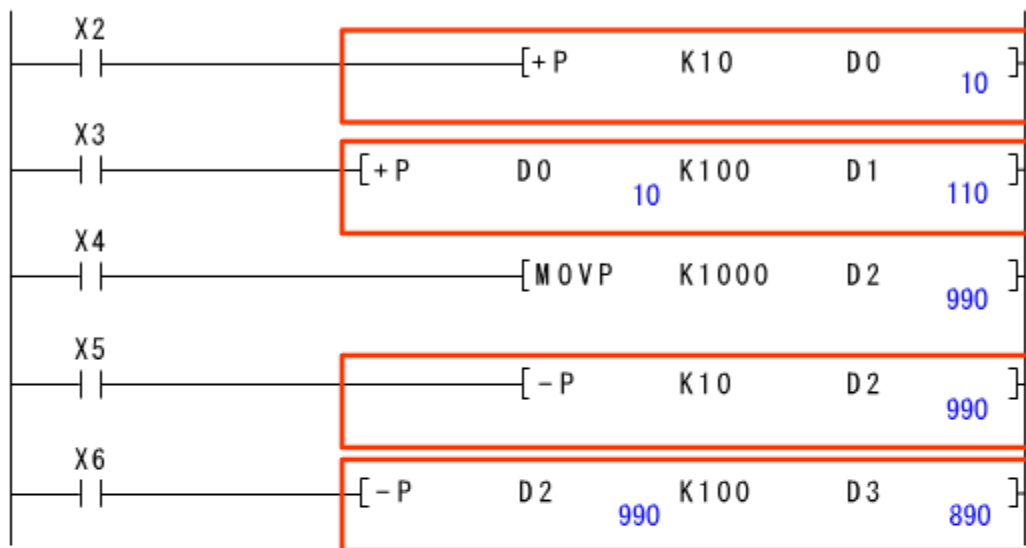
Instrukční kódy a funkce

Symbol	Funkce
<p>The symbol shows a red box containing the text "+ (P) S D". To the left of the box is a vertical line with a horizontal tick mark. To the right of the box is another vertical line with a horizontal tick mark.</p>	<p>Sčítání 16bitových binárních dat</p> <ul style="list-style-type: none"> - : Je provedena operace „D + S = D“. - : Je provedena operace „S1 + S2 = D“.
<p>The symbol shows a red box containing the text "+ (P) S1 S2 D". To the left of the box is a vertical line with a horizontal tick mark. To the right of the box is another vertical line with a horizontal tick mark.</p>	
<p>The symbol shows a red box containing the text "- (P) S D". To the left of the box is a vertical line with a horizontal tick mark. To the right of the box is another vertical line with a horizontal tick mark.</p>	<p>Odčítání 16bitových binárních dat</p> <ul style="list-style-type: none"> - : Je provedena operace „D - S = D“. - : Je provedena operace „S1 - S2 = D“.
<p>The symbol shows a red box containing the text "- (P) S1 S2 D". To the left of the box is a vertical line with a horizontal tick mark. To the right of the box is another vertical line with a horizontal tick mark.</p>	

3.3.1 Sčítání a odčítání

■ Žebříkový program a operace

Klikněte do rámečku

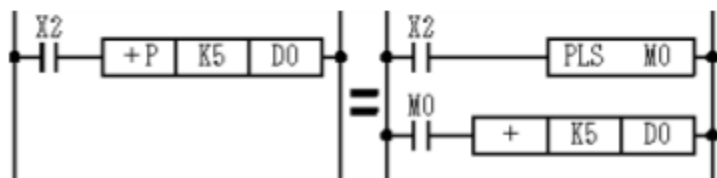


■ Poznámka k instrukcím pro sčítání a odčítání

Za normálních okolností použijte k provádění sčítání / odčítání instrukce +P/-P.

Pokud použijete instrukce +/-, bude sčítání / odčítání prováděno opakovaně, dokud bude splněna vstupní podmínka.

S kteroukoli z následujících příček žebříku bude sčítání provedeno pouze jednou, když se X2 zapne (ON).



* Příklad je založený na řadách MELSEC iQ-R/Q/L/iQ-F.

3.3.2

Násobení a dělení

V následujícím grafu jsou znázorněny instrukce, které provádějí násobení a dělení a ukládají získanou hodnotu do specifikovaných proměnných.

■ Žebříkový program a operace

Simulujte operaci s následujícími instrukcemi kliknutím na vstupní vypínače vpravo. Každé modré číslo označuje hodnotu (aktuální hodnotu) uloženou v proměnné.



Když se každý vstupní signál zapne (ON), je provedena aritmetická operace.

* Příklad je založený na řadách MELSEC iQ-R/Q/L/iQ-F.

3.3.2

Násobení a dělení

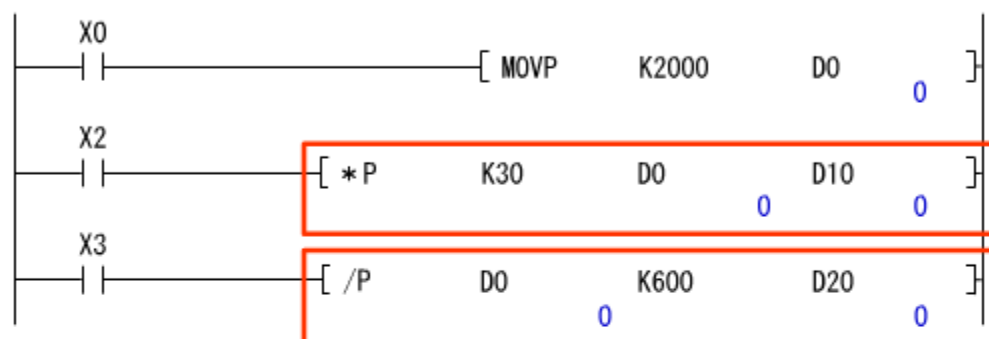
■ Instrukční kódy a funkce

Symbol	Funkce
	<p>Násobení 16bitových binárních dat (*) Je provedena operace „$S1 \times S2 = (D+1 D)$“. („D+1“ je proměnná, která následuje po D. Pokud D je D100, „D+1“ je D101.) Výsledkem operace jsou 32bitová data tvořená 2 slovními proměnnými („D“ a „D+1“).</p>
	<p>Dělení 16bitových binárních dat Je provedena operace „$S1/S2 = (D [kvocient], D + 1 [zbytek])$“. („D + 1“ je proměnná, která následuje po D. Pokud D je D100, „D + 1“ je D101.) Výsledkem operace je celé číslo.</p>

3.3.2

Násobení a dělení

■ Žebříkový program a operace

Klikněte do rámečku 

■ Poznámka k instrukcím pro násobení a dělení

Pro provedení instrukce pro násobení nebo dělení jsou zapotřebí dvě po sobě následující slovní proměnné (D, D+1) pro destinaci (D).

