

17 Programovatelné automaty (PLC-systémy)

17.1 Konstrukce

Programovatelné automaty (PLC – Program Logic Control) jsou řídicí programovatelné systémy orientované na zpracování jednobitových informací a logické řízení pomocí jednobitových povelů. Programovatelné automaty (PA) jsou řízeny procesorem a používají se při řízení a regulaci procesů (obr. 1).

PA přijímá přes vstupní jednotky (většinou pro 16 jednobitových signálů) logické signály informující o stavu procesu (např. o tom, zda je ve skládce upnut nástroj či nikoliv). Vstupní signály jsou zpracovávány procesorem, který realizuje logické funkce. Výsledkem jsou jednobitové logické signály, které se dostávají jako řídicí signály přes výstupní jednotky zpět do procesu (např. jako povely k zapnutí či vypnutí motoru) (obr. 2).

Program do programovatelného automatu je nahráván buďto ze speciálního programovacího přístroje nebo z PC přes speciální rozhraní (interface). Často je používán PC typu laptop (obr. 2) se speciálním programovým vybavením.

PA jsou sestavovány jako modulární systémy dle požadavků na počty vstupů, výstupů a speciální moduly (např. rychlé čítače pulzů). Kompletně sestavený PA (obr. 3) obsahuje napájecí díl, centrální jednotku, jednotky vstupů, jednotky výstupů a speciální jednotky.

V propojovacích řídicích jednotkách (např. ovládání stykačů) může být funkce realizována prodrátováním (propojením vstupních a výstupních svorek dráty).

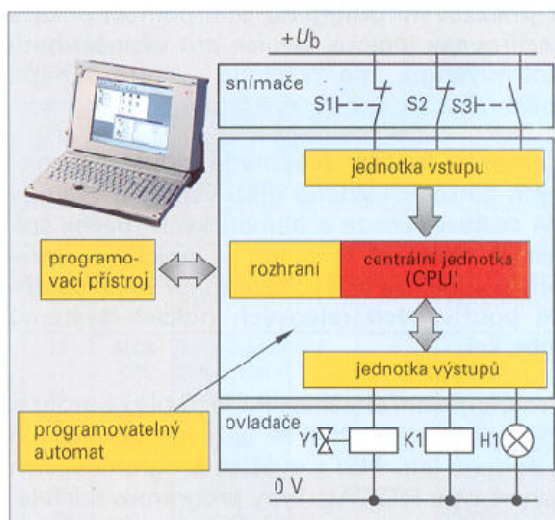
Dnešní PA jsou univerzálně programovatelné jako PC, na rozdíl od pevně zadrátovaných modulů nebo dřívějších typů PA, do kterých bylo nutno při změně programu vložit paměťové moduly s novým programem.

Použití programovatelných automatů (příklady)

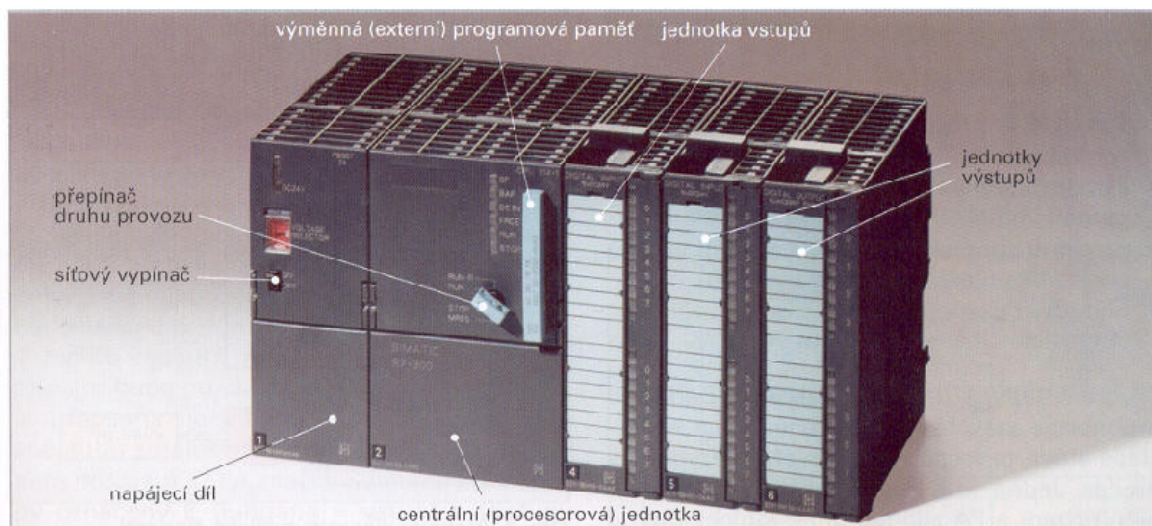
řízení pro:

- mycí linky pro osobní auta
- dopravní systém (ve výrobní hale)
- balicí linky
- plnicí linky
- lis s automatickou obsluhou

Obr. 1 Příklady použití programovatelných automatů



Obr. 2 Blokové schéma programovatelného automatu



Obr. 3 Programovatelný automat Simatic S7-300 firmy Siemens

17.2 Programovací jazyky

Pro programování PA byly vyvinuty příkazové i grafické programovací jazyky. K nejznámějším programovacím prostředkům dříve patřily:

- logické funkční schéma,
- seznam symbolických příkazů (strojového jazyka),
- kontaktní funkční schéma.

Logické funkční schéma je modelem logických funkčních vztahů mezi vstupy a výstupy PA, sestaveným ze schematických značek logických členů (obr. 2a). Logické vztahy mezi vstupy a výstupy jsou zde přehledně vidět.

V **příkazovém programu** jsou pomocí příkazů realizovány logické rovnice pro výpočet hodnot výstupů na základě vstupů, např. $E 2.0 \wedge E 2.1 \vee E 2.2 = A 6.0$ (obr. 2b).

Kontaktní funkční schéma je modelem logických funkčních vztahů mezi vstupy a výstupy PA sestaveným ze schematických značek spínacích kontaktů (sepnuto = logická 1, rozepnuto = logická 0), vycházejícím z principu dříve používaných reléových řídicích systémů (obr. 2c).

