

18

18. ZPŮSOBY ŘEŠENÍ AUTOMATICKÉ VÝMĚNY NÁSTROJE A TECHNOLOGICKÉ TABULKY

18.1 Způsoby řešení automatické výměny nástroje

Automatická výměna nástroje může být řešena asi dvěma různými přístupy:

- 1) Každý nástroj je vždycky ve svém lůžku. Tento způsob je jednodušší, nevyžaduje vytvářet žádnou tabulku nástrojů. Na začátku se začíná s nástrojem ve vřetení a je nastavena tzv. "paměť starého nástroje". Při výměně nástroje se provedou akce:
 - a) Podle funkce T se nalezne lůžko nového nástroje a nástroj se vybere.
 - b) Podle paměti starého nástroje se nalezne lůžko pro nástroj ve vřetení.
 - c) Proveďte se výměna nástroje.
 - d) Přepíše se paměť starého nástroje číslem lůžka nového nástroje.
 - e) Starý nástroj se zasune do zásobníku

Nevýhody tohoto řešení:

Obsluha musí mít rozmístění nástroje před obráběním v lůžkách podle toho, jak budou volány z programu.

Při každé výměně zásobník nástrojů musí vyhledat nástroj 2x.

- 2) Každý nástroj je v lůžku podle "tabulky nástrojů". PCL program má vytvořenu tabulku nástrojů, která určuje, jaký nástroj je v lůžku. Obsluha si musí před spuštěním programu vyplnit tabulku nástrojů podle toho, jak je program potřebuje a podle toho, kde jsou fyzicky umístěny. PLC program je složitější, protože musí obsluhovat editaci a zobrazování tabulky nástrojů.

Pro vytváření a obsluhu tabulky nástrojů dále načtneme některé vhodné způsoby řešení:

18.2 Zálohovaná paměťová oblast LABEL_MEM

V modulu INCLEXT.ASH (EXT_04.ASH pro procesor CPU04) je deklarováno návěstí LABEL_MEM. Toto návěstí ukazuje na zálohovanou paměťovou oblast o velikosti 1000 bajtů. Data uchovaná v této oblasti se zachovávají po vypnutí napájení systému. Procesor CPU02 má externí zálohování a proto se nesmí vysunout z kazety. Procesor CPU04 má zálohování na kartě a proto se může vysunout z kazety a nedojde ke ztrátě dat. Zálohovanou paměťovou oblast je možno s výhodou použít pro uchovávání tabulky nástrojů.

Od systémové verze panelu 30.09 a kazety 5.014 (20.9.1999) je možnost využít archiv paměťové oblasti do souboru **PLC_MEM.SYS**. Zálohování i obnova PLC paměti se provedou na stisk příslušného softwarového tlačítka v menu pro systémové prostředky.

Od systémové verze panelu 30.23 a kazety 5.034 (15.5.2000) má PLC program možnost sám si vyžádat zálohu PLC paměti nastavením bitu **REQ_SEND_PLCMEM** na hodnotu 1. Po úspěšném odvysílání dat z paměťové oblasti LABEL_MEM do panelu systému je bit REQ_SEND_PLCMEM shozen na hodnotu 0. V PLC programu je proto vhodné naprogramovat vyslání dat z LABEL_MEM například pomocí mechanismu:

```
FL      1,REQ_SEND_PLCMEM;požadavek na vyslání
EX
LDR     REQ_SEND_PLCMEM  ;čekání na konec přenosu
EX1
```

18.3 Orientace v zálohované oblasti LABEL_MEM

Je několik možností, jak zlepšit orientaci v zálohované oblasti LABEL_MEM.

Nejjednodušší způsob je přistupovat k jednotlivým prvkům pomocí symbolicky definovaných offsetů:

Příklad:

```
EQUI   PRVEK_A,6
EQUI   PRVEK_B,8
```

```
LOD     WORD.(LABEL_MEM + PRVEK_A)
STO     WORD.(LABEL_MEM + PRVEK_B)
```

Orientace v zálohované oblasti pomocí struktury:

Příklad:

```
ALFA      struc          ;deklarace struktury ALFA
    PRVEK_A  DB ?          ;PRVEK_A 1 byte
    PRVEK_B  DW ?          ;PRVEK_B 2 byte
ALFA
```

```
LOD     BYTE.LABEL_MEM.PRVEK_A      ;nacte PRVEK_A
STO     WORD.LABEL_MEM.PRVEK_B      ;zapise do PRVEK_B
```

18.4 Využití indexace v adresaci pro paměťovou oblast LABEL_MEM

Způsob indexace je možný v jazyku TECHNOL pomocí registru BX procesoru. Při nastavení registru BX je možno řídit posun v paměťové oblasti nebo v jednotlivých prvcích struktury. Nastavení registru BX se musí provést pomocí instrukcí jazyka "assembler 86 (386)".