Násobení

$$\begin{array}{|c|} \hline S1 \\ \hline K30 \\ \hline \end{array} * \begin{array}{|c|} \hline S2 \\ \hline D0(2000) \\ \hline \end{array} = \begin{array}{|c|c|} \hline D+1 & D \\ \hline D11 & (60000) & D10 \\ \hline \end{array}$$

Dělení

$$\begin{array}{|c|} \hline S1 \\ \hline D0(2000) \\ \hline \end{array} / \begin{array}{|c|} \hline S2 \\ \hline K600 \\ \hline \end{array} = \begin{array}{|c|c|} \hline D & D+1 \\ \hline D20(3) & D21(200) \\ \hline \end{array}$$

Kvocient Zbytek

* Příklad je založený na řadě MELSEC iQ-R/Q/L/iQ-F.

3.3.3

Rozdíly mezi MELSEC iQ-R/Q/L/iQ-F a MELSEC-F

Řady MELSEC iQ-R/Q/L/iQ-F a řada MELSEC-F se liší symboly.
V následující tabulce je uveden přehled hlavních rozdílů.

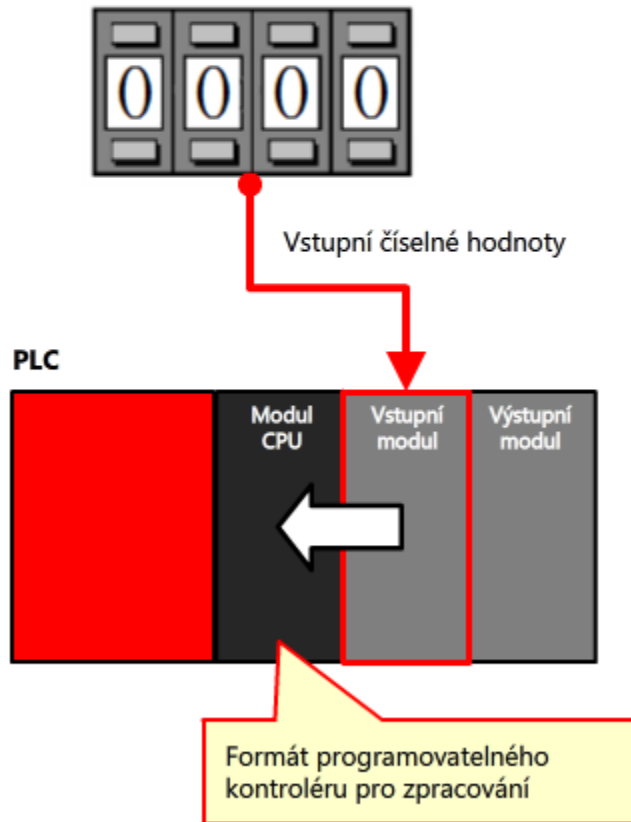
Aritmetická operace	Instrukce používaná u řad MELSEC iQ-R/Q/L/iQ-F	Instrukce používaná u řady MELSEC-F	Rozdíly
Sčítání (+)			Řady MELSEC iQ-R/Q/L/iQ-F: +(P) Řada MELSEC-F: ADD(P)
Odčítání (-)			Řady MELSEC iQ-R/Q/L/iQ-F: -(P) Řada MELSEC-F: SUB(P)
Násobení (*)			Řady MELSEC iQ-R/Q/L/iQ-F: *(P) Řada MELSEC-F: MUL(P)
Dělení (/)			Řady MELSEC iQ-R/Q/L/iQ-F: /(P) Řada MELSEC-F: DIV(P)

3.4 Přenos/příjem dat mezi zařízeními PLC a I/O

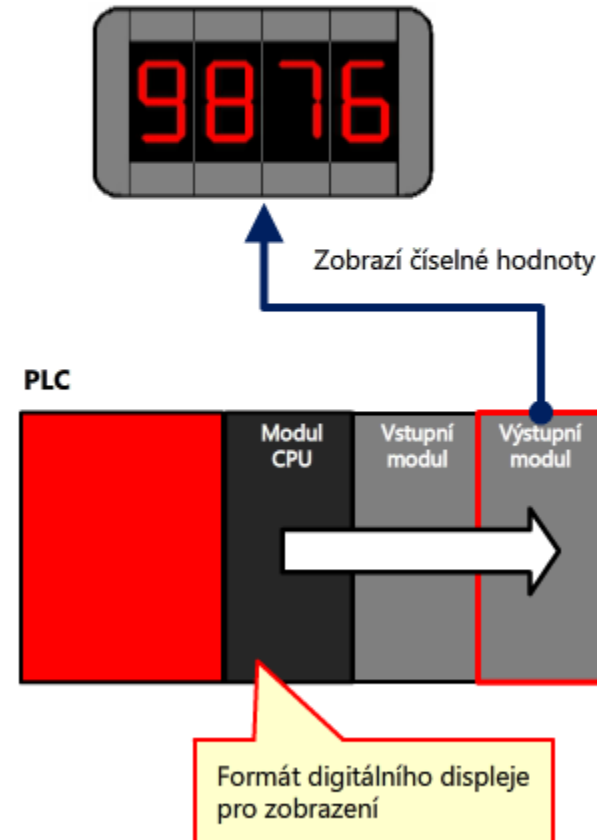
Vypínač digitálního vstupu je zařízení, které přenáší data do programovatelného kontroléru v číselných hodnotách. Digitální displej je výstupní zařízení, které zobrazuje data z programovatelného kontroléru v číselných hodnotách.

Data přijatá z vypínače digitálního vstupu musí být formátována tak, aby mohla být zpracována programovatelným kontrolérem. Podobně výstup dat do digitálního displeje musí být formátován ve formátu, který je pro digitální displej čitelný.

Vypínač digitálního vstupu




Digitální displej



3.4.1 Příjem vstupů vypínače digitálního vstupu

Aby mohl programovatelný kontrolér přijmout vstupy z vypínače vstupu, používá se povel BIN.