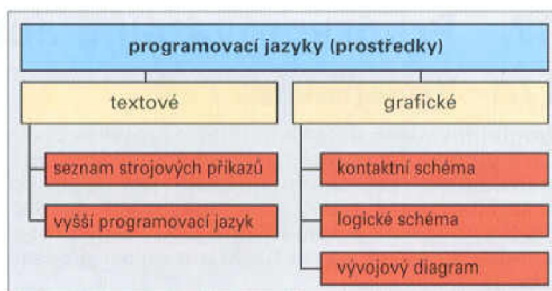
Programování dnešních PA je již blízké programování PC a používá se běžných vývojových diagramů (str. 418) a vyšších programovacích jazyků typu PASCAL. Typy programovacích jazyků popisuje mezinárodní norma IEC 1131-3.

17.3 Činnost programovatelného automatu

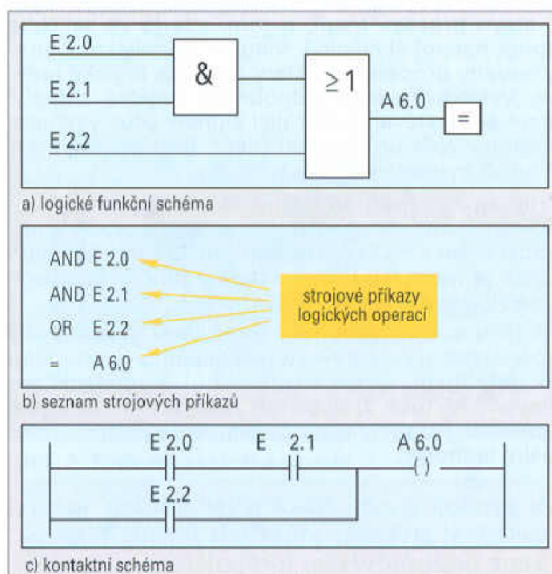
Program PA probíhá cyklicky (opakovaně) v těchto krocích:

- aktuální stav vstupních signálů je uložen do paměti vstupů,
- program zpracuje hodnoty z paměti vstupů a výsledky uloží do paměti výstupů,
- hodnoty z paměti výstupů jsou na určitý časový okamžik přeneseny na výstupy.

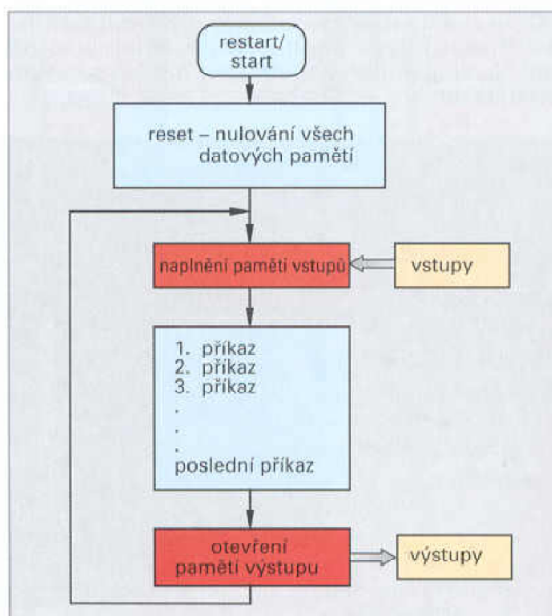
Program tedy s určitou opakovací periodou monitoruje stav řízeného procesoru a na základě stavu procesu dává povely, kterými řídí proces. Jedná se o logické (nikoliv proporcionální) řízení a PA zde simuluje logický obvod se střídavě spínanými vstupy a výstupy.



Obr. 1 Programovací jazyky (prostředky)



Obr. 2 Zobrazení programu různými vývojovými prostředky



Obr. 3 Blokový diagram cyklické činnosti programu PA

Změní-li se stav procesu během programového cyklu, je změna zaznamenána až při dalším čtení vstupních signálů PA. Změní-li vstupní signál hodnotu několikrát během periody snímání vstupů, nejsou všechny tyto změny zohledněny při zpracování programu (pokud není předřazen rychlý čítač). Při řízení pomocí PA je třeba tedy počítat s určitou reakční dobou (zpožděnou reakcí), která se může blížit dvojnásobku doby programového cyklu (přibližně 8 ms při 1000 strojových příkazech).

PA zpracovávají program cyklicky. Během programového cyklu je přečtena paměť vstupů, data jsou zpracována a výsledky jsou uloženy do paměti výstupů. Po ukončení programového cyklu jsou hodnoty z paměti výstupů přeneseny na určitou dobu na výstupy. Řídicí systémy s PA mají zpožděnou reakci se zpožděním závislým na délce programu.

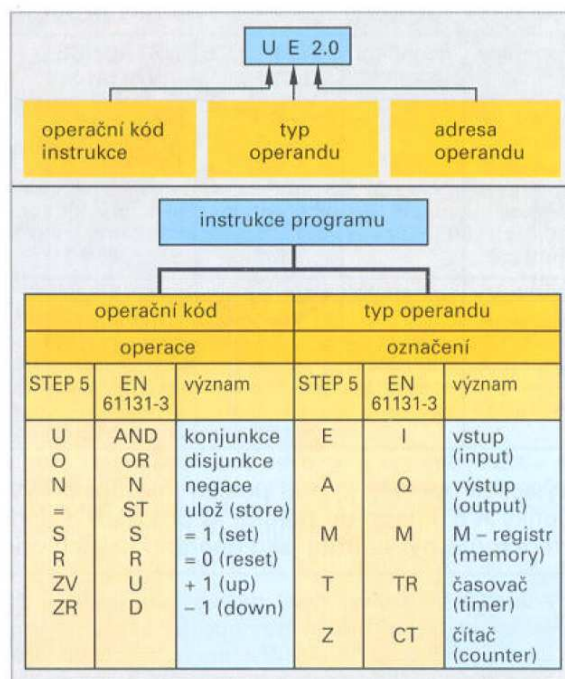
Činnost programu

Výkonný program se skládá z jednotlivých po sobě jdoucích příkazů. Následující příklad je uveden v programovacím jazyku STEP 5 pro PA typu SIMATIC S-7 firmy Siemens a pro srovnání jsou uvedeny znaky mezinárodního anglického standardu EN 61131-3 (**obr. 1**).

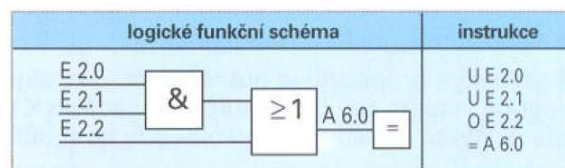
Program vytvořený ve vyšším jazyce, strojově orientovaném jazyce na PC nebo programovacím přístroji je ze zdrojového textu přeložen do strojového kódu a nahrán do paměti PA. Činnost bude dále vysvětlována na programu vytvořeném ve strojově orientovaném symbolickém jazyce.

