

Převod dat z DATALOGu do MS ACCESS

V dnešním vydání XM budeme převádět data z DATALOGu do MS ACCESS (PeC) prostřednictvím textového souboru s oddělovači (TAB). Tento textový soubor by se dal využít snad i jinde, ale mám odzkoušeno, že EXCEL a ACCESS tyto data vezmou bez řečí.

Nejprve si osvětlíme strukturu dat v DATALOGu, ale jen co se převodu samotných dat týče.

Soubor DATALOGu obsahuje pět částí, každá část je ukončena znaky 0 3 0 (EOF) kromě poslední, ta je ukončena 0 0 0 (konec souboru). Každá část se skládá z vět a věta má takovouto strukturu: **LEN, ID, DATA**

LEN = délka věty včetně tohoto bajtu, pokud je položka delší než 255 pak je 0,low,high (3 bajt).

ID = identifikátor, v datech (poslední 5. část) určuje pořadí položky v záznamu, navíc jeho horní 2 bity určují zda se jedná o první, vybranou nebo aktivní položku záznamu:

00xxxxxx = vnitřní položka záznamu
01xxxxxx = první položka nevybraného záznamu
11xxxxxx = první položka vybraného záznamu
10xxxxxx = první položka aktivního záznamu

DATA= jsou vlastní data věty a jejich význam záleží na identifikátoru.

1. část: IDENTIFIKACE SOUBORU - pro naše účely nepodstatné
2. část: SEZNAM POLOŽEK - Obsahuje čísla a názvy položek. Budeme potřebovat zjistit, kolik je použito položek
3. část: SEZNAM ZPRÁV - pro naše účely nepodstatné
4. část: DEFINICE ZPRÁV - pro naše účely nepodstatné
5. část: DATA - pro naše účely velmi podstatné.

Náš úkol je tedy jasný. Projít 1. až 4. část, přitom spočítat položky a vlastní data překonvertovat do češtiny pro NOTEPAD, seřadit podle číslování (ID), prázdné položky vytvořit a oddělit TABulátorem a to celé nějak dostat do souboru. Já jsem volil napojení souboru přes 3. kanál (LPRINT). Jak na to? Sledujte zdroják s komentářem:

```
START:      org 24500                ;tady ta hrůza začíná
            ld  a,3                  ;otevři 3.kanál
            call #1601

            ld  hl,ZAZNAM            ;adresa kam se ukládá
            ld  (ADRZAP),hl         ;nastav do proměnné
            ld  hl,DATALOG          ;zde je nahrán "DATALOG"
            ld  de,TABULKA         ;tabulka s ID a adresami
            ld  (ADRTAB),de

            di

LP2:        call CTIVETU            ;načte jednu větu
            ;ID vyskočí na adr. ID a délka na DELKAVET
            jr  nz,LP2             ;přeskakuj do konce části
```



```

ld    (de),a
inc  de
ld    (de),a
ld    a,(hl)
ld    (DELKAID),a
ld    e,a
ld    d,0
or    a
call  z,DELKA255
ld    (DELKAVET),de
push hl
or    a
ld    hl,3
sbc  hl,de
add  hl,de
pop  hl
jr   nz,NONZERO
or    a
jr   z,KONEC
NONZERO:
or    d
or    e
jr   z,KONEC
inc  hl
ld    a,(hl)
and  %00111111
ld    (ID),a
ld    de,(ADRZAP)
ld    (de),a
inc  de
ld    bc,(ADRZAP)
ld    a,c
ld    (de),a
inc  de
ld    a,b
ld    (de),a
inc  de
ld    (ADRZAP),de
inc  hl
ld    de,(ADRZAP)
ld    bc,(DELKAVET)
ld    a,(DELKAID)
or    a
jr   nz,JP1
dec  bc
dec  bc
JP1:
dec  bc
dec  bc
LP1:
ld    a,(hl)
call  CESTINA
ld    (de),a
inc  hl
inc  de
dec  bc
ld    a,b
or    c
jr   nz,LP1
ld    a,1
or    a
ld    (ADRZAP),de
call  KONECZN
ret

