

RsLogix5000 - Časovače a počítadlá

Körösi Ladislav · Elektrotechnika

01.10.2014



Časovače a počítadlá patria do skupiny funkčných blokov, ktoré majú vlastnú inštanciu (dátovú štruktúru). Časovače sú dôležité v aplikáciách kedy je dôležitý cyklus stroja alebo keď sú potrebné oneskorenia medzi sekvenciami algoritmu. Tieto inštrukcie sú výstupné inštrukcie a vznikli s cieľom nahradenia časových relé. Počítadlá slúžia na počítanie udalostí alebo počtov v procese ako napr. počet výrobkov, porúch, zmien nastavení a iné. Nahrádzajú základné matematické inštrukcie ako ADD a SUB kvôli ich dobre prepracovanej dátovej štruktúre. Cieľom článku je oboznámiť čitateľa s časovačmi a počítadlami softvéru RsLogix 5000 v jazyku rebríkových schém. Článok je pokračovaním článkov „RsLogix5000 - Bitové inštrukcie“ a „RsLogix5000 - Matematické inštrukcie“.

Časovače a počítadlá používajú štruktúrované dátové typy TIMER a COUNTER. Prístup k štruktúre je cez bodkovú syntax vo forme Názov.Parameter. Na úvod si všeobecne opíšeme dátové štruktúry časovačov a počítadiel a neskôr konkrétne inštrukcie, ktoré využívajú tieto dátové štruktúry. Štruktúra dátového typu TIMER je nasledovná:

- .EN - Časovač je povolený (Bit „Timer enabled“). Bit má hodnotu TRUE, ak je na vstupe časovača platná podmienka.
- .DN - Časovač skončil (Bit „Timer done“). Stavom TRUE alebo FALSE indikuje či časovač skončil, t.j. či aktuálna hodnota časovača je väčšia alebo rovná prednastavenej hodnote (.ACC>=.PRE).
- .TT - Časovač ráta (Bit „Timer timing“). Ak má bit .TT hodnotu TRUE, potom časovač je povolený, ráta a ešte neskončil.
- .PRE - Prednastavená hodnota časovača („Preset value“ - dátového typu DINT). Zadáva sa v milisekundách. Všetky časovače v RsLogix 5000 majú časovú základňu 1 ms.
- - .ACC - Aktuálna hodnota časovača („Accumulated value“ - dátového typu DINT). Rovnako ako hodnota .PRE je v milisekundách. Ak je hodnota .ACC>=.PRE, potom sa hodnota .ACC nemení lebo časovač skončil. Aktuálny čas sa ráta na základe vzťahu $ACC = ACC + (\text{aktuálny čas} - \text{čas z predchádzajúceho cyklu})$.

Štruktúra dátového typu COUNTER je nasledovná:

- .CU - Pripočítanie povolené (Bit „Count up“). Pri nábežnej hrane signálu sa zvýši aktuálna hodnota počítadla o 1.
- .CD - Odpočítanie povolené (Bit „Count down“). Pri nábežnej hrane signálu sa zníži

aktuálna hodnota počítadla o 1.