Příklad:

Načtení prvku z LABEL_MEM, který je posunutý od počátku oblasti podle wordové buňky POSUN:

```
LOD     POSUN              ;nacte posun v oblasti (0,1,2,..)
MOV     BX,CX              ;instrukce ASM86 - nastavení BX
LOD     BYTE.LABEL_MEM[BX] ;nacte prvek z pameti
```

18.5 PLC obrazovky a PLC menu struktury pro výměnu nástroje

Obsluha tabulky nástrojů v PLC programu vždy vyžaduje využit prostředky vytváření PLC obrazovek a PLC menu struktur.

V PLC menu struktuře je asi vhodné mít možnost volby tabulky nástrojů, možnost volby lůžka nástroje a možnost volby nástroje ve zvoleném lůžku.

V PLC obrazovce pro tabulku nástrojů je vhodné mít očíslovaná lůžka v prefixu obrazovky dynamicky vyplňovat čísla nástrojů v jednotlivých lůžkách. Kromě toho je možné mít zobrazen nástroj ve vřetenu a stav výměny.

Pro editaci tabulky je možno například vytvořit samostatné položky pro zadávání lůžka a pro zadávání nástroje. PLC program pak snímá všechna tlačítka z panelu například pomocí mechanismu popsaném v kapitole "Snímání tlačítek z panelu systému CNC836 do PLC". V tomto případě je ještě vhodné nějakým způsobem na PLC obrazovce zobrazit, že je položka v editaci a editaci ukončit například tlačítkem ENTER.

Příklady:

;Příklad deklarace struktury (syntax ASM86) pro lepší orientaci v zálohované paměťové oblasti ;LABEL_MEM pro zásobník nástrojů:

;(Struktura je deklarována "bajtově" proto, aby se zjednodušilo vysílání dat pomocí sekvencí z paměti ;LABEL_MEM do PLC obrazovky.)

;Deklarace struktury je v oblasti deklarace dat.

cmos_mem	struc	
	VN_FAZE	DB ? ; fáze výměny nástroje
	VR_T	DB 2 DUP (?) ; číslo nástroje ve vřetenu
	VN_RU_T_VR	DB 2 DUP (?) ; číslo nástroje v ruce u vřetena
	VN_RU_T_ZN	DB 2 DUP (?) ; číslo nástroje v ruce u zásobníku
	VN_T_CTRL	DB ? ; status
	ZN_T_TAB	DB (ZN_MAX*2) DUP (?) ; tabulka nástrojů
	ZN_T_SUM_TAB	DB ? ; kontrolní suma tabulky nástrojů
cmos_mem	ends	

;Příklad načtení čísla nástroje ve vřetenu:

;(Využijeme způsob orientace pomocí struktury.)

LOD WORD.LABEL_MEM.VR_T ;načte do DR nástroj ve vřetenu

;Příklad načtení čísla nástroje v lůžku podle buňky EDIT_LUZKA:

;(Využijeme způsob orientace pomocí struktury a indexace.)

LOD	LUZKO	;načte do DR číslo lůžka
CALL	SET_INDEX	;nastaví BX podle lůžka
LOD	WORD.LABEL_MEM.ZN_T_TAB[BX]	;načte do DR číslo nástroje
		;v zadaném lůžku

;Příklad zápisu nového čísla nástroje v NASTROJ do tabulky podle buňky LUZKO:
;(Využijeme způsob orientace pomocí struktury a indexace.)

LOD	LUZKO	;načte do DR číslo lůžka
CALL	SET_INDEX	;nastaví BX podle lůžka
LOD	NASTROJ	;načte do DR číslo nástroje
STO	WORD.LABEL_MEM.ZN_T_TAB[BX]	;zápis čísla nástroje do zadaného lůžka

;PODPROGRAMY - NA KONCI PŘED INSTRUKCÍ STOP

;Nastavení indexu BX podle DR registru.
;V DR registru je BCD hodnota maximálně 0 - 99.

