

Esperanto programátorů PLC: programování podle normy IEC/EN 61131-3 (část 10)

V předchozích částech tohoto seriálu byl výklad zaměřen na logické funkce, jejichž výstup byl jednoznačně určen okamžitými hodnotami jejich operandů a jejich aktuální kombinací. Logické funkce s touto vlastností se nazývají kombinační logické funkce. V praxi se lze ale častěji setkat s logickými funkcemi, které na shodné kombinace vstupů mohou reagovat různě, podle aktuálního stavu, v závislosti na čase a na předchozí posloupnosti (sekvenci) hodnot vstupů. Tato skupina logických funkcí se souhrnně nazývá sekvenční logické funkce.

Sekvenční logické funkce v řízení budov

Se sekvenčními funkcemi se lze často setkat v technice budov, nejčastěji při řešení časových závislostí, při obsluze mžikových ovládacích prvků nebo krátkodobých senzorů. Komplikovanější funkce jsou potřebné pro řízení složitějších mechanismů, mříží, vrat, žaluzií, markýz, ale i pro simulaci „života“ v prázdném domě.

Tlačítka s nestálými polohami a senzory PIR

Při řešení „chytré elektroinstalace“ se jen výjimečně používají tradiční vypínače nebo přepínače se stabilními polohami, které byly dosud brány v potaz. Téměř výhradně se zde používají ovládací prvky (vypínače) s krátkodobým (mžikovým) sepnutím kontaktu, které se po uvolnění stisku vracejí do klidové polohy, v níž není žádný kontakt sepnut. Dodavatelé prvků pro inteligentní instalaci nabízejí mnoho typů takovýchto ovládacích prvků v různém provedení (např. s jedním, dvěma nebo čtyřmi hmatníky). Jednotlivé prvky jsou často v podobě tlačítek s jednou aktivní polohou (obvykle horní), kde druhá poloha je klidová, bez spínací funkce. Tlačítka bývají obvykle s kolíbkovými hmatníky. Mohou ale být obdobou jednoduché klávesnice, např. se čtvercovými nebo obdélníkovými hmatníky, maticově uspořádanými. Mnohdy jsou používány ovládací prvky s kolíbkovými hmatníky, z nichž každý má dvě aktivní polohy (horní a dolní), v každé je spínán oddělený kontakt. Klidová poloha je uprostřed mezi nimi. Někdy jsou doplněny signály LED nezávisle ovládané z PLC. Tyto ovládací prvky budou uvažovány v prvních příkladech. Každé takové tlačítko předává do PLC dva dvouhodnotové vstupy (pro horní a pro dolní polohu) a přijímá jeden dvouhodnotový výstup pro ovládání signály LED. Dvojnásobnému prvku odpovídají čtyři binární vstupy a dva binární výstupy, pro čtyřnásobný prvek to je osm binárních vstupů a čtyři binární výstupy. Sepnutí kontaktu pro kteroukoliv polohu je tedy krátkodobou (většinou mžikovou) událostí.

Program obsluhy těchto ovládacích prvků je sice logicky komplikovanější než u tradičních vypínačů se stálými polohami (dlouhodobě sepnutými kontakty), ale programátor má více možností, jak je upravit. Jedním tlačítkem lze předat více informací – např. rozlišením ojedinelého a opakovaného stisku („dvojitého kliknutí“), krátkého nebo dlouhého stisku, popř. jejich kombinace a posloupnosti. Tvořivosti se meze nekladou, „zarytý telegrafista“ může např. svou domácnost ovládat povely zapsanými v Morseově abecedě. Možné je leccos, důležitá je ale především

názorná, přívětivá a spolehlivá obsluha, aby dům svou „přílišnou inteligencí“ nestresoval své obyvatele nebo jejich hosty.