Příklad 1: Výstup (A 6.0) = (E 2.0) AND (E 2.1) OR (E 2.2) dle schématu na **obrázku 2**.

Program nejprve načte vstupy do paměti vstupů. Např. E 2.0 = 1, E 2.1 = 0, E 2.2 = 1. Pak procesor vykoná jednotlivé instrukce (**obr. 3**). Výsledkem každé instrukce (příkazu) je logická operace. V příkladu 1 je v prvním příkazu ve vstupním registru E 2.0 = 1. Po konjunkci (AND) s hodnotou druhého vstupního registru E 2.1 = 0 bude ve výsledkovém registru výsledek 0 a po disjunkci (OR) s hodnotou E 2.2 = 1 bude výsledek 1 a ten bude přepsán do výstupního registru A6.0. Hodnota výstupního registru bude po ukončení programového cyklu přepsána na výstup (na výstupní svorku spojenou s nějakým ovládacím prvkem v řízeném procesu). U PA značky Siemens jsou vstupy označeny E (Eingang = vstup) a výstupy A (Ausgang = výstup).



Obr. 1 Instrukce programovacího jazyka STEP 5



Obr. 2 Logické funkční schéma a program (seznam příkazů) k příkladu 1

číslo řádku programu	instrukce (příkaz)	hodnota operandu	operace	hodnota výsledku	řídící instrukce
1	U E 2.0	1		1	
2	U E 2.1	0		0	
3	O E 2.2	1		1	
4	= A 6.0				1

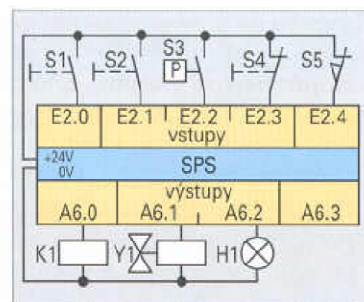
Obr. 3 Činnost programu PA

17.4 Programování

17.4.1 Základní logické operace

Při převodu řídicí úlohy do programu pro PA jsou nejprve stavovým signálům snímačů a ručních ovládacích tlačítek přiřazeny vstupy PA. Stavů sepnuto odpovídá logická 1 a stavů rozepnuto odpovídá logická 0 (obr. 1). Pak jsou řídicím signálům ovladačů (např. relé nebo stykačů) přiřazeny výstupy PA.

V tabulce 1 jsou uvedeny k základním logickým operacím ekvivalentní sekvence logických instrukcí a ekvivalentní schémata s logickými i kontaktními členy. Logické instrukce jsou uvedeny jak v kódu jazyka STEP 5, tak i ve standardním kódu dle EN 61131-3.



Obr. 1 Připojení PA k řízenému procesu

Tabulka 1: Názvy, značky a symboly programovacího jazyka pro PA					
operace	označení (STEP 5)	INSTRUKCE (STEP 5)	INSTRUKCE (EN 61131-3)	logické schéma	kontaktní schéma
AND (konjunkce)	U	U E 2.0 U E 2.1 = A 6.0	LD I 2.0 AND I 2.1 ST Q 6.0		
OR (disjunkce)	O	O E 2.0 O E 2.1 = A 6.1	LD I 2.0 OR I 2.1 ST Q 6.1		
negace	N	UNE 2.0 U E 2.1 = A 6.2	LDN I 2.0 AND I 2.1 ST Q 6.2		
přiřazení	=	= A 6.3 = M 3.6	ST Q 6.3 ST M 3.6		

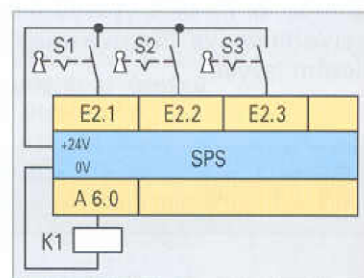
Výsledek operace je buď poslán (uložen) do výstupní paměťové buňky A 6.0 nebo do paměti M pro mezivýsledky (M – registru), která může být použita jako operand dalších operací.

Příklad 2: Výrobní zařízení může být spuštěno buď jedním servisním spínačem S1 na klíč mistra nebo současným zapnutím spínačů S2, S3 klíči vedoucích směn. Zařízení je připraveno ke spuštění, je-li sepnut stykač K1. Sestavte program pro ovládací zařízení.

Nejprve je třeba provést přiřazení vstupů a výstupů stavovým a ovládacím signálům (tabulka 2).

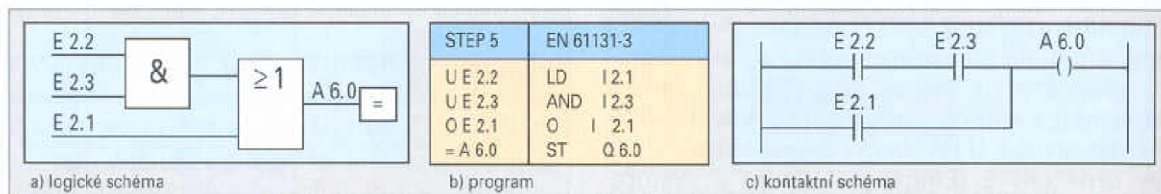
K sestavení programu je nutné vědět, jaké signály je nutno připojit na vstupy, aby byl výstup 1 (ovládající K1) vodivý. Otázka, zda je stykač již sepnut či nikoliv, zde není důležitá.

Zadání úlohy v příkladu 2 lze formulovat následovně: Výstup A 6.0 vede (je sepnut) signál logická 1, jsou-li na vstupech E 2.2 AND E 2.3 (tj. současně) signály logická 1 OR (nebo) když samotný vstup E 2.1 vede (je sepnut) signál logická 1. Odtud lze odvodit logické schéma (obr. 3a), program (obr. 3b) a kontaktní schéma (obr. 3c).