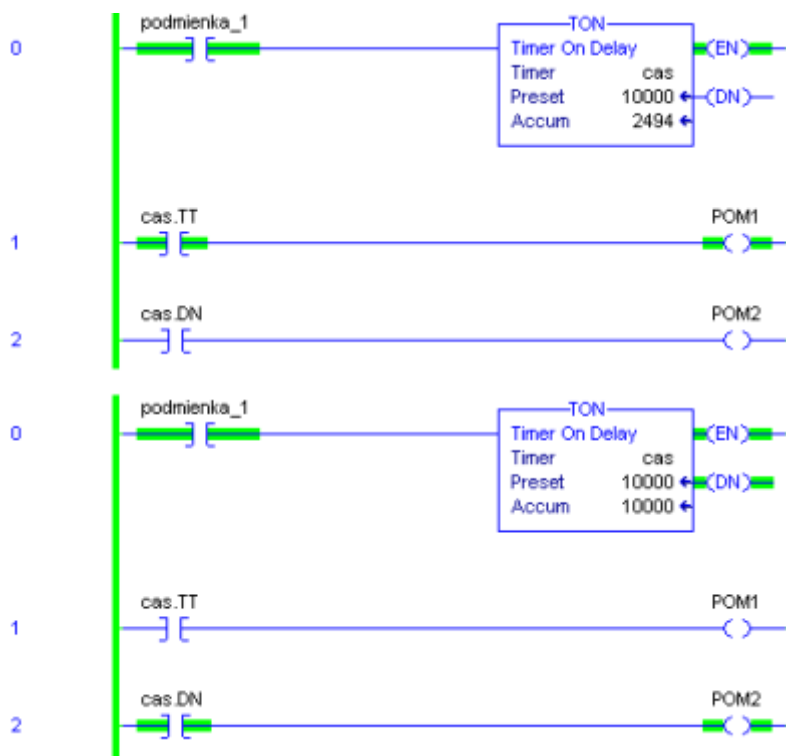
- .DN - Počítadlo dosiahlo prednastavenú hodnotu. Bit má stav TRUE, ak .ACC >= .PRE.
- .OV - Pretečenie počítadla (Bit „Overflow“). Ak je aktuálna hodnota počítadla 2 147 483 648 a na vstupe .CU je požiadavka na zvýšenie aktuálnej hodnoty časovača, tak sa hodnota v počítadle preklopí na -2 147 483 647 a nastaví sa príznak .OV na TRUE.
- .UN - Podtečenie počítadla (Bit „Underflow“). Ak je aktuálna hodnota počítadla -2 147 483 648 a na vstupe .CD je požiadavka na zníženie aktuálnej hodnoty časovača, tak sa hodnota v počítadle preklopí na 2 147 483 647 a nastaví sa príznak .UN na TRUE.
- .PRE - Prednastavená hodnota počítadla dátového typu DINT).
- .ACC - Aktuálna hodnota počítadla (dátového typu DINT).

Inštrukcia „Timer On Delay“ (TON)

Časovač TON je neretentívny časovač, ktorý pri platnej vstupnej podmienke ráta čas od 0ms po prednastavený čas uložený v parametri ACC. Po spustení časovača a uplynutí prednastaveného času sa nastaví bit DN na TRUE. Časovač TON sa volá „Timer On Delay“ preto lebo oneskorí „zapnutie“ signálu DN po čase PRE. Ak sa počas rátania času alebo po ukončení rátania privedie na vstup časovača signál FALSE, tak sa časovač vynuluje (t.j. DN = FALSE, TT = FALSE, EN = FALSE, ACC = 0). Postupnosť stavov časovača TON:

- V stave PRESCAN je výstupom inštrukcie logická nula (FALSE) a parametre sú nastavené nasledovne: DN = FALSE, TT = FALSE, EN = FALSE, ACC = 0.
- Ak je stav rebríka FALSE, potom výstupom inštrukcie je FALSE a parametre sú nastavené rovnako ako stave PRESCAN: DN = FALSE, TT = FALSE, EN = FALSE, ACC = 0.
- Ak je stav rebríka TRUE, potom časovač zvyšuje svoju aktuálnu hodnotu až kým neplatí podmienka ACC >= PRE.
- V stave POSTSCAN je výstupom inštrukcie logická nula (FALSE).

Na obr. 1 sú dve ukážky časovačov TON s názvom tagu cas a s prednastaveným časom 10s. V prvom (hornom) príklade je časovač spustený lebo vstupná podmienka inštrukcie je TRUE (Inštrukcia XIC s parametrom podmienka_1 dáva signál TRUE). Aktuálny čas časovača je 2494ms. Nakoľko časovač je spustený a ešte ráta, tak bit TT má stav TRUE a bit DN stav FALSE. V druhom príklade (nižšie) časovač skončil lebo bol dosiahnutý prednastavený čas 10s (ACC = 10000). Bit DN má hodnotu TRUE lebo časovač dorátal a bit TT má hodnotu FALSE lebo už neráta.