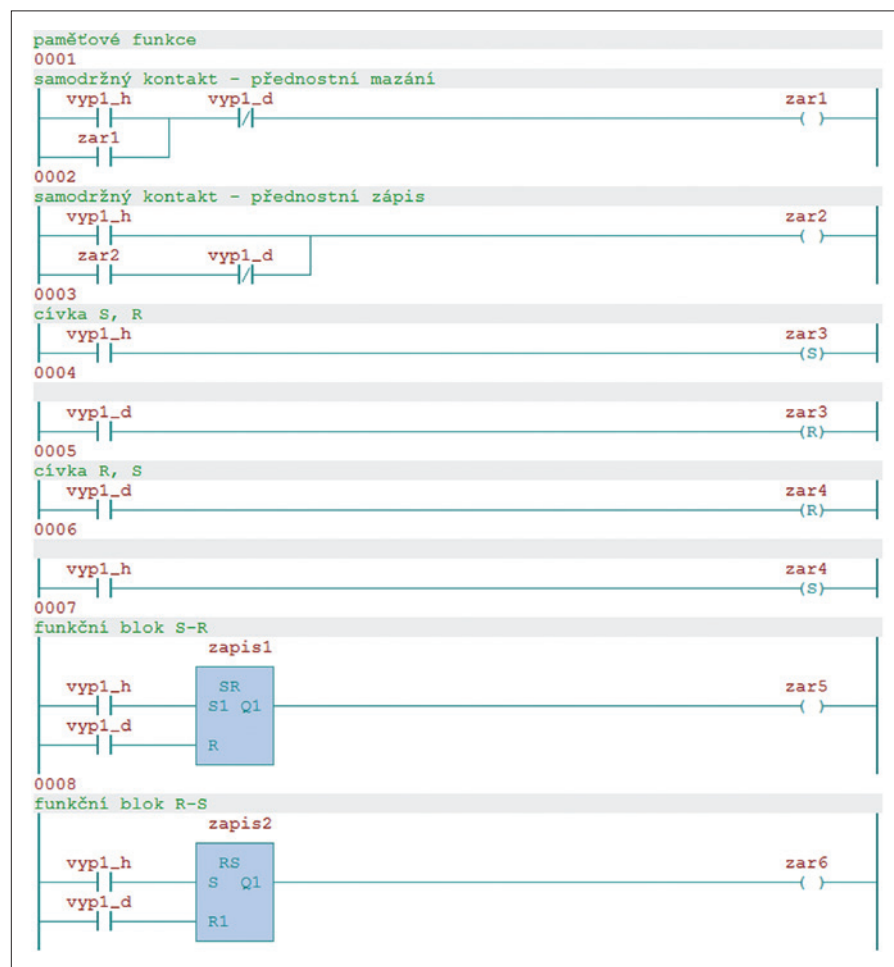
Déletrvajícím, ale rovněž krátkodobě bývá i sepnutí kontaktu infračerveného senzoru pohybu PIR. Mylně je považován za senzor přítomnosti osoby, ve skutečnosti je ale senzorem pohybu osoby. Osobu zaregistruje, jen když je v zorném poli snímače; kdyby se zde ale osoba na čas zastavila, přestane ji po určité době registrovat, dokud se opět nepohne.

Příklad 11: paměť tlačítka

Je požadováno stiskem tlačítka do horní aktivní polohy rozsvítit světlo, které má svítit do doby, než bude totéž tlačítko stisknuto do opačné (dolní) aktivní polohy.

Řešení 1: zpětnovazební zapojení kontaktů (samodržný kontakt)

Na obr. 29 (obvod 0001) je kontaktní schéma možného řešení zapsané v jazyku LD. Tento postup je obvyklý v reléové technice,



Obr. 29. Řešení paměťové funkce RS a SR v jazyku LD

kde je většinou označováno jako „relé se samodrzným kontaktem s přednostním zápisem“. Pro popisované zadání je rovnocenné i řešení označené jako obvod 0002. Rozdílnost obou variant by se projevila, pokud by byly současně tisknuty oba kontakty $vyp1_h$ a $vyp1_d$, což mechanický vypínač neumožňuje. V tom případě by podle prvního řešení byla žárovka vypnutá, zatímco podle druhého řešení by svítila. Proto se první zapojení nazývá *paměťový obvod s přednostním mazáním (nulováním)* nebo podle normy *bistabilní prvek (klopný obvod) s dominantním mazáním (vypnutím)*. Druhé řešení se nazývá *paměťový obvod s přednostním zápisem (nastavením)* nebo podle normy *bistabilní prvek (klopný obvod) s dominantním nastavením (sepnutím)*.

Poznámka: Bude-li program laděn v systému Mosaic pro virtuální PLC, dvojitým poklepáním na levé tlačítko myši bude změněn stav dvouhodnotové proměnné (typu BOOL) pod kurzorem na opačný. Takto je možné nastavit libovolné hodnoty proměnných $vyp1_h$ a $vyp1_d$, tedy i obě jedničkové (pravdivé, s hodnotou TRUE). V tomto režimu ladění je třeba více pozornosti, protože se pracuje s obecnějšími prvky, než jsou uvažované mžikové spínače. To platí i pro ladění dalších příkladů a zadaných úloh.

Zpětná vazba

Všimněme si, že v obou případech se výstupní proměnná $zar1$ nebo $zar2$ vyskytuje i mezi kontakty vstupních proměnných $vyp1_h$ a $vyp1_d$. Je tak realizována *zpětná vazba* z výstupu logického obvodu na jeho vstup. Ta je podstatou paměťového chování. Obyčejné typy paměťového obvodu je možné v jazyku ST popsat logickými výrazy:

```
zar1 := (zar1 OR vyp1_h) AND NOT vyp1_d ;
zar2 := (zar2 AND NOT vyp1_d) OR vyp1_h ;
```

V prvním výrazu se výstupní proměnná $zar1$ vyskytuje na obou stranách rovnítky. Ve skutečnosti nejde o identickou proměnnou – symbol $zar1$ sice označuje stejné místo v paměti, ale hodnoty se liší časovou aktualností. Výskyt symbolu na pravé straně výrazu označuje hodnotu proměnné, která bude

použita pro vyčíslení výrazu, zatímco stejný symbol na levé straně označuje právě vyčíslenou hodnotu, která bude na označené paměťové místo uložena. Fakticky je příkazem realizována rovnost:

$$zar1_k = (zar1_{k-1} \text{ OR } vyp1_h) \text{ AND NOT } vyp1_d,$$

kde

$zar1_{k-1}$ označuje hodnotu proměnné $zar1$ před vyčíslením příkazu, $zar1_k$ označuje její nově vyčíslenou hodnotu.