Obr. 2 Připojení PA k procesu v příkladu 2

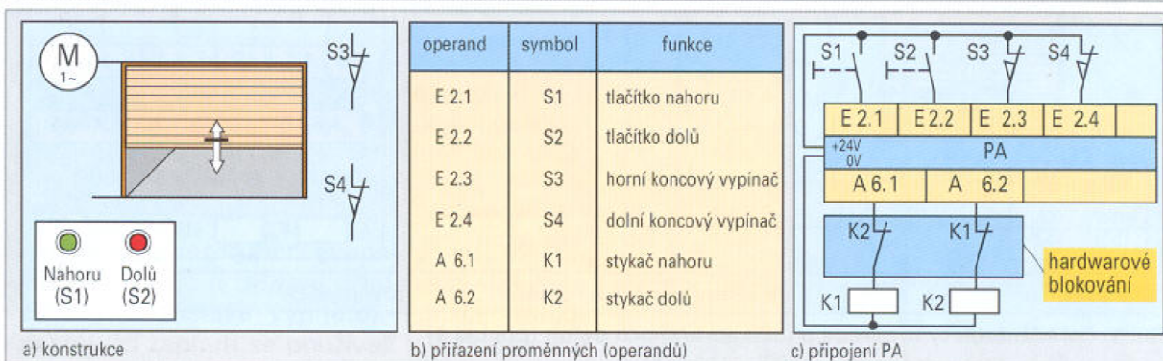
Tabulka 2: Přiřazení vstupů a výstupů		
operand	symbol	komentář
E 2.1	S1	mistr
E 2.2	S2	vedoucí směny 1
E 2.3	S3	vedoucí směny 2
A 6.0	K1	připravenost zařízení (napájení)



Obr. 3 Program a schémata k příkladu 2

17.4.2 Spínací a rozpínací kontakty

Příklad 3: Roletová vrata dílny (obr. 1a) je třeba v tiskacím režimu tlačítkem S1 otvírat (pohyb nahoru) nebo tlačítkem S2 zavírat (pohyb dolů). Horní koncovou (otevřenou) polohu hlásí koncový spínač S3 a dolní koncovou polohu (roleta zatažena) hlásí koncový spínač S4. Sestavte program pro ovládání rolety programovatelným automatem SIMATIC – S7.



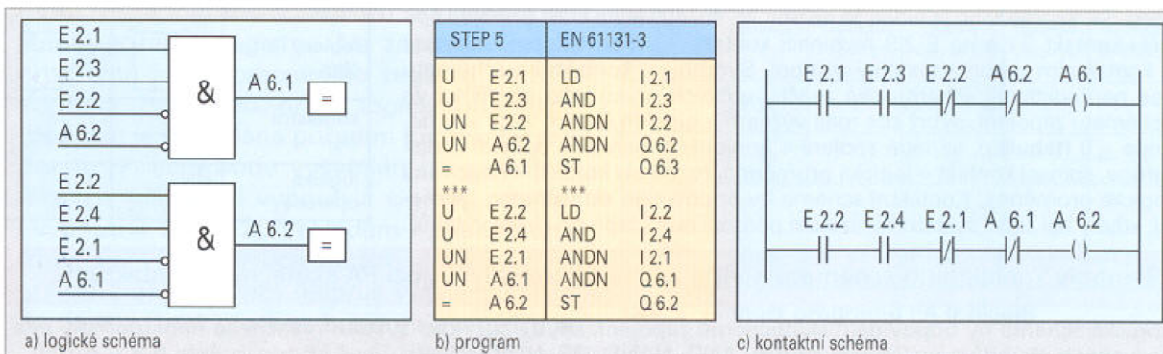
Obr. 1 Ovládání pohybu rolety střídavým zapínáním (příklad 3)

Oba výstupy ovládající levý a pravý chod servomotoru nesmějí být nikdy sepnuty současně, aby nedošlo ke zkratu. Proto musí být vzájemně blokovány (zapnutí jednoho blokuje současné zapnutí druhého). Blokování je možno zajistit pomocí SW (programu) (obr. 2) i pomocí HW (rozpínacími kontakty) (obr. 1c). Provedení bez hardwarového zajištění norma nepřipouští.

K sestavení programu je nutno nejprve přiřadit signálům svorky (tj. i odpovídající paměťová místa) vstupních a výstupních jednotek PA. Pak může logický popis programu znít:

- otevírání:** Povel pro vytahování rolety nahoru (motor zapnut pro vytahování) bude odvozen z výstupu A 6.1 = 1, jehož stav v logické 1 bude podmíněn současným stlačením tlačítka S1, tj. spínacího kontaktu pro povel vytahování, sepnutým stavem rozpínacího kontaktu S3 horního koncového spínače (roleta ještě není v horní poloze), rozepnutým stavem tlačítka S2 pro stahování rolety dolů (softwarové blokování) a nulovou hodnotou výstupu A 6.2, tj. absencí povelu pro stahování rolety dolů (druhé softwarové jističení).
- zavírání:** Povel pro stahování rolety dolů (motor zapnut pro stahování) bude odvozen z výstupu A 6.2 = 1, jehož stav logické 1 bude podmíněn současným stlačením S2 (tj. E 2.2 = 1), sepnutým stavem rozpínacího kontaktu dolního koncového rozpínače S4 (tj. E 2.4 = 1), rozepnutým stavem tlačítka S1 pro vytahování rolety (tj. E 2.1 = 0) a nulovou hodnotou výstupu A 6.1 = 0, tj. absencí povelu pro vytahování rolety.

Podle uvedeného logického popisu je sestavené logické schéma (obr. 2a) a na jeho základě pak program (obr. 2b) a kontaktní schéma (obr. 2c).



Obr. 2 Schémata a program k příkladu 3

Poznámka: Je-li pro testování funkce hotového programu k dispozici testovací PA, který je vybaven jenom spínacími kontakty, je k simulaci rozpínacích koncových kontaktů S3, S4 nutné, aby byly při chodu rolety oba sepnuty a koncová poloha musí být simulována uvolněním (nikoliv stlačením) spínače. Hodnoty signálů je možno kontrolovat podle indikátorů tvořených LED diodami na vstupních i výstupních jednotkách. Svítící dioda indikuje logickou 1, zhasnutá dioda logickou 0.

Ze samotného programu (obr. 2) není zřejmé, zda jednička na vstupu je signálem ze stisknutého spínacího kontaktu nebo signálem z uvolnění rozpínacího kontaktu. PA registruje jen logickou hodnotu a je nepodstatné, zda roleta při nájedu do koncové polohy stlačí rozpínací kontakt, nebo uvolní stlačený spínací kontakt. Podmínkou spuštění motoru pro pohyb rolety dolů je 1 z dolního koncového indikátoru, říkáme, že program učiní **dotaz na 1**. Otázka použití spínače nebo rozpínače je spíše otázkou bezpečnosti provozu. Přerušení kteréhokoliv vodiče (nebo uvolnění jeho kontaktu na svorce) může vést k blokování spuštění motoru, nikoliv však ke zkratu nebo mechanickému poškození zařízení.