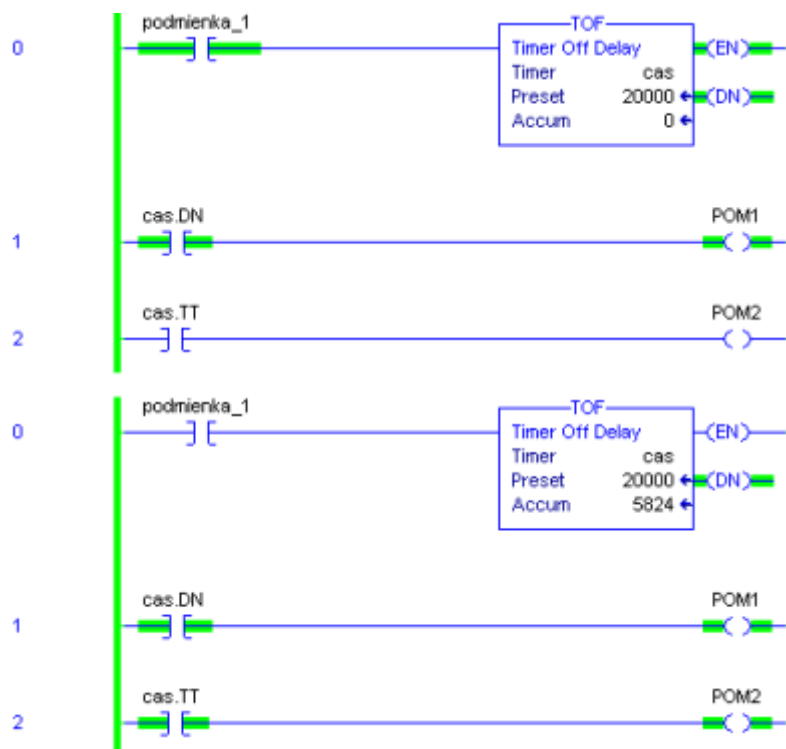
Obr. 1 Ukážka časovača TON v jazyku LD

Inštrukcia „Timer Off Delay“ (TOF)

Inštrukcia TOF je neretenčný časovač, ktorý sa spustí pri zmene vstupného signálu z TRUE na FALSE. Časovač TOF sa preto volá „Timer Off Delay“ lebo po uplynutí prednastaveného času sa bit DN, ktorý mal dovtedy hodnotu TRUE zmení na FALSE, t.j. bit DN sa „vypne“. Postupnosť stavov časovača TOF:

- V stave PRESCAN je výstupom inštrukcie logická nula (FALSE) a parametre sú nastavené nasledovne: DN = FALSE, TT = FALSE, EN = FALSE a aktuálnej hodnote ACC sa priradí prednastavená hodnota PRE.
- Ak je stav rebríka FALSE (zmena z TRUE na FALSE), tak časovač spustí a ráta čas po prednastavenú hodnotu.
- Ak je stav rebríka TRUE, potom má časovač nasledovné stavy: EN = TRUE, DN = TRUE, TT = FALSE a ACC = 0. Výstupom inštrukcie TRUE.
- V stave POSTSCAN je výstupom inštrukcie logická nula (FALSE).

Na obr. 2 sú dve ukážky časovačov TOF s názvom tagu cas a s prednastaveným časom 20s. V hornom príklade je na vstupe časovača hodnota TRUE, preto časovač „čaká“ na zmenu podmienky z TRUE na FALSE, aby sa spustil časovač. V príklade nižšie došlo k zmene vstupnej podmienky časovača na FALSE a preto je časovač spustený. Aktuálny čas časovača je 5824ms, DN = 1 lebo časovač ešte neskončil a TT = TRUE lebo časovač pracuje.