■ Instrukční kódy a funkce

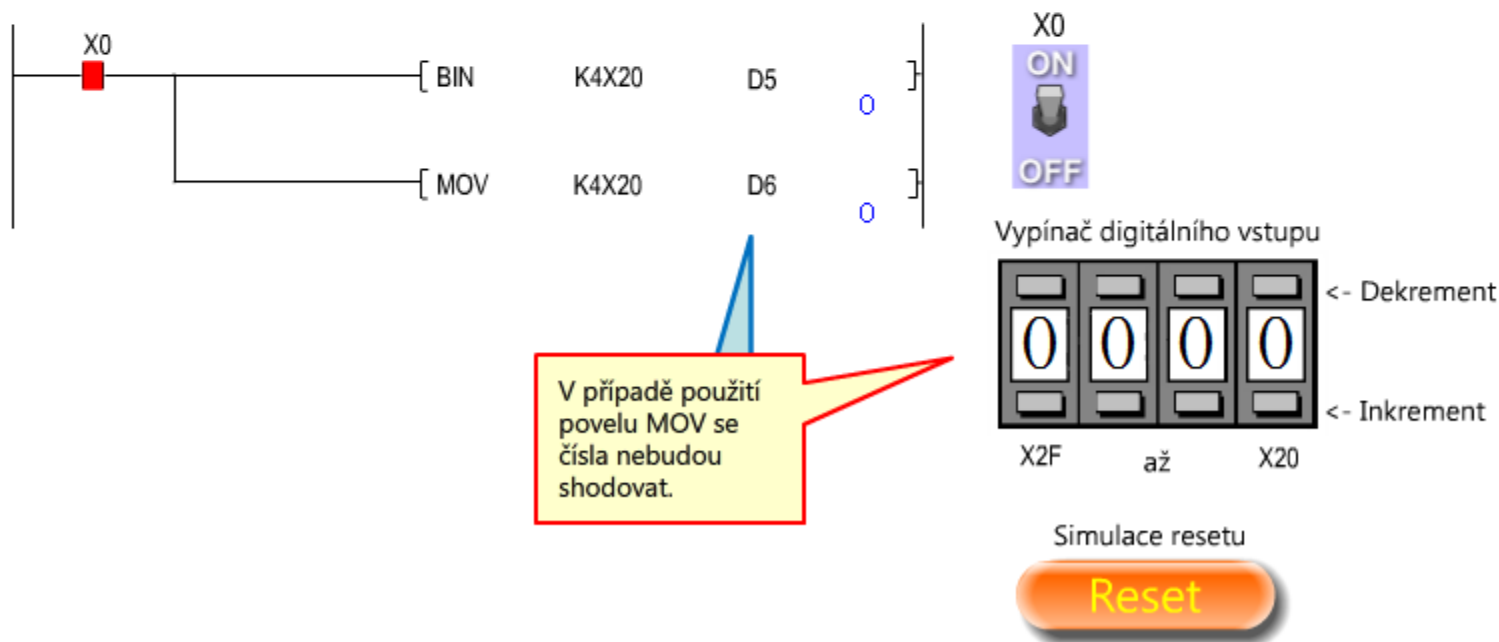
Symbol	Funkce
	Data v proměnné (S) jsou formátována do formátu, který je zpracovatelný programovatelným kontrolérem, a poté se uloží do proměnné (D).

■ Žebříkový program a operace

Simulujte operaci s následujícími instrukcemi kliknutím na vstupní vypínače vpravo. Každé modré číslo označuje hodnotu (aktuální hodnotu) uloženou v proměnné.

Jakmile budou data přijatá z vypínače digitálního vstupu formátována do povelu BIN, budou uchována v D5.

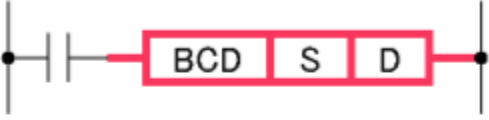
D6 uchovává neformátovaná data přijatá z vypínače digitálního vstupu.



3.4.2 Zobrazení dat PLC na digitálním displeji

Aby bylo možné zobrazit data programovatelného kontroléru na digitálním displeji, použije se povel BCD.

■ Instrukční kódy a funkce

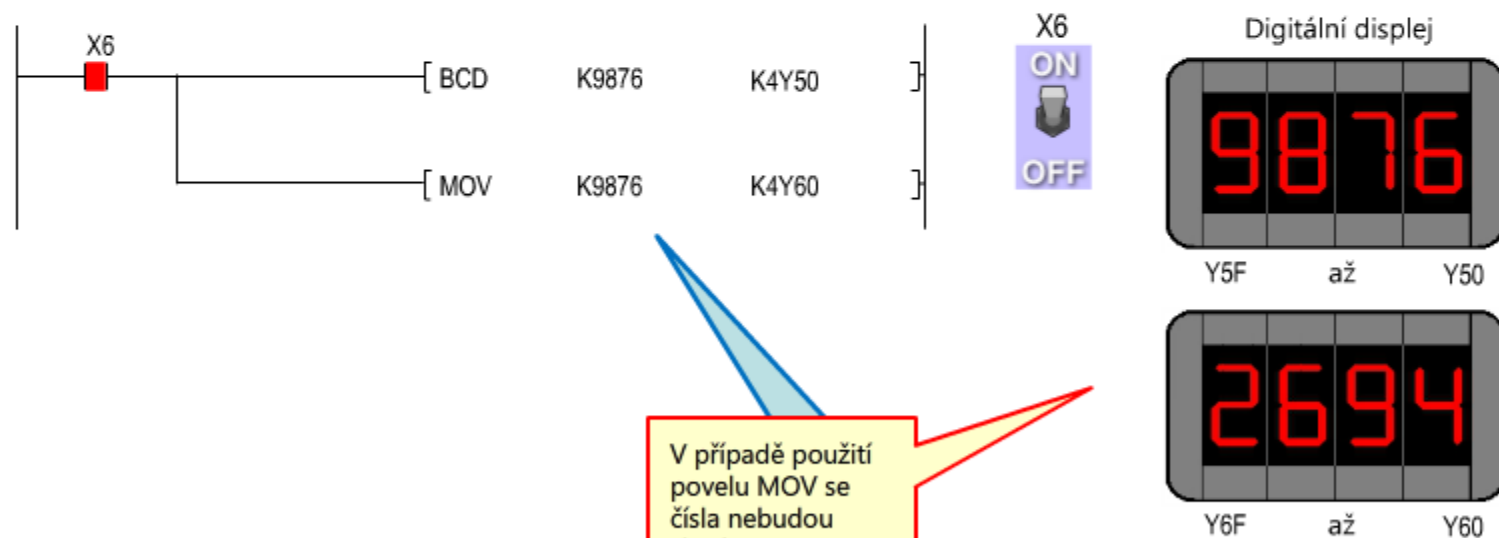
Symbol	Funkce
	Data v proměnné (S) jsou formátována do formátu, který je zobrazitelný na digitálním displeji, a poté se uloží do proměnné (D).

■ Žebříkový program a operace

Simulujte operaci s následujícími instrukcemi kliknutím na vstupní vypínače vpravo.