SET_INDEX:

MOV	AL, CL	;převod byte BCD do BIN
AND	CL, 0FH	
SHR	AL, 4	
MOV	CH, 10	
MUL	CH	
ADD	AL, CL	
MOV	BL, AL	
XOR	BH, BH	
DEC	BX	
SHL	BX, 1	
RET		

18.6 Technologické tabulky

Od softwarové verze panelu systému 30.23 a kazety 5.034 (15.5.2000) jsou v CNC systému zpřístupněny technologické tabulky. Technologické tabulky jsou plně k dispozici pro PLC program a využívají se například pro nástrojové hospodářství nebo pro nastavení technologie obrábění v závislosti na materiálu.

V konfiguračním souboru CNC836.KNF je zaveden konfigurační parametr **\$66**, kde je uveden filtr pro technologické tabulky. Pokud není parametr \$66 uveden, je filtr nastaven na implicitní hodnotu **"TAB*.TCH"**. Všechny technologické tabulky jsou umístěny v implicitním adresáři CMOS. Práce s technologickými tabulkami je stejná, jako práce s běžnými tabulkami - například tabulkami korekcí. Přístup k nim je z hlavního menu přes softwarové tlačítko tabulek a tlačítko **"TECHNOLOG"**. Je povolena jejich editace standardním editorem a všechny periferní operace jako u běžných tabulek.

Po ukončení edice se provede syntaktická kontrola technologické tabulky a potom se tabulka automaticky přesune do zálohované paměťové oblasti **"TAB_TECHNOL"** přístupné z PLC programu. Přesun technologické tabulky lze provést také nastavením kurzoru na příslušnou tabulku a stiskem tlačítka ENTER.

PLC program má možnost upravenou tabulku vyslat nazpátek do panelu a uložit do souboru s příslušnou tabulkou pomocí bitu **"REQ_SEND_TABTECH"**.

V systému je také možnost použít okamžitý **náhled do technologické tabulky** bez možností editace na stisk jednoho tlačítka. Konkrétní tlačítko se určí přiřazením makra 0E00h v definičním souboru klávesnice KLAV.KNF (viz příloha L - tlačítka na panelu systému).

18.7 Syntaktická pravidla pro tvorbu technologické tabulky

Technologická tabulka je sestavena ze čtyřbyteových (double-wordových) BCD hodnot s pevným formátem (podobně jako strojní konstanty BUKONxx). Ve čtvrtém byte je bit s váhou 80h rezervován pro znaménko:

high.....	low			
x x	x x	x, x	x x	formát
+3	+2	+1	+0	offset adresy
1 2h	3 4h	5 6h	7 8h	číslo: + 12345.678

Technologická tabulka může obsahovat maximálně 400 těchto čtyřbyteových hodnot sestavených do libovolné matice. Počet řádků a sloupců matice se musí zadat za klíčovým slovem **\$TCH** (viz dále).

1. Úvodní informační popis tabulky

Celý prostor v souboru technologické tabulky od začátku až po výskyt klíčového slova se syntakticky nekontroluje a může se využít pro různé textové popisy pro obsluhu. Například se tam popíše význam jednotlivých sloupců tabulky. Tento popis se zobrazí také při náhledu do technologické tabulky (pomocí makra 0E00h) a pomáhá také při editaci tabulek.

2. Klíčové slovo

Po úvodním informačním popisu následuje klíčové slovo pro technologické tabulky, které musí mít tvar:

\$TCH v ss rr

- **\$TCH** je klíčové slovo pro technologické tabulky
- **v** číslo verze pro syntaktickou kontrolu (do softwarové verze 30.23 je 1)
- **ss** počet sloupců tabulky v rozsahu 1 až 20
- **rr** počet řádků tabulky v rozsahu 1 až 400

Například pro tabulku 5 sloupců a 30 řádků bude napsáno: **\$TCH 1 5 30**

Počet sloupců a řádků musí být uveden tak, aby počet položek tabulky nebyl větší než 400.

3. Oblast dat tabulky

Po uvedení klíčového slova se začne provádět syntaktická kontrola tabulky.

- Každý řádek tabulky musí být uveden pořadovým číslem zakončeným dvojtečkou. Číslování řádků tabulky musí být od nuly. Řádky tabulky nemusí odpovídat textovým řádkům. Jeden řádek tabulky může obsahovat položky napsané ve vícero textových řádcích. Když se má využít zpětný přenos tabulky z PLC do souboru, doporučujeme na jeden řádek umístit maximálně 6 položek.

Příklad řádku tabulky: "052:".

- Každá položka se píše ve standardním formátu běžným pro všechny tabulky včetně znaménka. Položky musí být odděleny mezerou (tabulátorem).