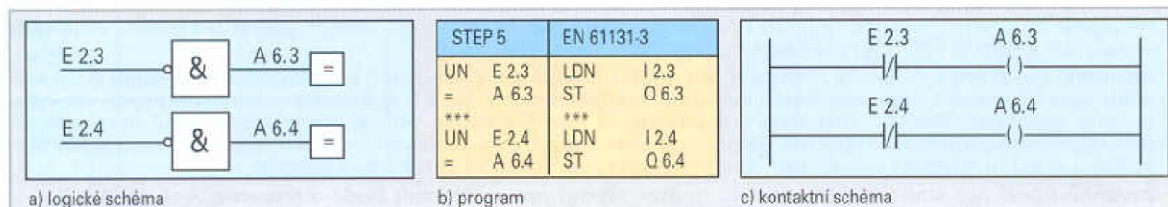
Příklad 4: Zadání příkladu 3 je rozšířeno o požadavek, aby obě krajní polohy vrat dílny byly indikovány světelnou signalizací v kanceláři mistra. Žárovka H1 nechtě svítí při zcela zvednuté roletě a H2 při zcela spuštěné roletě (obr. 1). Sestavte program pro ovládání pomocí PA.



Obr. 1 Ovládání rolety rozšířené o indikaci krajních poloh (příklad 4)

Horní koncový vypínač S3 dává signál 0, je-li roleta zcela otevřena. Dolní koncový vypínač S4 dává při zcela zavřené roletě signál 0.

Logický popis řídicí úlohy zní: Ovládací výstup A 6.3 = 1 když E 2.3 = 0. Ovládací výstup A 6.4 = 1 když E 2.4 = 0. Je možno říci, že při spuštění servomotoru činí program dotaz na nulu. Logické schéma této podmínky je na obr. 2a, odpovídající úsek programu na obr. 2b a kontaktní schéma na obr. 2c.



Obr. 2 Program pro PA k příkladu 4

Při porovnání kontaktního schématu na obr. 2c s kontaktním schématem na obr. 2c, str. 409 je vidět, že vstupy E 2.3 a E 2.4 mají v obou schématech různé symboly, i když je na oba vstupy vždy připojen rozpínací kontakt S3, resp. S4. Kromě toho je vidět, že na obr. 2c, str. 409 je na E 2.1 připojen spínací kontakt S1 a na E 2.3 rozpínací kontakt S3. Oba vstupy však mají v kontaktním schématu stejný symbol. Symboly v kontaktním schématu zde nepředstavují schématické značky určitých prvků jako například ve schématu zapojení, nýbrž zde mají význam logických funkcí: vede = 1, nevede = 0 (tabulka), sériové spojení = konjunkce, paralelní spojení = disjunkce, spínací kontakt = logická proměnná, rozpínací kontakt = negovaná logická proměnná. Kontaktní schéma by odpovídalo skutečnému zapojení, kdyby byl řídicí systém realizován pomocí relé (např. při řízení výtahů).

Tabulka: Aktivace výstupu na 1		
potřebný vstup	0	1
logický člen		
kontaktní člen		
logická operace	UN... ON...	U... O...

Symboly kontaktního schématu neodpovídají při řízení pomocí PA skutečným kontaktům.

Logické schéma by odpovídalo skutečnému zapojení, kdyby byl řídicí systém realizován např. pomocí integrovaných obvodů typu 74xxxx s prvky AND, NAND, OR, NOR a XOR.

Pro sestavení programu pro dnešní PA stačí napsat logické rovnice definující výstupy a strojový program se automaticky vygeneruje a uloží.

Otázky k opakování

1. Popište činnost PA.
2. Co znamená „programovatelný automat“?
3. Na co je třeba dát pozor při střídavém ovládání (opačných pohybů) pomocí PA?
4. Jakou výhodu má PA oproti pevně zapojenému (neprogramovatelnému) řízení?
5. Ve vodárně jsou instalována 3 čerpadla A, B, C. Vždy musí být zapojena právě 2 čerpadla. Činnost čerpadel je indikována snímači proudění B0 (E 2.0) pro čerpadlo A, B1 (E 2.1) pro čerpadlo B a B2 (E 2.2) pro čerpadlo C. Signální žárovka H1 indikuje normální provoz, H2 indikuje poruchu. Nakreslete zapojení PA a napište řídicí program.

17.4.3 Paměťové funkce

Při vyhodnocování logických podmínek pro nastavení řídicích signálů používají PA dvě paměťové funkce, a to funkci upřednostňující zapnutí a funkci upřednostňující vypnutí (obr. 1).

Při současných signálech pro činnost i pro odpojení zařízení uloží funkce upřednostňující 0 do pomocné paměti 0 a funkce upřednostňující 1 uloží do pomocné paměti (nebo na výstup) 1.

Funkce upřednostňující vypnutí se používá při rozběhu strojů a zařízení (impulzem se vypne zařízení a zůstává vypnuto). Funkce upřednostňující zapnutí se používá k vyhodnocení signálu o havarijním stavu.

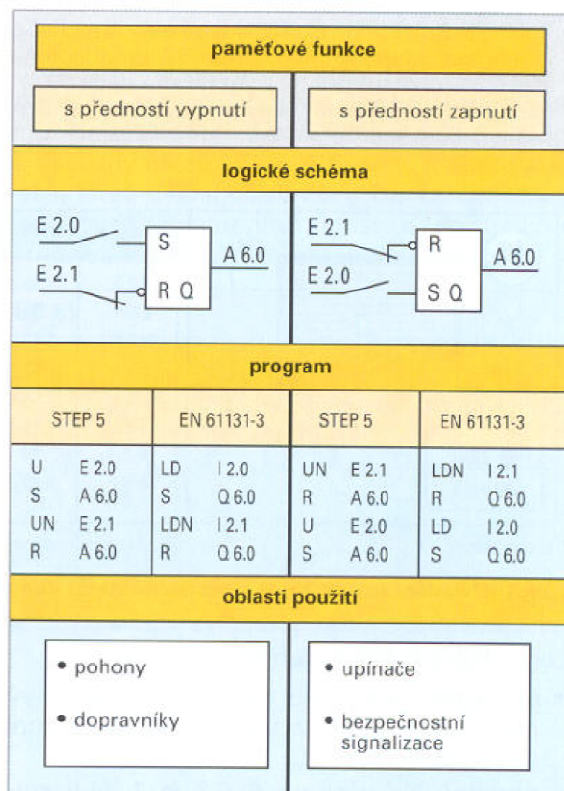
Stroje a zařízení musí být odpojeny ($A\ 6.0 = 0$) nouzovým tlačítkem ($E\ 2.1 = 0$), i když jsou programem zapnuty. Hlásiče nebezpečí nebo poruchy musí zůstat zapnuty, pokud porucha trvá i poté, co skončil signál ($E\ 2.1 = 0$) indikující vznik poruchy.