Na horním digitálním displeji se zobrazí data formátovaná povelům BCD.

Na dolním displeji se zobrazí neformátovaná data.



V případě použití povelu MOV se čísla nebudou shodovat.

Simulace resetu

Reset

V tomto kurzu jste získali tyto znalosti:

- Systém vstupů a výstupů do/od programovatelných kontrolérů
- Hlavní instrukce, které ovládají programovatelné kontroléry
- Informace přijímané programovatelným kontrolérem MELSEC jsou v programovatelném kontroléru prováděny v žebříkových programech a výsledky provádění jsou externě přenášeny jako výstupy
- Rozdíly mezi bitovými a slovními datovými formáty
- Základy ovládacích programů

Pokud se chcete dozvědět, jak editovat a registrovat programy do modulu CPU řad MELSEC iQ-R/iQ-F, absolvujte kurz „Engineering Software MELSOFT GX Works3 (Ladder)” (Engineeringový software MELSOFT GX Works3 (žebříkový)).

Pokud se chcete dozvědět, jak editovat a registrovat programy do modulu CPU řad MELSEC-Q/L/F, absolvujte kurz „GX Works2 Basics” (GX Works2 – základy).

Když jste nyní dokončili všechny lekce kurzu **Základy programování PLC** můžete podstoupit závěrečný test. Pokud si nejste jisti ohledně nějakého tématu, máte nyní možnost si jednotlivá témata zopakovat.

Tento závěrečný test obsahuje celkem 11 otázek (54 bodů).

Závěrečný test můžete podstoupit kolikrát chcete.

Způsob provedení testu

Po vybrání odpovědi nezapomeňte kliknout na tlačítko **Odpověď**. Pokud nekliknete na tlačítko Odpověď, bude vaše odpověď ztracena. (Otázka bude tedy považována za nezodpovězenou.)

Hodnocení výsledků

Na stránce hodnocení se zobrazí počet správných odpovědí, počet otázek, procento správných odpovědí a výsledek úspěšný/neúspěšný.

Počet správných odpovědí: **3**

Celkový počet odpovědí: **4**

Procento: **75%**

Abyste úspěšně složili tento test, musíte správně odpovědět na **60%** otázek.

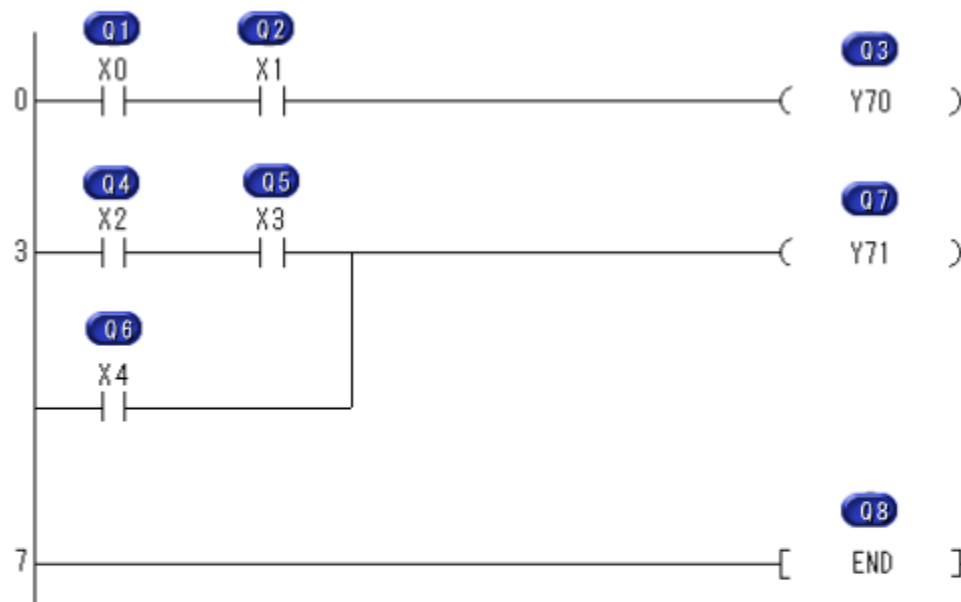
Pokračovat

Zkontrolovat

- Test můžete ukončit kliknutím na tlačítko **Pokračovat**.
- Test si můžete zkontrolovat kliknutím na tlačítko **Zkontrolovat**. (Kontrola správnosti odpovědí)
- Test si můžete zopakovat kliknutím na tlačítko **Znovu**.

Test Závěrečný test 1

Očíslete následující pokyny v pořadí zpracování.



Q1 --Select-- ▾

Q2 --Select-- ▾

Q3 --Select-- ▾

Q4 --Select-- ▾

Q5 --Select-- ▾

Q6 --Select-- ▾

Q7 --Select-- ▾

Q8 --Select-- ▾

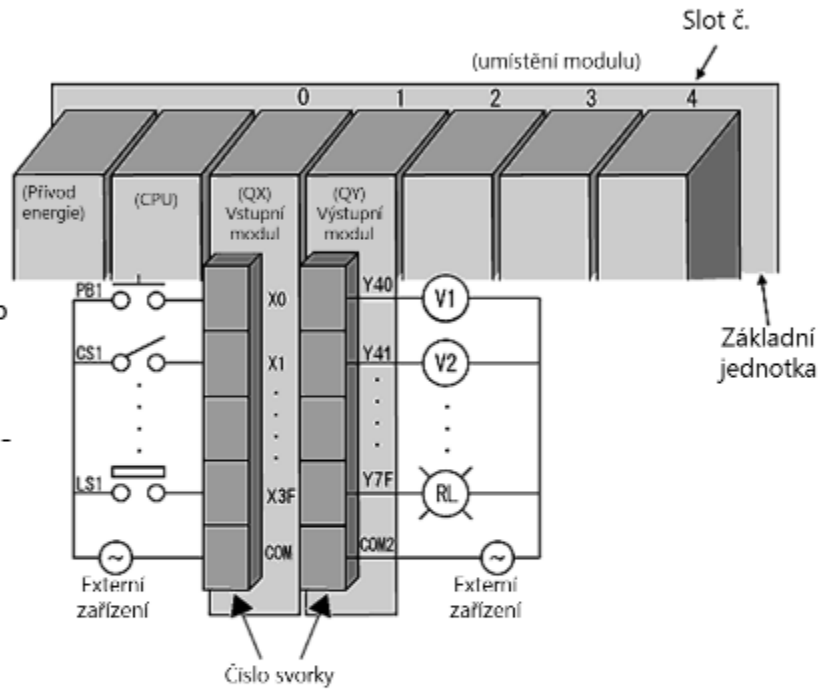
Odpověď

Zpět

Test Závěrečný test 2

Následující věta popisuje externí vybavení I/O a signály I/O do/od programovatelných ovládačů. Zvolte správná slova pro doplnění vět.