Příklady zápisu položek: +12.23 -45. 12 0.45 .05 1 -100 12345.678

- Je povoleno vkládat komentáře:
 - a) začáteční znak komentáře je ' ; '. Komentář platí do konce textového řádku (použito např. ve strojních konstantách)
 - b) začáteční znak komentáře je ' " '. Komentář platí buď do konce řádku nebo po nejbližší výskyt znaku ' " '. (použito např. v partprogramech)

Syntaktická kontrola uloží položku tabulky v BCD tvaru do paměťové oblasti TAB_TECHNOL podle vzorce (číslo řádku tabulky) * (počet sloupců tabulky) + (pořadové číslo položky v řádku) (viz dále).

Příklad technologické tabulky:

Technologická tabulka 108
tabulka má 3 sloupce a 4 řádky

```

.....VÝŠKA .....POLOHA.....ČAS NÁHŘEVU
$TCH 1 3 4
00: 180.520 -5.1 0.0
01: 252.222 -3.0 "nebo -2" 6.1
02: 0.0 12.5 5.3 "nebezpečné
03: 121.5 -5.6 0.08

```

18.8 Zálohovaná paměťová oblast TAB_TECHNOL

Syntaktická kontrola technologické tabulky ukládá data do zálohované paměťové oblasti **TAB_TECHNOL**, která je přístupná pro PLC program. Organizace oblasti **TAB_TECHNOL** je zabezpečena pomocí datové struktury **TAB_TECHNOLS**, která je přístupná pro PLC program v souboru **EXT_04.ASH**

Struktura deklarovaná v souboru **EXT_04.ASH**

```

TAB_TECHNOLS          STRUC          ;definice struktury
    TCH_VERZE          DW      ?      ;verze
    TCH_SLOUPCE        DW      ?      ;počet sloupců tabulky (bin)
    TCH_RADKY          DW      ?      ;počet řádků tabulky (bin)
    TCH_VOLBA          DW      ?      ;vyvolený řádek tabulky (bin)
    TCH_AKTIV          DB      ?      ;příznak aktivace (0, 1)
    TCH_REZB          DB      ?      ;rezerva
    TCH_REZ            DW 3 DUP (?)    ;rezerva
    TCH_FILENAME       DB 16 DUP (?)  ;název souboru tabulky
    TCH_DATA           DD      ?      ;začátek pro oblast dat (bcd)
TAB_TECHNOLS          ENDS          ;konec definice struktury

```

PLC program má přístup k zálohované oblasti **TAB_TECHNOL**. Tato paměťová oblast začíná hlavičkou, kde je uvedena verze, počet sloupců, počet řádků a název souboru pro aktivní technologickou tabulku. Potom následují data, ve kterých jsou uloženy jednotlivé položky tabulky. Pro každou datovou položku tabulky je vyhrazeno místo 4 bajty.

Pro některé způsoby použití technologických tabulek je vhodné využít možnost volby řádku tabulky. Pomocí příznaku aktivace je zabezpečeno převzetí aktualizovaných dat PLC programem.

adresa	0,1	2,3	4,5	6,7	,89	10,11	12,13	14,15
0	verze	počet sloupců	počet řádků	vyvolený řádek	aktivace			
16(10h)	název souboru (TAB123.TCH)							
32(20h)	pol[0,0]		pol[1,0]				
48(30h)							

Příklady práce PLC programu s technologickou tabulkou:

Příklad načtení počtu sloupců a řádků do DR registru (binární hodnota):

```

LOD    WORD.TAB_TECHNOL.TCH_SLOUPCE ;načte počet sloupců
LOD    WORD.TAB_TECHNOL.TCH_RADKY   ;načte počet řádků

```

Příklad načtení osmé položky (od začátku) z tabulky bez ohledu na strukturu (například jednosloupcová tabulka) (BCD hodnota):

```

    LOD   CNST.(8-1)           ;osmá položka v tabulce
    RL    CNST.2               ;x4
    MOV   BX,CX                ;nastaví index BX
    LOD   DWRD.TAB_TECHNOL[BX].TCH_DATA ;načte osmou položku