Paměťová funkce s předností vypnutí se používá např. u strojů a zařízení, paměťová funkce s předností zapnutí se používá např. při signalizaci nebezpečí.

Instrukce UN E 2.1 znamená, že se provede konjunkce negované hodnoty vstupu E 2.1 (obr. 1) s hodnotou v registru a výsledek zůstane uložen v registru. Na začátku činnosti programu je do pracovního registru uložena hodnota operandu první logické instrukce. Přednost je u PA dána pořadím instrukcí v programu. Princip bude vysvětlen na paměťové funkci s předností vypnutí a signály pro běh i vypnutí = 1 (obr. 2). Vzhledem k postupnému zpracování jednotlivých instrukcí (příkazů) programu uloží program nejprve vstup $E\ 2.0 = 1$ do registru, pak přeneseme hodnotu z registru na výstup A 6.0, pak provede konjunkci hodnoty 1 vstupu E 2.1 s registrem a pak přeneseme negovanou (reset) hodnotu 0 registru na výstup.

Při současném splnění podmínek pro nastavení řídicího signálu na 1 i pro nastavení řídicího signálu do 0 je pro nastavení rozhodující naposled vyhodnocená podmínka.

Z důvodu bezpečnosti při přerušení signálního nebo uzemňovacího vodiče jsou stroje a zařízení zapínány signálem 1 a vypínány signálem 0, tedy rozpojovacím kontaktem. Při obsluze bezpečnostní signalizace to platí obráceně, signalizace je zapínána signálem 0 a udržována vypnutá signálem 1.



Obr. 1 Paměťové funkce PA

řádek programu	instrukce	hodnota na vstupu	přesuny operandů	hodnota v registru	hodnota na výstupu
1	U E 2.0	1		1	
2	S A 6.0		S		1
3	U E 2.2	1		1	
4	R A 6.0		R		0

Obr. 2 Činnost programu PA (příklad)

Příklad 5: Roletová vrata dílny jsou otevírána tlačítkem S1 a zavírána tlačítkem S2 (obr. 1a). Horní poloha je hlášena koncovým vypínačem S3, dolní poloha koncovým vypínačem S4. Roleta se úplně otevře po stisku S1, zavírání je v tlučacím režimu (tedy S2 je nutno držet). Otevírání je možno zastavit tlačítkem S0 (stop). Na spodní liště rolety je bezpečnostní rozpínací kontakt, který zastaví zavírání při nárazu na překážku. Sestavte program pro PA.



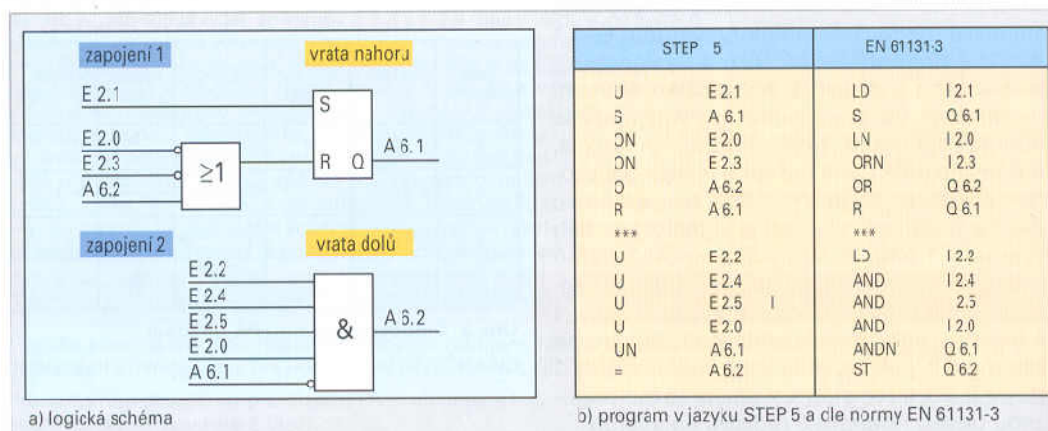
Obr. 1 Ovládání rolety (vrat) s bezpečnostním kontaktem (příklad 5)

Logický popis řídicí úlohy zní:

- **otevírání:** Při signálu 1 z tlačítka S1 je na výstup A 6.1 vyslána 1. Na výstup A 6.1 je nastavena 0, je-li vyslán z tlačítka S0 nebo koncového vypínače S3 signál 0, nebo A 6.2 = 1 (blokování).
- **zavírání:** Na výstupu A 6.2 je 1, je-li současně na S2, S4, S5 a S0 stav 1 a A 6.1 = 0 (blokování).

Z uvedeného popisu lze sestavit logické schéma (obr. 2a) a program (obr. 2b).

Kvůli přehlednosti je logické schéma i program rozdělen na dvě části.



Obr. 2 Logické schéma a program k příkladu 5

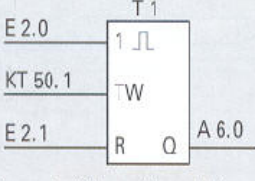
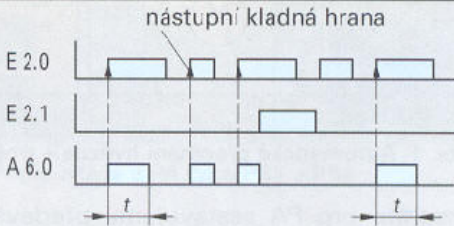
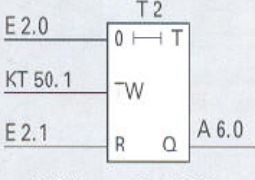
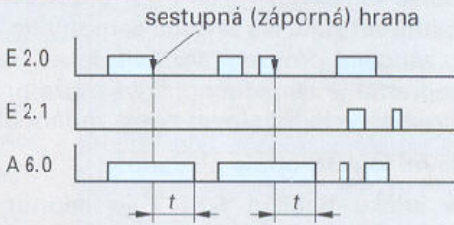
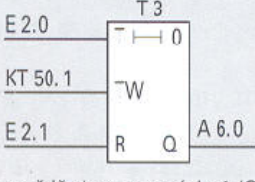
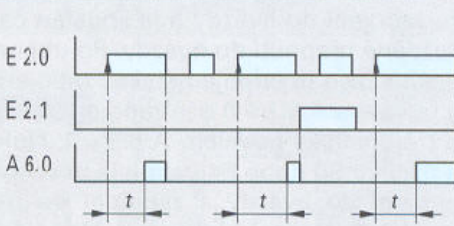
Otázky k opakování

1. Vrata dílny z příkladu 5 je nyní možno zavřít jediným stiskem tlačítka. Z bezpečnostních důvodů je možné jen přepnutí ze zavírání na otevírání. Sestavte logické schéma a program pro PA.
2. Trojfázový motor může být zapínán a vypínán tlačítky ze dvou míst. Určete zapojení PA, formulujte podmínky pro zapnutí a vypnutí motoru. Sestavte logické schéma a program pro PA.