- 1) Čísla vstupů a výstupů u programovatelných ovládačů Q Series začínají od (--Select--) a jsou v (--Select--) hodnotách.
- 2) Stejná čísla se používají pro vstupní a výstupní signály. Proto před vstupy předchází (--Select--) a před výstupy předchází (--Select--).
- 3) Čísla přiřazená vstupům signálů z externího vybavení jsou určována těmito podmínkami:
 - Kde na základní jednotce je (--Select--) nainstalován na - svorku číslo
- 4) Čísla přiřazená výstupům (cívkám) do externího vybavení jsou určována těmito podmínkami:
 - Kde na základní jednotce je (--Select--) nainstalován na - svorku číslo



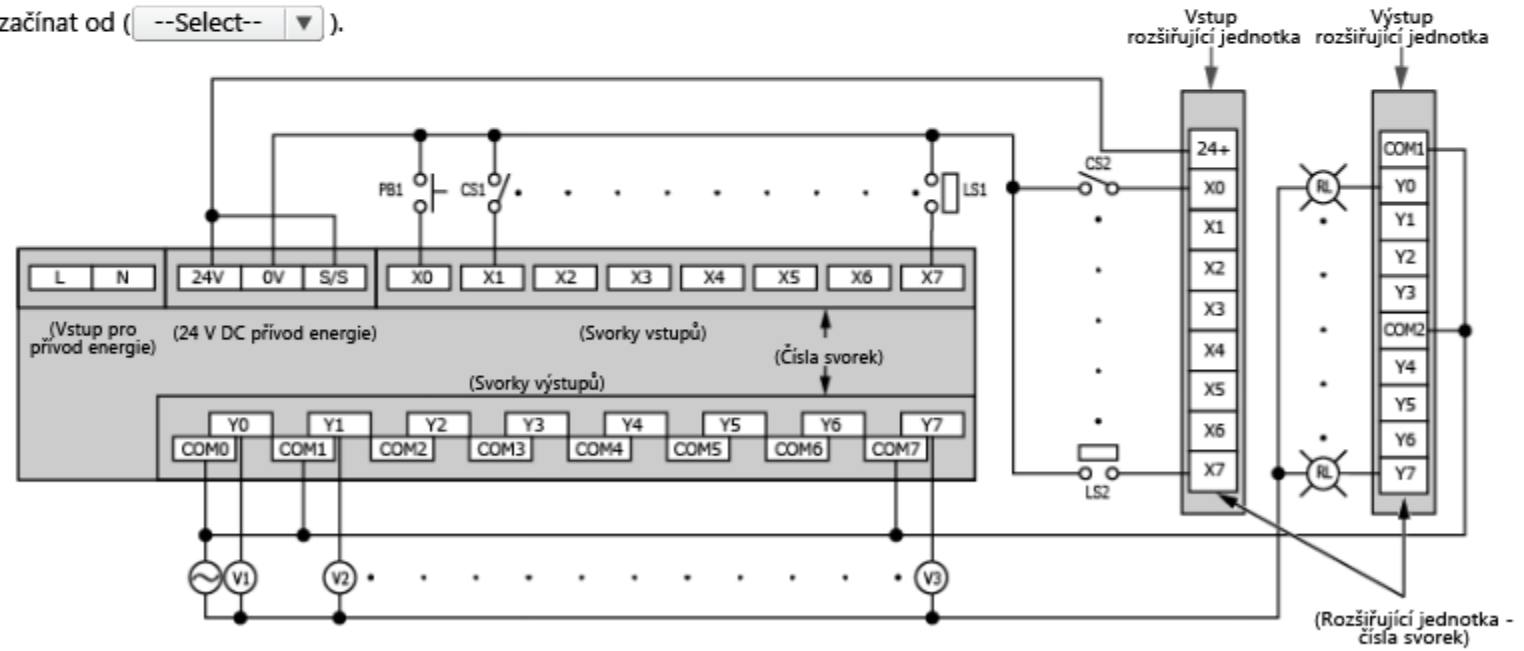
Odpověď

Zpět

Test Závěrečný test 3

Následující věty popisují externí vybavení I/O a číslo I/O přiřazené programovatelným ovládačem. Zvolte správná slova pro doplnění vět. (MELSEC-F Series)

- 1) Čísla I/O u programovatelných ovládačů MELSEC-F Series začínají od (--Select--) a jsou v (--Select--) hodnotách.
- 2) Stejná čísla se používají pro vstupní a výstupní signály. Proto před vstupy předchází (--Select--) a před výstupy předchází (--Select--).
- 3) Pokud je přidána rozšiřující jednotka I/O, bude této jednotce přiřazeno číslo, které následuje po čísle přiřazenému předcházející (--Select--).
- 4) Číslo I/O jednotky vždy začíná od čísla, které má jako první číslici "0". Pokud číslo I/O předcházející jednotky končí na X17, bude číslo I/O následující jednotky začínat od (--Select--).

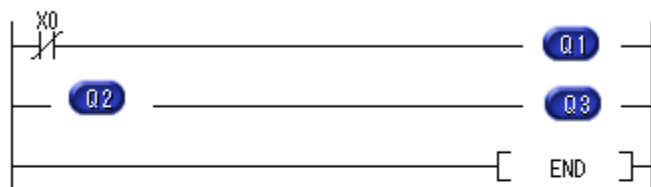


Test Závěrečný test 4

Přetáhněte myší vhodné povely pro dokončení programu, který provádí tyto operace:

Když je vypínač X0 OFF, kontrolka A je ON. (Y70 je ON)

Když je vypínač ON, kontrolka B je ON. (Y71 je ON)



Q1 --Select-- ▼

Q2 --Select-- ▼

Q3 --Select-- ▼

Odpověď

Zpět

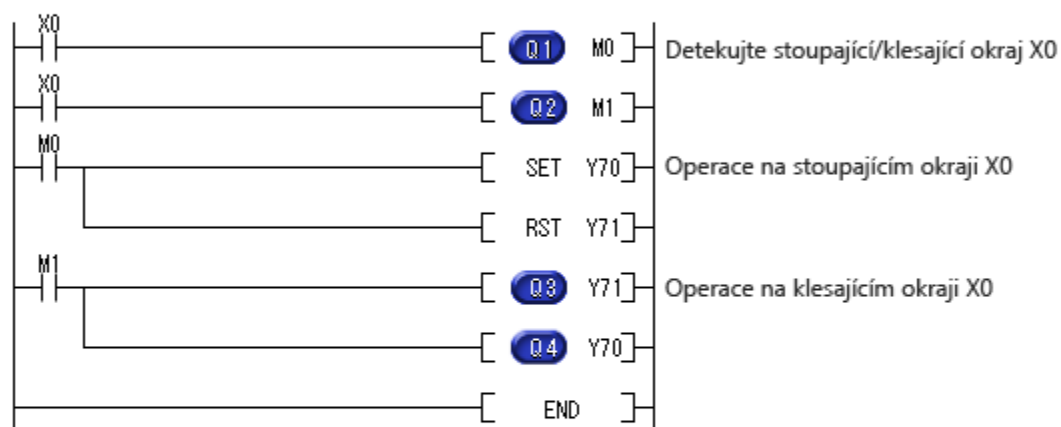
Test Závěrečný test 5

Přetáhněte myši vhodné povely pro dokončení programu, který provádí tyto operace:

Když je zpracováván materiál, je "procesní signál" (X0) ON

Na stoupajícím okraji "procesního signálu" (X0), je kontrolka A ON (Y70 je ON) a kontrolka B je OFF (Y71 je OFF).

Na klesajícím okraji "procesního signálu" (X0), je kontrolka B ON (Y70 je ON) a kontrolka A je OFF (Y71 je OFF).



Q1 --Select-- ▾

Q2 --Select-- ▾

Q3 --Select-- ▾

Q4 --Select-- ▾

Odpověď

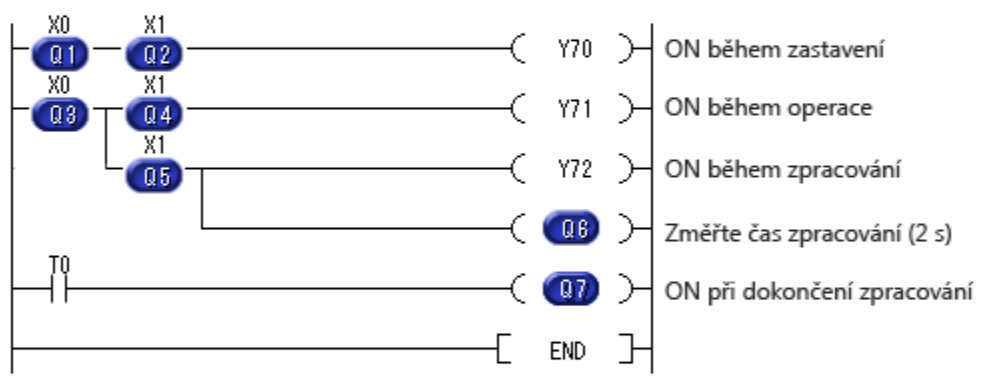
Zpět

Test Závěrečný test 6

Přetáhněte myši vhodné povely pro dokončení programu, který provádí tyto operace:

Kontrolka se rozsvítí (ON) přepnutím na ON/OFF vypínače pro začátek operace (X0) a vypínače pro začátek zpracování (X1). Po 2 sekundách po přepnutí obou vypínačů na ON se rozsvítí (ON) kontrolka D.

[Začátek operace (X0)]	[Vypínač pro začátek zpracování (X1)]	[Kontrolka]
OFF	OFF	Kontrolka A (Y70 je ON)
ON	OFF	Kontrolka B (Y71 je ON)
ON	ON	Kontrolka C (Y72 je ON), a po 2 s Kontrolka D D (Y73 je ON)



Q1 Q2 Q3 Q4
 Q5 Q6 Q7

Test Závěrečný test 7

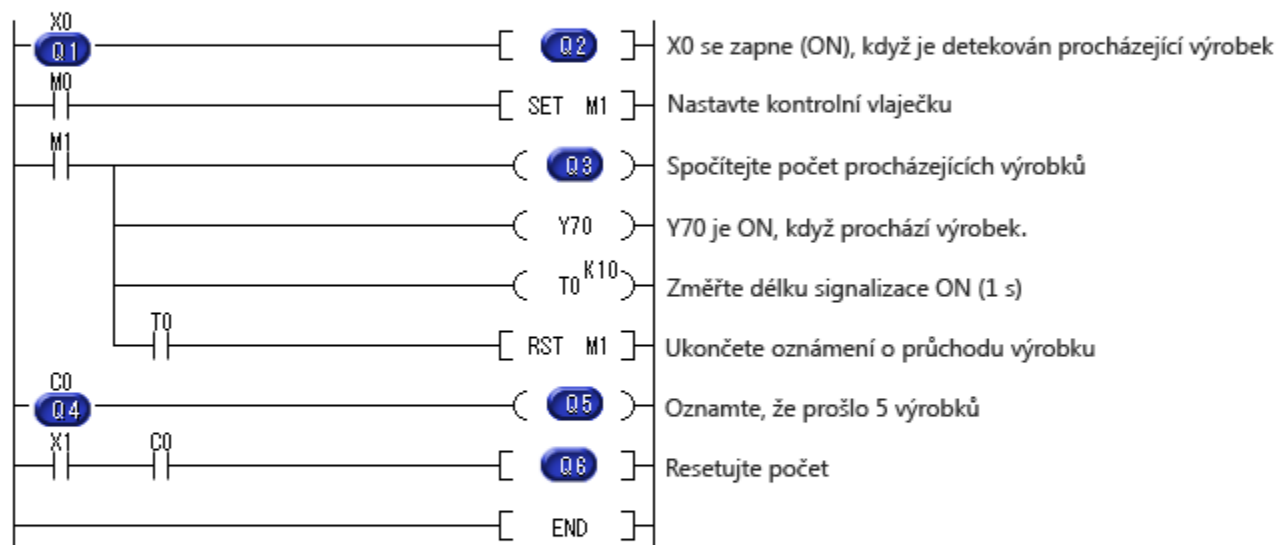
Přetáhněte myši vhodné povely pro dokončení programu, který provádí tyto operace:

Když prochází na dopravníku výrobek, je signál X0 ON.

Po průchodu výrobku (po 3 s) se kontrolka A přepne na ON. (Y70 je ON na 1 s.)

Po průchodu 5 výrobků se zapne (ON) kontrolka B. (Y71 je ON)

Když se zapne (ON) kontrolka B, zapne se (ON) vypínač pro potvrzení (X1).