```

Příklad pro vybrání položky [I, J], kde I je číslo sloupce v bajtové buňce POL_I a J je číslo řádku v bajtové buňce POL_J. V buňkách jsou binární hodnoty sloupce a řádku počítáno od 0.

```

POL_I:      DS    1           ;číslo sloupce
POL_J:      DS    1           ;číslo řádku
POL_IW:     DS    2           ;počet sloupců pro wordové operace

    LOD     POL_I
    STO     POL_IW             ;přepis sloupce do wordu
    LOD     POL_J             ;načte číslo řádku
    MULB    BYTE.TAB_TECHNOL.TCH_SLOUPCE ;vynásobí řádek počtem
                                         ;sloupců
    AD      POL_IW             ;připočte počet sloupců
    RL      CNST.2             ;x 4 (čtveřice bajtů)
    MOV     BX,CX              ;nastaví index BX
    LOD     DWRD.TAB_TECHNOL[BX].TCH_DATA ;čtení dat z pozice I,J

```

Příklad pro zjištění čísla vyvoleného řádku tabulky:

```

    LOD     WORD.TAB_TECHNOL.TCH_VOLBA ;vyvolený řádek
    EQ      CNST.0FFFFh              ;test zda byl vyvolen
    JL1     NEBYL_VYVOLEN             ;odskok když nebyl
    STO     NOVY_RADEK                 ;zápis a zpracování

```

Příklad pro test aktivace na převzetí aktualizovaných dat tabulky:

```

    LOD     BYTE.TAB_TECHNOL.TCH_AKTIV ;příznak aktivace
    EQ      CNST.0                     ;test
    JL1     NENI_ZMENA                 ;odskok když není změna
    LOD     CNST.0                     ;převzetí
    STO     BYTE.TAB_TECHNOL.TCH_AKTIV ;a zpracování dat ....

```

18.9 Modifikace a zálohování paměťové oblasti TAB_TECHNOL

PLC program může jednotlivé položky v paměťové oblasti TAB_TECHNOL měnit. Tato paměťová oblast je zálohovaná, proto se hodnoty zachovávají i po vypnutí a zapnutí systému. Modifikací dat v paměťové oblasti TAB_TECHNOL samotným PLC programem vznikne nesoulad se souborem zvolené technologické tabulky. V závislosti na způsobu použití technologické tabulky vzniklý nesoulad dat může, nebo nemusí být přípustný.

V případě, že PLC program požaduje modifikovat data i ve vzorovém souboru technologické tabulky, může požádat o zpětný přenos dat z paměťové oblasti TAB_TECHNOL do příslušného souboru s technologickou tabulkou (TABxx.TCH) nastavením bitu "REQ_SEND_TABTECH" na hodnotu 1. Po úspěšném odvysílání dat do panelu systému je bit REQ_SEND_TABTECH shoden na hodnotu 0. V PLC programu je proto vhodné

naprogramovat vyslání dat z TAB_TECHNOL například pomocí mechanismu:

```

FL      1,REQ_SEND_TABTECH      ;požadavek na vyslání
EX
LDR     REQ_SEND_TABTECH ;čekání na konec přenosu
EX1

```

Vyslání dat z oblasti TAB_TECHNOL by se nemělo provádět počas pohybu souřadnic. Je vhodné například spustit požadavek na vyslání v přípravných funkcích a čekat na ukončení přenosu. Tím je zabezpečeno, že případný pohyb bude vykonán až po ukončení přenosu.

Přijátá data se v panelu systému uloží do souboru, kterého název je v hlavičce paměťové oblasti TAB_TECHNOL (TCH_FILENAME), který musí existovat. Zachová se úvodní informační popis tabulky před klíčovým slovem. Data budou zaznamenána ve sloupcích, maximálně 6 položek na textový řádek. Systém provede potlačení vedoucích nul. Organizace tabulky bude převzata z hlavičky paměťové oblasti TAB_TECHNOL.

18.10 Přenos dat z PLC programu do NC tabulek

Od softwarové verze panelu 30.24 a kazety 5.035 (23.5.2000) je umožněno nastavování dat v tabulkách korekcí, posunutí počátku a tabulkách parametrů. Systém zabezpečí přenos dat z kazety do panelu a nastavení příslušné položky v aktuální tabulce, která je v paměti systému. Odpovídající hodnoty položek v souborech s tabulkami zůstanou nezměněny. Přenos dat z PLC programu do tabulek NC systému nezmění žádné soubory v paměti CMOS.

V NC programu je tak umožněno vytvořit zpětnou technologickou vazbu na PLC program. V NC programu je výhodné použít instrukce z aritmetiky parametrů pro přesuny dat mezi tabulkami korekcí, posunutí, parametrů, strojních konstant a registrů bloku.