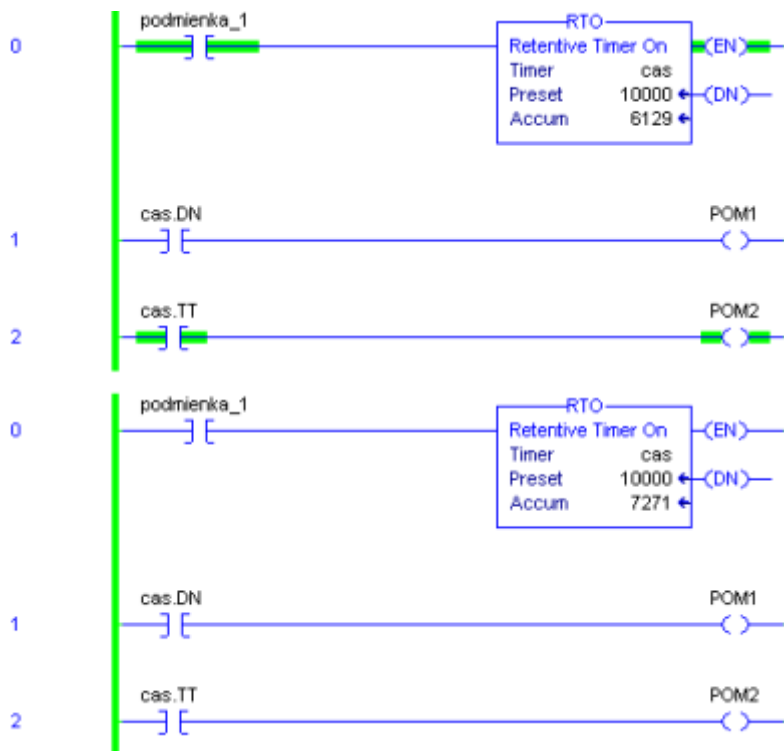
Obr. 2 Ukážka časovača TOF v jazyku LD

Inštrukcia „Retentive Timer On“ (RTO)

Inštrukcia RTO je retenčný časovač, ktorý sa spustí pri zmene vstupného signálu z FALSE na TRUE. Časovač RTO sa líši od časovača TON tým, že po zmene vstupného signálu z TRUE na FALSE sa časovač nevynuluje. Časovač si pamätá aktuálnu hodnotu a pri jeho opätovnom spustení pokračuje v rátaní od aktuálnej hodnoty po prednastavenú. Postupnosť stavov časovača RTO:

- V stave PRESCAN je výstupom inštrukcie logická nula (FALSE) a parametre sú nastavené nasledovne: DN = FALSE, TT = FALSE, EN = FALSE a aktuálnej hodnote ACC sa priradí prednastavená hodnota .PRE.
- Ak je stav rebríka FALSE, potom má časovač nasledovné stavy: EN = FALSE, DN = FALSE, TT = FALSE a hodnota ACC sa zachová. Výstupom inštrukcie FALSE.
- Ak je stav rebríka TRUE, tak časovač začne rátať čas.
- V stave POSTSCAN je výstupom inštrukcie logická nula (FALSE).

Na obr. 3 sú dve ukážky časovačov TOF s názvom tagu cas a s prednastaveným časom 20s. V hornom príklade je na vstupe časovača hodnota TRUE, preto časovač „čaká“ na zmenu podmienky z TRUE na FALSE, aby sa spustil časovač. V príklade nižšie došlo k zmene vstupnej podmienky časovača na FALSE a preto je časovač spustený. Aktuálny čas časovača je 5824ms, DN=1 lebo časovač ešte neskončil a TT = TRUE lebo časovač pracuje.

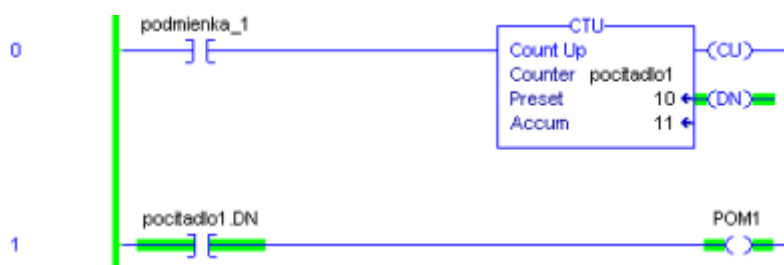


Obr. 3 Ukážka časovača RTO v jazyku LD

Inštrukcia „Count Up“ (CTU)