Q1 --Select-- ▼ Q2 --Select-- ▼ Q3 --Select-- ▼ Q4 --Select-- ▼

Q5 --Select-- ▼ Q6 --Select-- ▼

Odpověď

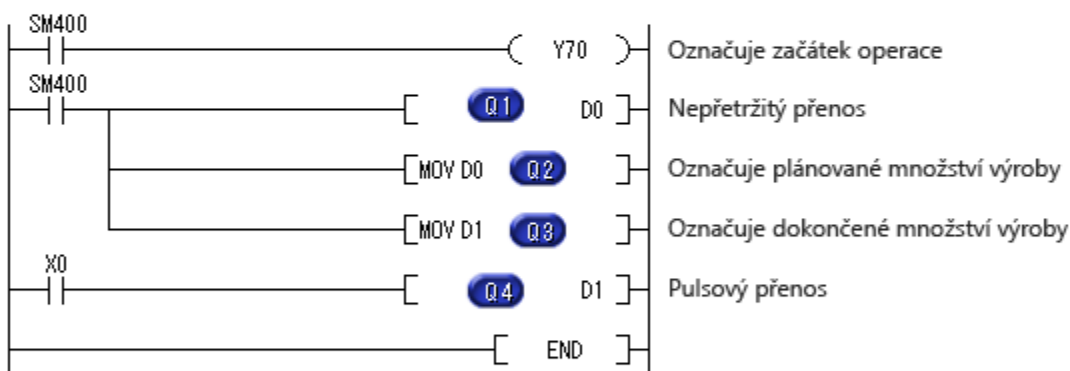
Zpět

Test Závěrečný test 8



Přetáhněte myši vhodné povely pro dokončení programu, který provádí tyto operace:

- 1) Když operace začne, kontrolka A se rozsvítí (ON). (Y70 je ON)
- 2) Plánované množství výroby se zadává pomocí digitálních vypínačů (X20-X2F). Toto množství se při každém zadání přenáší do datového registru D0.
- 3) Hodnoty uložené v datovém registru (D0, D1) jsou nepřetržitě přenášeny na digitální displej a aktualizovány, jak je uvedeno níže.
 - Y40-Y4F: Označuje plánované množství výroby (D0)
 - Y50-Y5F: Označuje dokončené množství výroby (D1)
- 4) Digitální vypínače X30 až X3F se používají k zadání dokončeného množství výroby. Když se vypínač pro nastavení dokončení (X0) přepne na ON, dokončené množství výroby je přeneseno do datového registru D1.



* V tomto programu je povel MOV použit pro přenos dat.
 * Pro monitorování D0 a D1 použijte hexadecimální hodnoty.

Q1 Q2 Q3 Q4

Odpověď

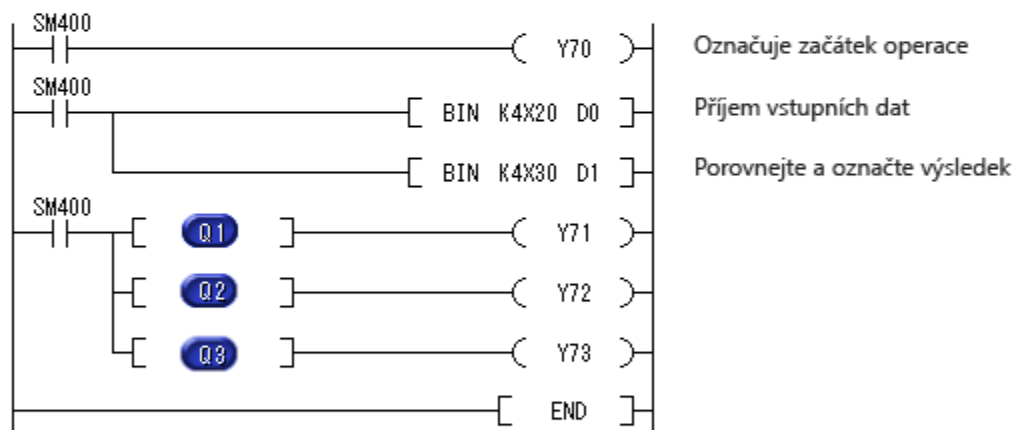
Zpět

Test Závěrečný test 9



Přetáhněte myši vhodné povely pro dokončení programu, který provádí tyto operace:

- 1) Když operace začne, kontrolka A se rozsvítí (ON). (Y70 je ON)
- 2) Jsou jednou provedeny následující operace.
 - Plánované množství výroby A, které bylo zadáno za pomoci digitálních vypínačů (X20-X2F), se zformátuje a přenese do datového registru D0.
 - Plánované množství výroby A, které bylo zadáno za pomoci digitálních vypínačů (X30-X3F), se zformátuje a přenese do datového registru D1.
 - Datové registry D0 a D1 se navzájem porovnají a výsledek je signalizován kontrolkou.
 - D0>D1: Kontrolka B (Y71 je ON/OFF)
 - D0=D1: Kontrolka C (Y72 je ON/OFF)
 - D0<D1: Kontrolka D (Y73 je ON/OFF)

Q1 Q2 Q3

Odpověď

Zpět

Test Závěrečný test 10



Přetáhněte myší vhodné povely pro dokončení programu, který provádí tyto operace:

- 1) Když operace začne, kontrolka A se rozsvítí (ON). (Y70 je ON)
- 2) Na začátku je plánované množství výroby 100 uloženo do datového registru D0.
- 3) Pokaždé, když je dokončen výrobek, uloží se do datového registru následující informace.

D1: Dokončené množství výroby (spočítané na stoupajícím okraji X0)

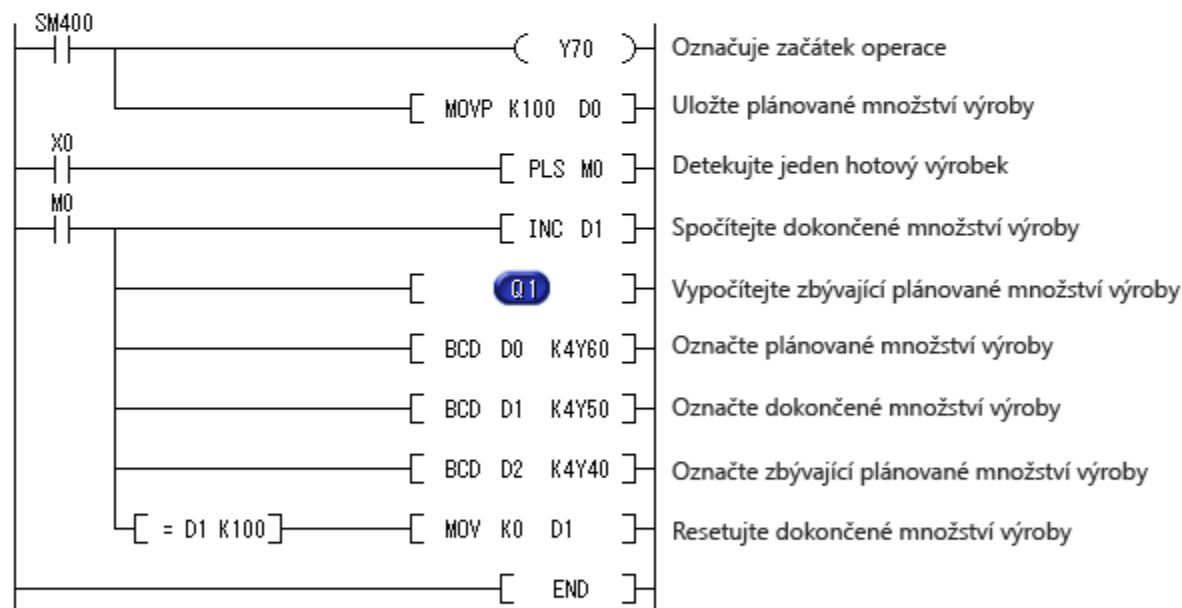
D2: Zbývající plánované množství výroby ($D2 = D0 - D1$)