V PLC programu je zpřístupněna paměťová oblast **SET_NCTAB**, která slouží na upřesnění požadavku na nastavení příslušné hodnoty v tabulce. Tato paměťová oblast je organizovaná pomocí datové struktury:

```

SET_NCTABS  STRUC                ;začátek definice struktury
NCTAB_VERZE      DB      ?      ;verze (1)
NCTAB_MOD        DB      ?      ;mód přenosu (0=asynch. 1=synch.)
NCTAB_TAB        DB      ?      ;kód tabulky (1=KOR, 2=POS, 6=PAR)
NCTAB_SLOUPEC    DB      ?      ;sloupec v tabulce (1, 2, 3, ....)
NCTAB_RADEK      DB      ?      ;řádek (KOR:1-99 POS:53-59 PAR:0-95)
NCTAB_DATA       DD      ?      ;BCD hodnota (DWRD)
SET_NCTABS  ENDS

```

Přenos požadavku do systému se zprostředkuje nastavením bitu **REQ_SEND_NCTAB** na hodnotu **1**. Po ukončení přenosu se bit REQ_SEND_NCTAB sám vynuluje. PLC program by měl tento bit nastavit jednorázově na hodnotu 1 a pak testovat ukončení přenosu. Proto je vhodné aktivaci přenosu umístit do sekvenčního celku.

Příklad aktivace přenosu:

```

FL      1,REQ_SEND_NCTAB      ;nastavení požadavku
EX
LDR     REQ_SEND_NCTAB        ;čekání na konec přenosu
EX1

```

Význam jednotlivých položek struktury pro oblast **SET_NCTAB** :

adresa	název	popis	hodnota
0	NCTAB_VERZE	verze plnění	1
1	NCTAB_MOD	nastavení způsobu synchronizace plnění tabulky	0= okamžité nastavení hodnoty 1= nasatvení hodnoty v tabulce se provede na začátku nejbližší přípravy bloku
2	NCTAB_TAB	kód tabulky (natažená v aktuální paměti systému, ne v souborech TABxx.KOR..)	1= tabulka korekcí nástroje 2= tabulka posunutí počátku 6= tabulka parametrů
3	NCTAB_SLOUPEC	sloupec v atbulce korekcí	1, 2, 3, 4, 5
		sloupec v tabulce posunutí	1, 2, 3, 4, 5, 6
		sloupec v tabulce parametrů	1
4	NCTAB_RADEK	řádky v tabulce korekcí (bin)	1, 2, 3, , 99
		řádky v tabulce posunutí	53, 54, , 59
		řádky v tabulce parametrů	0, 1, 2, , 95
5	NCTAB_DATA	čtyřbajtová BCD hodnota dat	(12345.678)

Příklad plnění jednotlivých položek struktury vysvětlíme na příkladu PLC programu. Budeme přenášet hodnotu 12.345 do parametru R62. Vzorový kód programu musí být umístěn v mechanismu nebo v přípravných funkcích:

```

LOD   CNST.12345h
STO   DWRD.pozadovana_hodnota      ;požadovaná hodnota parametru

LOD   CNST.1
STO   BYTE.SET_NCTAB.NCTAB_VERZE  ;nastavení verze plnění

LOD   CNST.0
STO   BYTE.SET_NCTAB.NCTAB_MOD     ;okamžité nastavení hodnoty
                                         ;v tabulce

LOD   CNST.6
STO   BYTE.SET_NCTAB.NCTAB_TAB     ;požadujeme přenos do tabulky
                                         ;parametrů

LOD   CNST.1
STO   BYTE.SET_NCTAB.NCTAB_SLOUPEC ;tabulka parametrů má jen jeden
                                         ;sloupec

LOD   CNST.62
STO   BYTE.SET_NCTAB.NCTAB_RADEK   ;plníme parametr R62

LOD   DWRD.pozadovana_data
STO   DWRD.SET_NCTAB.NCTAB_DATA    ;nastavení požadovaných dat

FL    1, REQ_SEND_NCTAB            ;požadavek na přenos
EX

LDR   REQ_SEND_NCTAB                ;čekání na konec přenosu
EX1

```

18.11 Vypínání a zapínání systému řady CNC8x9-DUAL

Systémy řady CNC8x9 – DUAL, na rozdíl od standardní verze, nemají paměť CMOS pro zálohování dat z technologických tabulek, paměťové oblasti LABEL_MEM apod. Místo paměti CMOS mají ve všech variantách HARDISK. Používání HARDISKu pro PLC data vyžaduje vždy před vypnutím systému nejdříve zálohovat potřebná data na HARDISK a při zapnutí systému je obnovit. V dalším článku je doporučený postup při vypínání a při zapínání systému.