17.4.4 Časové funkce

Časové funkce PA mohou časově omezit nebo zpoždit akce programu (**tabulka 1**). PA disponuje většinou časovými funkcemi různých vlastností, např. generování impulzu, zpoždění při nastavení hodnoty výstupu nebo načtení hodnoty vstupu. Nejzákladnější časovou funkcí je zpoždění načtení vstupu. Z této funkce lze vytvořit následně všechny ostatní časové funkce. V automatickém řídicím systému bývá k dispozici více časovačů, které bývají očíslovány T0, T1, ... Impulz pro zpoždění sepnutí je spuštěn náběžnou hranou startovacího vstupního signálu a zpoždění pro rozpojení (výstup do nuly) je spuštěno sestupnou hranou startovacího vstupního signálu.

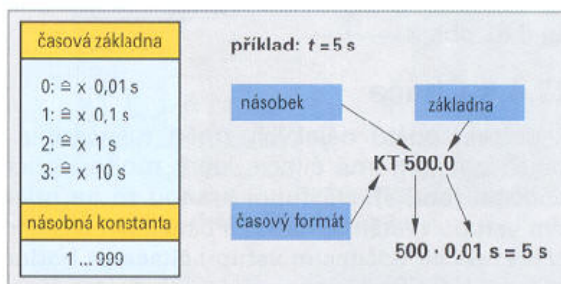
Tabulka 1: Časové funkce PA (výběr z jazyka STEP 5)

logické schéma	program	časový průběh impulzů
 <p>impulz (SI – set impuls)</p>	<pre> U E 2.0 L KT 50.1 SI T 1 U E 2.1 R T 1 U T 1 = A 6.0 </pre>	 <p>nástupní kladná hrana</p>
 <p>zpoždění rozpojení (SA)</p>	<pre> U E 2.0 L KT 50.1 SA T 2 U E 2.1 R T 2 U T 2 = A 6.0 </pre>	 <p>sestupná (záporná) hrana</p>
 <p>zpoždění nastavení do 1 (SE)</p>	<pre> U E 2.0 L KT 50.1 SE T 3 U E 2.1 R T 3 U T 3 = A 6.0 </pre>	

Programování časové prodlevy je stejné pro všechny typy PA. Časový údaj se zadává násobnou konstantou a časovou základnou (**viz obr.**) a čas prodlevy je součinem násobné konstanty a časové základny. Přesnost prodlevy je závislá na zvolené časové základně (**tabulka 2**).

Čím přesněji chceme odměřit časovou prodlevu, tím kratší musí být volena časová základna. Čím menší je časová základna, tím delší bude čas cyklu PA.

V automatickém řídicím systému PS3 nebo PS4 následuje po naprogramování funkce časové konstanty KW časová prodleva v desetinách sekundy. Časová prodleva délky např. 5 s je naprogramována pomocí konstanty KW 50.



Obr. Časový formát pro PA: Simatic S5

Tabulka 2: Tolerance časových funkcí

časová základna	přiřazení konstant	časové rozpětí	tolerance
0 = 0,01 s	$t = 5 \text{ s}$ KT 500.0	4,99 s ... 5,00 s	-0,01 s
1 = 0,1 s	$t = 5 \text{ s}$ KT 50.1	4,9 s ... 5,0 s	-0,1 s
2 = 1 s	$t = 5 \text{ s}$ KT 5.2	4 s ... 5 s	-1 s
3 = 10 s	$t = 50 \text{ s}$ KT 5.3	40 s ... 50 s	-10 s

Příklad 6: Automatické přepínání z hvězdy do trojúhelníku při rozběhu motoru je realizováno pomocí PA. K přepnutí dochází 4 sekundy po zapnutí. Motor se zapíná tlačítkem S1 a vypíná se tlačítkem S0. Při přetížení je motor vypnut pomocí nadproudového relé F4. Sestavte řídicí program pro PA.



Obr. 1 Automatické přepínání hvězda – trojúhelník (příklad 6)

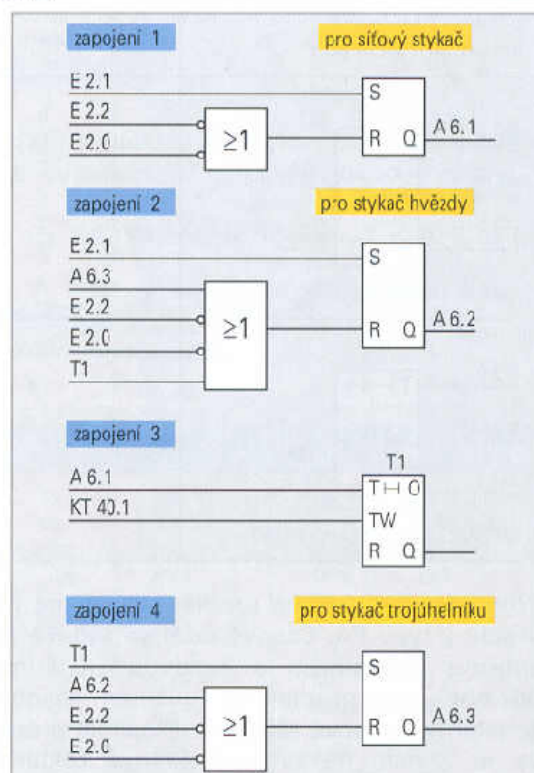
Program pro PA sestavujeme především podle popisu logických vazeb mezi požadovanými činnostmi zařízení. Na základě samotného elektrického zapojení program sestavit nelze. Sestavování programu je usnadněno, když může program jen formálně vyjádřit slovní popis zadání úlohy.