Na digitálním displeji jsou zobrazena tato data:

Y40-Y4F: Hodnota v D2 (zbývající plánované množství výroby (0 až 100))

Y50-Y5F: Hodnota v D1 (dokončené množství výroby (0 až 100))

Y60-Y6F: Hodnota D0 (plánované množství výroby (100))



Q1

Odpověď

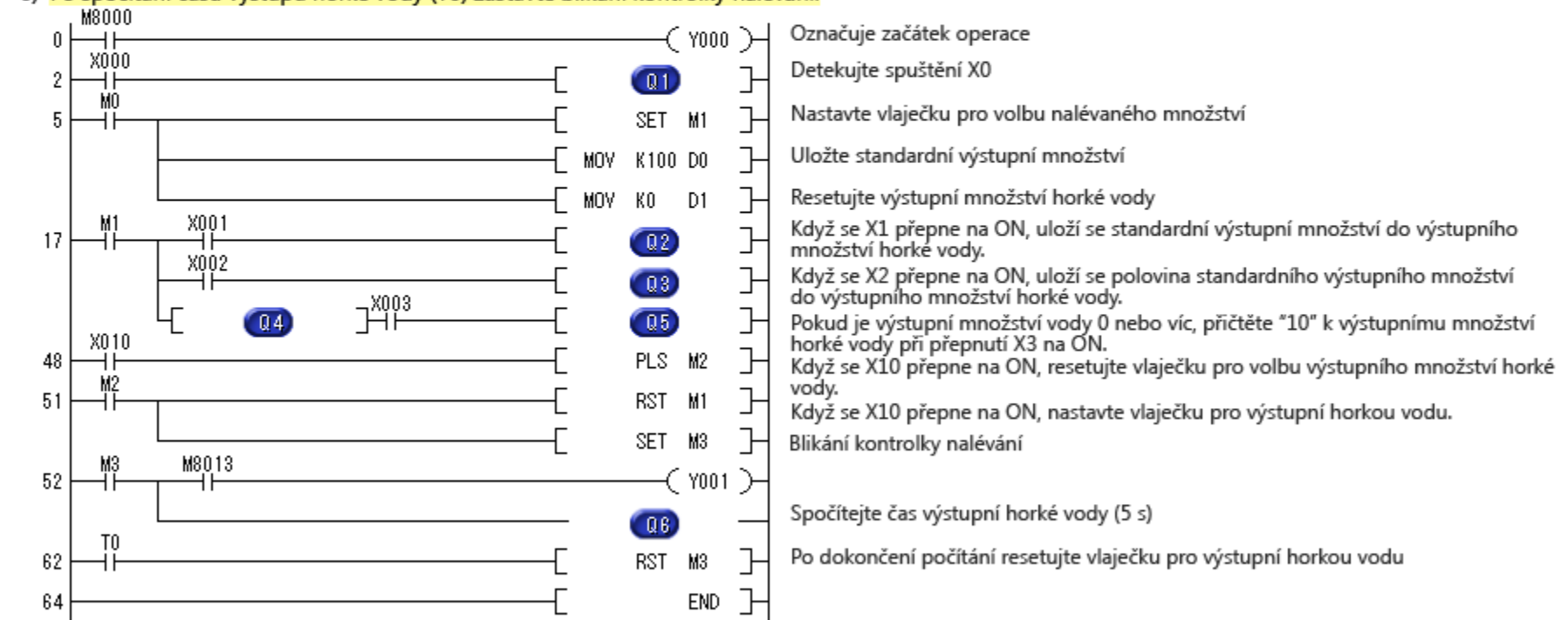
Zpět

Test Závěrečný test 11



Níže uvedený ovládací program platí pro MELSEC-F Series a obsahuje povely a speciální relé.
Přetáhněte myši vhodné povely pro dokončení programu, který nalévá horkou vodu ze zásobníku horké vody:

- 1) Když operace začne, kontrolka se rozsvítí (ON). (Y0 je ON)
- 2) Na stoupajícím konci začátku operace zásobníku horké vody (X0 se přepne na ON) se do standardního dávkovaného množství vody D0 uloží "100" a do výstupního množství horké vody D1 se uloží "0". (Reset dat)
- 3) Zvolte výstupní množství horké vody
Na stoupajícím okraji X1 se uloží standardní výstupní množství D0 do výstupního množství horké vody D1.
Na stoupajícím okraji X2 se uloží polovina standardního výstupního množství D0 do výstupního množství horké vody D1.
- 4) Pokud je zvoleno výstupní množství horké vody D1 a je 0 nebo víc, přičte se k výstupnímu množství horké vody D1 na stoupajícím konci X3 "+10" a poté se uloží přičtená hodnota do výstupního množství horké vody.
- 5) Na stoupajícím okraji výstupu horké vody (X10) bliká kontrolka nalévání v 1 s intervalech (Y1 opakuje ON/OFF) a počítá se čas výstupu horké vody 5 s (T0).
- 6) Po spočítání času výstupu horké vody (T0) zastavte blikání kontrolky nalévání.



Q1 Q2 Q3 Q4

Q5 Q6

Dokončili jste závěrečný test. Vaše výsledky jsou následující.
Pro úplné dokončení závěrečného testu pokračujte na další stranu.

Počet správných odpovědí: **11**

Celkový počet otázek: **11**

Procento: **100%**

Pokračovat

Zkontrolovat

Blahopřejeme. V testu jste uspěli.

Dokončili jste kurz **Základy programování PLC** .

Děkujeme za vaši účast v tomto kurzu.

Doufáme, že se vám lekce líbily a že informace získané v průběhu tohoto kurzu vám budou užitečné.

Celý kurz si můžete projít kolikrát chcete.

Zkontrolovat

Zavřít