O potřebě zálohovaných dat rozhodují tyto strojní konstanty:

- Osmá dekáda strojní konstanty R97 (**8.R97**) určuje požadavek zálohování odměřování při vypnutí a obnovu odměřování při zapnutí systému.
- Čtvrtá dekáda strojní konstanty R329 (**4.R329**) určuje požadavek zálohování technologických tabulek a paměťové oblasti LABEL_MEM při vypnutí a jejich obnovu při zapnutí systému.

VYPÍNÁNÍ SYSTÉMU:

- SIGNÁL PRO VYPNUTÍ.** PLC program dostane signál pro vypnutí systému. Signálem může být libovolný vstup ze stroje, ale doporučuje se používat na systému připravené softwarové tlačítko **vypnutí systému**. Softwarové tlačítko je v menu systém (oko) (bývalé softw.tlačítko HELP pro kontextovou nápovědu). Po stisku softwarového tlačítka pro vypnutí se objeví dotaz na potvrzení akce a potom se vyšle do PLC programu v buňce **MATTl** kód **0E7h**. Systém na stisk tlačítka neprovede žádnou akci. Pokud softwarové tlačítko nevyhovuje, může se pomocí definičního souboru klávesnice KLAV.KNF umístit kód 0E7h na libovolné volné tlačítko panelu.
- ZÁLOHA PLC DAT.** Když je požadavek na zálohování PLC dat, je vhodné vytvořit v PLC programu zálohovací mechanismus a v něm uvést:

Vypnutí všech pohonů, hydrauliky apod.

Když je požadavek na zálohování odměřování, musí být nastavena osmá dekáda strojní konstanty R89 na hodnotu 1. Požadavek na zálohování odměřování se naprogramuje zadáním log.1 do bitu **REQ_BACKUP_POS** a systém uloží zálohu odměřování do souborů \$BKP_POS.SYS a \$BKP_POS.BAK v adresáři SYSFILES. Po zapsání na disk systém bit REQ_BACKUP_POS vynuluje.

```
FL      1, REQ_BACKUP_POS      ;záloha odměřování 8.R97=1
EX
LDR     REQ_BACKUP_POS        ;čekáme na zápis na disk
EX1
```

Když je požadavek na zálohování technologických tabulek a paměťové oblasti LABEL_MEM, musí být nastavena čtvrtá dekáda strojní konstanty R329 na hodnotu 1. Požadavek na zálohování paměti se naprogramuje zadáním log.1 do bitu **REQ_BACKUP_MEM** a systém uloží zálohu odměřování do souborů \$BKP_MEM.SYS, \$BKP_TCH.SYS a \$BKP_MEM.BAK, \$BKP_TCH.BAK v adresáři SYSFILES. Po zapsání na disk systém bit REQ_BACKUP_MEM vynuluje.

```
FL      1, REQ_BACKUP_MEM      ;záloha paměti 4.R329=1
EX
LDR     REQ_BACKUP_MEM        ;čekáme na zápis na disk
EX1
```

- c) **ÚKLID SYSTÉMU.** PLC program dá požadavek na úklid systému (zavření všech souborů) pomocí bitu **REQ_CLOSE_ALL**.

```

FL      1, REQ_CLOSE_ALL      ;úklid systému
EX
LDR     REQ_CLOSE_ALL        ;čekáme na ukončení
EX1

```

- d) **ZÁVĚREČNÉ AKCE.** PLC program může pomocí zprávy (message BZH21) informovat obsluhu o možnosti vypnout stroj. V tomto případě se jen čeká na vypnutí stroje. Když je potřeba vypnout systém z PLC programu, může se napsat požadavek na vykonání modulu PIS_HALT (MODULE_HALT) pomocí bitu **REQ_PIS_HALT**. Ve výkonu modulu PIS_HALT se může celý systém vypnout, například rozepnutím rozpínacího kontaktu relé, přes který je systém napájen.

```

FL      1, REQ_PIS_HALT      ;požadavek na modul PIS_HALT
EX
LDR     REQ_PIS_HALT        ;nekonečný cyklus, protože se systém nakonec zastaví
EX1

```

ZAPÍNÁNÍ SYSTÉMU:

Při zapínání systému PLC program nemusí provádět žádné akce týkající se obnovy dat.

Když je požadavek na zálohování odměřování a je nastavena osmá dekáda strojní konstanty R89 na hodnotu 1, systém obnoví odměřování ze souboru \$BKP_POS.SYS a po obnově (zapnutí) soubor vymaže. V případě, že záloha systému nebyla provedena a systém při zapínání nenalezne soubor \$BKP_POS.SYS, zahlásí se na obrazovce chybové hlášení a odměřování se neobnoví.