Slovní logický popis příkladu:

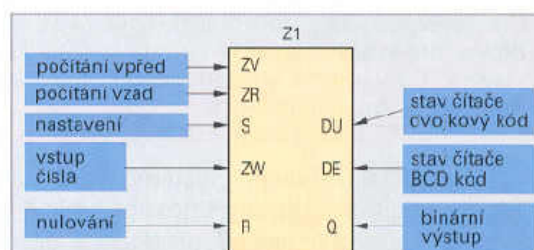
Po stisku tlačítka S1 = 1 je sepnut povel A 6.1 = 1 síťový stykač a povel A 6.2 = 1 stykač pro zapojení do hvězdy a je spuštěn časovač pro zpožděné přepnutí do hvězdy. Po uplynutí nastaveného času je odpojen stykač zapojení do hvězdy povel A 6.2 = 0 a je zapojen stykač zapojení do trojúhelníka povel A 6.3 = 1. Nulový signál na tlačítko S0 nebo F4 vynuluje všechny výstupy. Zapojení do hvězdy a zapojení do trojúhelníka musí být vzájemně blokována. Na základě uvedeného popisu je sestaveno logické schéma příkladu 6 na obr. 2.

17.4.5 Čítače

K počítání počtu nějakých změn na vstupu PA slouží obousměrné čítače, které mohou přičítat i odčítat (obr. 3). Nástupní hranou se na přičítacím vstupu zvětšuje hodnota čítače o 1 a nástupní hranou na odčítacím vstupu čítače se hodnota snižuje. Nástupní hranou pulzu na nastavovacím vstupu (SET) přebere čítač hodnotu čísla na vstupu. Nástupní hranou signálu na nulovacím vstupu (RESET) je hodnota čítače nastavena na 0. Binární výstup (Q) je roven 1 při nenulovém stavu čítače a je roven 0 při nulovém stavu čítače. Při zpětném čítání od nastavené hodnoty se výstup Q změní z 1 na 0 při dosažení nulového stavu čítače. Tato změna se využívá pro přepnutí výstupního signálu.



Obr. 2 Program PA k příkladu 6



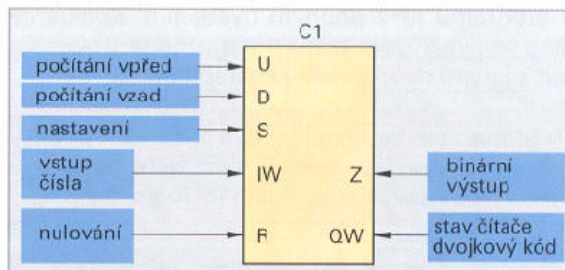
Obr. 3 Popis čítače PA Simatic pro program STEP 5 a STEP 7

Aktuální stav čítače PA Simatic S5 (nebo S7) je možno přečíst na výstupu DU (dualcodiert), a na výstupu DE (BCD – dezimalcodiert) (obr. 3, str. 414). Stav čítače je možno uložit pro další zpracování do příznakových pamětí MW (Merkerworten). Příznakové paměti mají 16 bitů ve 2 bytech MB (Merkerbyte) (obr. 1). Řídicí systémy PS-3 a PS-4 (obr. 2) mají jen 2 výstupy (nemají BCD výstup) a mají jiné značení (obr. 2).

Příklad 7: Je třeba napsat řídicí program pro PA, který bude řídit tester ve zkušební laboratoři továrny na nábytek. Tester opakovaně vysune a zasune nábytkovou zásuvku. Tlačítkem S1 se má spustit cyklus 500 otáček excentrického vytahovacího mechanismu. Kontrolní žárovka H1 má indikovat běh testu. Koncový spínač S2 indikuje zasunutí zásuvky. Tlačítkem S0 lze testování přerušit. Při novém startu započne opět automatický cyklus 500 opakování zasunutí (obr. 3).

MW 2 (pomocná paměť 2)							
MB 2 (M-byte 2)				MB 3 (M-byte 3)			
M 2.7	...	M 2.1	M 2.0	M 3.7	...	M 3.1	M 3.0
a) dělení slova MW 2 u PA Siemens S5, S7							
MW 2 (pomocná paměť 2)							
M 2.15	...	M 2.9	M 2.8	M 2.7	...	M 2.1	M 2.0
b) dělení slova MW 2 u PA typu PS 3 a PS 4							

Obr. 1 Dělení pomocné paměti PA



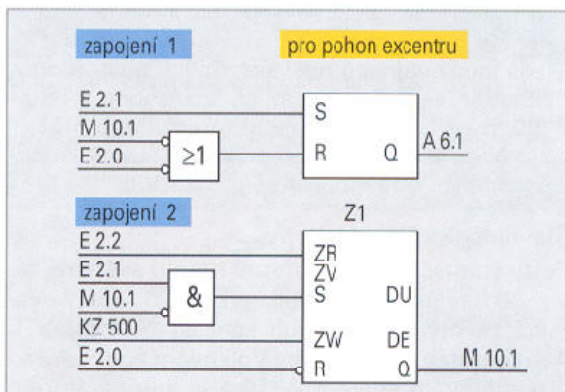
Obr. 2 Popis čítače u PA typu PS3 a PS4



Obr. 3 Řízení testeru k příkladu 7

Logický popis úlohy zní:

Výstup A 6.1 bude nastaven na 1 (A 6.1 = 1), když z tlačítka S1 přijde 1 (S1 = 1). Výstup A 6.1 bude nulován po 500 cyklech nebo po signálu 0 z rozpínače S0. Čítač je programován na odčítání po 1 od nastavené hodnoty 500, která je nastavena při S1 = 1 a současně při nulové hodnotě čítače. Pokud bychom nastavení čítače nevázáli na jeho nulování, mohl by být opakovaně nastavován na 500 i před koncem testu. Z tohoto popisu lze nakreslit logické schéma (obr. 4).



Obr. 4 Logické schéma k příkladu 7

Otázka k opakování

Centrifuga běží po zapnutí nízkými otáčkami n_1 a může být nejdříve po 10s přepojena na vyšší otáčky n_2 . Kontrolní žárovka H1 indikuje připravenost k přepnutí. Nadproudové relé F2 chrání vinutí motoru při vysokých otáčkách, relé F3 chrání vinutí motoru při nízkých otáčkách. Centrifugu lze kdykoliv vypnout vypínacím tlačítkem.

- sestavte tabulku přiřazení operandů,
- nakreslete schéma připojení PA,
- formulujte logické podmínky pro zapnutí a vypnutí otáček n_1 , n_2 ,
- sestavte program pro PA.



K1: stykač pro nízké otáčky n_1
K2: stykač pro vysoké otáčky n_2