Podobně druhý příkaz vyjadřuje rovnost:

$$zar2_k := (zar2_{k-1} \text{ AND NOT } vyp1_d) \text{ OR } vyp1_h$$

Zpětnými vazbami lze realizovat i podstatně složitější sekvenční logické funkce. Pak se mluví o realizaci asynchronních logických obvodů. Tento přístup má mnoho nevýhod a v praxi se již nevyužívá. Řešení se samodrznými kontakty je zde uvedeno jen proto, aby byla tato skutečnost objasněna. Dále uváděná řešení jsou formálně jednodušší a k realizaci sekvenčních funkcí používají standardní „prefabrikáty“ – kontakty a cívky se specializovanými funkcemi nebo funkčními bloky sekvenčního charakteru. Při bližším zkoumání je ale možné zjistit, že i tyto „prefabrikáty“ jsou vnitřně realizovány s využitím zpětných vazeb.

Řešení 2: využití cívek Set a Reset

Paměťovou funkci s *dominantním mazáním (vypnutím)* podle obvodu 0001 z obr. 29 lze rovnocenně realizovat dvojicí obvodů 0003 a 0004. Využívají paměťové chování výstupních cívek typu Set (cívka s vepsaným symbolem S) a Reset (cívka s vepsaným symbolem R). Řešení je formálně jednodušší a přehlednější. Obdobně je paměťová funkce s *dominantním nastavením (sepnutím)* podle obvodu 0002 realizovaná dvojicí obvodů 0005 a 0006. Obě řešení se liší jen pořadím větvi s paměťovými výstupními cívkami. V případě sporu, kdy se vyskytnou současně požadavky na zapnutí a vypnutí, platí, že poslední paměťová funkce v programu (ve schématu) je převažující („poslední má pravdu“).

Výstupní cívky Set a Reset

Nově se zde objevuje důležitý prvek jazyka LD – paměťové výstupní cívky typu Set a Reset. Obě mění stav své výstupní pro-

měnné, jen jestliže k nim zleva přichází pravdivý signál – sepnuto. V tom případě cívka Set nastavuje svůj výstup na hodnotu pravda (TRUE, sepnuto), zatímco cívka Reset svůj výstup nastavuje na hodnotu nepravda (FALSE, rozepnuto). V opačném případě (pokud k cívkám přichází nepravdivý signál – rozepnuto) nemění cívky stav své proměnné – pamatují si naposledy nastavenou hodnotu.

Řešení 3: využití standardních funkčních bloků RS a SR

Funkčně rovnocenná řešení nabízejí i obvody 0007 a 0008 z obr. 29. Používají standardní funkční blok z knihovny STDLIB, který realizuje funkci bistabilního paměťového prvku (klopného obvodu) typu RS s dominantním mazáním (vypnutím) nebo SR s dominantním nastavením (sepnutím).

Funkční bloky RS a SR

Knihovna standardních funkčních bloků je popsána v příručce [12]. Typ funkčního bloku (RS, SR) je uveden uvnitř značky nahore, jméno jeho aktuálního použití (instance) je uvedeno nad značkou ($zapis1$, $zapis2$). Vstupy funkčního bloku (S, R1, R, S1) jsou uvedeny uvnitř u levého okraje značky – R1 a S1 jsou vstupy pro dominantní mazání a dominantní nastavení. Výstup (stav paměťového obvodu) je označen jako Q1 uvnitř u pravého okraje značky.

Úloha 44: Skupina tlačítek

Předpokládejme „skupinový vypínač“ se čtyřmi mžikovými tlačítky, která fungují stejně jako v předchozím příkladu 11. První tři vypínače zleva jsou určeny k individuálnímu ovládní svítidel v místnosti (např. orientační, intenzivní a svítidlo u křesla). Stiskem do horní polohy se svítidlo aktivuje, stiskem do dolní polohy se zhasne. Pravý vypínač ovládá celou skupinu svítidel naráz – stiskem do horní polohy se všechna svítidla aktivují, stiskem do dolní polohy se všechna vypnou. Realizujte program v LD pro popsanou obsluhu svítidel.

Úloha 45: Schodišťový vypínač – náhrada tradičního řešení

Předpokládejme, že na každém konci schodiště jsou umístěny jednoduché mžikové spínače. Vytvořte program v LD, který je vyhodnotí jako schodišťový vypínač.

Ladislav Šmejkal

**Připravujeme
veletržní vydání
Automa 8-9/2012**

Časopis ve zvýšeném nákladu bude k dispozici na MSV v Brně (10. až 14. 9. 2012). I firmy, které nebudou vystavovat, mohou dát o sobě vědět.

- upoutávky na veletržní stánky v předstihu na www.automa.cz
- témata vydání: automatizace slévárenství a plastikářské výroby pneumatické a hydraulické pohony
- přehled trhu: pneumatické ventilové terminály

Kontakt: automa@fccgroup.cz