Inštrukcia CTU je počítadlo (Obr. 4), ktoré pri zmene vstupného signálu z FALSE na TRUE zvýši svoju aktuálnu hodnotu o 1. Inštrukcia CTU na rozdiel od inštrukcie ADD nepotrebuje dodatočnú inštrukciu na detekciu nábežnej hrany (napr. ONS), aby sa zvýšenie vykonalo len raz lebo si to inštrukcia rieši interne. Po dosiahnutí prednastavenej hodnoty je bit DN nastavený na TRUE. Nulovanie počítadla je možné pomocou inštrukcie MOV, ktorým napr. vložíme do aktuálnej hodnoty časovača hodnotu 0 alebo s použitím inštrukcie RES. Postupnosť stavov počítadla CTU:

- V stave PRESCAN je bit CU nastavený na hodnotu TRUE, aby sa predišlo nežiadajúcim chybám a výstup inštrukcie je nastavený na logickú nulu.
- Ak je stav rebríka TRUE (zmena z FALSE na TRUE), potom sa k aktuálnej hodnote počítadla ACC pripočíta hodnota 1.
- Ak je stav rebríka FALSE, tak je výstupom inštrukcie FALSE a bit CU je nastavený tiež na FALSE.
- V stave POSTSCAN je výstupom inštrukcie FALSE.



Obr. 4 Ukážka počítadla CTU v jazyku LD

Inštrukcia „Count Down“ (CTD)

Inštrukcia CTD je počítadlo (Obr. 5), ktoré pri zmene vstupného signálu z FALSE na

TRUE zníži svoju aktuálnu hodnotu o 1. Bit DN je nastavený na TRUE v prípade, že aktuálna hodnota je väčšia alebo rovná prednastavenej aj keď sa jedná o počítadlo, ktoré znižuje svoju hodnotu. Postupnosť stavov počítadla CTD:

- V stave PRESCAN je bit .CD nastavený na hodnotu TRUE z rovnakého dôvodu ako bit CU pri inštrukcii CTU.
- Ak je stav rebríka TRUE (zmena z TRUE na FALSE), potom od aktuálnej hodnoty počítadla ACC odčíta hodnota 1.
- Ak je stav rebríka FALSE, tak je výstupom inštrukcie FALSE a bit CD je nastavený tiež na FALSE.
- V stave POSTSCAN je výstupom inštrukcie FALSE.



Obr. 5 Ukážka počítadla CTD v jazyku LD

Inštrukcia „Reset“ (RES)

Inštrukcia RES vynuluje časovač, počítadlo alebo štruktúru CONTROL, ku ktorej je priradený príslušný programový blok. Inštrukcia nuluje aktuálnu hodnotu časovača, počítadla a POS hodnotu štruktúry CONTROL spolu so príslušnými riadiacimi a stavovými bitmi. Z toho dôvodu sa neodporúča použiť inštrukciu RES pre časovač TOF. Parametrom inštrukcie je názov štruktúry, ktorá sa má vynulovať. Príklad použitia inštrukcie pre nulovanie počítadla je na obr. 6. Postupnosť stavov RES je:

- V stave PRESCAN je výstupom inštrukcie FALSE.
- Ak je stav rebríka FALSE, tak je výstupom inštrukcie FALSE.
- Ak je stav rebríka TRUE, potom RES vynuluje zadanú štruktúru a nastaví sa výstup inštrukcie na TRUE.
- V stave POSTSCAN je výstupom inštrukcie FALSE.



Obr. 6 Ukážka inštrukcie RES v jazyku LD

Záver

Časovače sú základným prostriedkom na realizáciu programového oneskorenia požadovaných operácií. V praxi pri riadení diskretných udalostných systémov je ich používanie pomerne časté. Častokrát je tiež výhodné použiť inštrukcie počítadiel na počítanie udalostí v systéme. Preto je ich znalosť potrebná pre každého programátora

PLC.

Podakovanie

Tento príspevok vznikol vďaka podpore Nadácie Tatra banky - E-Talent z Nadácie Tatra banky č.2013et030.

Literatúra

1. http://literature.rockwellautomation.com/idc/groups/public/documents/webassets/browse_category.hcst
2. Interné prednášky z predmetu Softvér riadiacich systémov na FEI STU Bratislava

Spoluautorom článku je Leo Mrafko, Ústav robotiky a kybernetiky, FEI STU, Ilkovičova 3, 812 19 Bratislava, Slovenská republika