Když je požadavek na zálohování technologických tabulek a paměťové oblasti LABEL_MEM a je nastavena čtvrtá dekáda strojní konstanty R329 na hodnotu 1, systém obnoví paměťové oblasti ze souborů \$BKP_MEM.SYS a \$BKP_TCH.SYS a po obnově (zapnutí) soubory vymaže. V případě, že záloha systému nebyla provedena a systém při zapínání nenalezne soubory \$BKP_MEM.SYS a \$BKP_TCH.SYS, zahlásí se na obrazovce chybové hlášení a systém se pokusí obnovit paměťové oblasti ze záložních souborů \$BKP_MEM.BAK a \$BKP_TCH.BAK, ve kterých ale nemusí být čerstvá data. Soubory \$BKP_MEM.BAK a \$BKP_TCH.BAK systém nemaže ale jen přepisuje při požadavku REQ_BACKUP_MEM. z PLC programu.

18.12 UPS se sériovým kanálem

Od verze software sekundárního procesoru 6.325 je možnost zálohování napájení pomocí UPS se sériovým kanálem RS232C. UPS musí používat protokol kompatibilní s protokolem firmy POWERCOM.

Na systému je UPS připojen na sériový kanál COM2. Tento kanál se také používá pro ladění PLC Wintechnolem. Pro přepínání sériového kanálu mezi Wintechnol a UPS slouží aktivace UPS v 1.dekádě strojní konstanty **R438**. Když se nastaví aktivační bit ve strojní konstantě R438, bude sériový kanál COM2 přidělen pro spojení s UPS.

Strojní konstanta R438

R438 (UPS se sériovým kanálem)

1.dekáda	0	UPS není použita	Aktivace UPS
	1		
2.dekáda	0	-	Typ UPS
	1	UPS typu POWERCOM	
5. a 6. dekáda	xx	Sekundy (1-59)	Přídavný čas pro vypnutí UPS od příkazu z PLC (nesmí být 0)
7. a 8. dekáda	xx	Minuty	

Příklad:

Když požadujeme, aby UPS byla vypnuta (a také systém) po 10 vteřinách od povelu z PLC na vypnutí, konstanta R438 bude nastavena:

R438: 00100.011

Signály pro PLC:

V PLC programu jsou k dispozici tyto bitové signály

Byte UPS_STAT - stavové informace z UPS		
Bit 0	UPS_LINEFAIL	1 = Přerušení přívodu napětí na vstupu UPS
Bit 1	(UPS_BATLOW)	1 = Nízké napětí baterie
Bit 2		
Bit 3	(UPS_BOOST)	1 = Regulace napětí (zvětšení, nebo pokles)
Bit 4		
Bit 5	(UPS_OVER)	1 = Přetížení odběru
Bit 6		
Bit 7	(UPS_OFF)	1 = Vypnuto

Byte UPS_CMD - příkazy pro UPS		
Bit 0	UPS_SHUTDOWN	1 = Příkaz pro vypnutí UPS. K vypnutí dojde po přídavném času v R438.

Byte UPS_FAULT - chyby z UPS		
Bit 0	UPS_ERRCOM	1 = Chyba sériového kanálu
Bit 1	UPS_ERRINT	1 = Chyba interruptu

Příklad použití UPS v PLC programu:

;Hlidani napajeni UPS (prubezna vetev)

```

      LDR      UPS_LINEFAIL      ;Přerušeni přívodu napětí na vstupu UPS
      FL1      1,MCH_VYPNUTI    ;Start mechanismu pro vypnutí

```

;Mechanismus vypnutí stroje

MECH_BEGIN MCH_VYPNUTI

```

;Pripadne zalohovani odmerovani
      FL      1,REQ_BACKUP_POS
      EX
      LDR      REQ_BACKUP_POS
      EX1

```

```

;Pripadne zalohovani technol.tabulky a pametove oblasti LABEL_MEM
      FL      1,REQ_BACKUP_MEM
      EX
      LDR      REQ_BACKUP_MEM
      EX1

```

```

;Uklid systemu
      FL      1,REQ_CLOSE_ALL
      EX
      LDR      REQ_CLOSE_ALL
      EX1

```

```

;Vypnutí UPS
      FL      1,UPS_SHUTDOWN    ;Prikaz pro vypnutí UPS

```

;Vypnutí systému

MECH_END MCH_VYPNUTI