```

```

KONECZN:
    ld    de, (ADRTAB)      ;vyvoření koncové značky
    ld    a,255             ;pro případ, že jde o posle-
    ld    (de),a           ;dní položku záznamu
    ld    (de),a           ;ID=255
    inc   de                ;zapiš a posuň
    ld    bc, (ADRZAP)     ;zapiš i adresu prázdného
    ld    a,c              ;místa
    ld    (de),a
    inc   de
    ld    a,b
    ld    (de),a
    ret

KONEC:
    call  KONECZN          ;zapiš koncovou značku
    inc   hl               ;posuň ukazatel dat
    xor   a                ;nastav zero
    ret

DELKAID:      defb 0      ;pár proměnných
DELKAVET:     defw 0
ID:           defb 0
ADRZAP:       defw 0
ADRTAB:       defw 0
POCETPOL:    defb 4

DELKA255:
    inc   hl               ;korekce délky věty kdy je delší
    ld    e,(hl)          ;než 255
    inc   hl
    ld    d,(hl)
    ret

CESTINA:
    cp    128              ;konverze češtiny z DATALOGu
    ret   c                ;do NOTEPADu
    push  de               ;schovej zapis
    ld    e,a              ;písmeno do e
    push  hl               ;schovej data
    ld    hl,DAT2NOT      ;adresa převodní tabulky

NOTFOUND:
    ld    a,(hl)          ;načti písmeno tabulky
    inc   hl               ;posuň se
    ld    d,(hl)          ;načti písmeno nové
    inc   hl               ;posuň se
    cp    0                ;je to písmeno, které neznám
    jr    z,FOUND         ;ano, skoč
    cp    e                ;je to písmeno
    jr    nz,NOTFOUND     ;ne,hledej

FOUND:
    ld    a,d              ;nahraď písmeno novým
    pop   hl               ;obnov ukazatele
    pop   de
    ret

SORT:
    xor   a                ;nyní pošleme data do kanálu
    ld    b,a             ;vynulujeme počítadlo
    ld    b,a             ;počítadlo bude v b

SORTLP:
    ld    hl,TABULKA     ;začneme na začátku

SORTLP1:
    ld    a,(hl)          ;čti ID (číslo položky)
    cp    b                ;je v pořadí

```

```

jr    z, POSLI          ;pošli na kanál
cp    255               ;je to konec tabulky
jr    nz, SORTLP2      ;skoč na další
ld    a, 9              ;tento záznam chybí
rst   16                ;vytvoř prázdný
ld    a, 32
rst   16

POSLEDNI:
ld    a, (POCETPOL)    ;test poslední položky
cp    b                 ;porovnej
jr    nz, SORTLP3      ;ne, pokračuj
ld    a, 13            ;ano, nový řádek
rst   16
ld    a, 10
rst   16
ret                               ;a návrat

SORTLP3:
inc   b                 ;zvyš číslo položky
jr    SORTLP            ;pokračuj

SORTLP2:
inc   hl                ;nastavení další položky
inc   hl
inc   hl
jr    SORTLP1

POSLI:
push  hl                ;pošli do tisku (3.kanál)
push  bc                ;schovej ukazatele
inc   hl                ;ukaz na adresu záznamu
ld    e, (hl)           ;načti adresu do de
inc   hl
ld    d, (hl)
inc   hl                ;skoč na ID dalšího záznamu
inc   hl                ;skoč na adr. záznamu
ld    c, (hl)           ;načti do bc
inc   hl
ld    b, (hl)
ld    h, b              ;přesuň do hl
ld    l, c
or    a
sbc   hl, de            ;odečtením zjistí délku
ld    a, 9              ;pošli oddělovač
rst   16

POSILEJ:
ld    a, (de)           ;pošli celý záznam
rst   16
dec   hl
inc   de
ld    a, h              ;v délce hl
or    l
jr    nz, POSILEJ
pop   bc                ;obnov ukazatele
pop   hl
jr    POSLEDNI         ;testuj poslední záznam

DAT2NOT:
defb  193,225           ;převodní tabulka češtiny
defb  195,232           ;první písmeno DATALOG
defb  196,239           ;druhé písmeno NOTEPAD
defb  215,233
defb  197,236
defb  201,237

```

```

defb 203,190
defb 204,190
defb 206,242
defb 207,210
defb 210,248
defb 198,224
defb 211,154
defb 212,157
defb 213,250
defb 202,249
defb 217,253
defb 218,158
defb 225,193           ;velka pismena
defb 227,200
defb 228,207
defb 247,201
defb 229,204
defb 233,205
defb 235,188
defb 236,197
defb 238,210
defb 239,211
defb 242,216
defb 230,192
defb 243,138
defb 244,141
defb 245,218
defb 234,217
defb 249,221
defb 250,142
defb 216,228           ;přehlasované
defb 214,235
defb 205,246
defb 200,252
defb 241,196
defb 246,203
defb 237,214
defb 232,220
defb 0,32

```

```

TABULKA:      defs 3*64           ;ID+adresa (max 64)
ZAZNAM:       defs 8000          ;8kB pro záznam
DATALOG:      equ 31000         ;tady nahraj DATA

```

Možná vylepšení? Tak těch je teda dost. Např. rutina posílá vždy na začátku řádky oddělovač, tím pak při natažení do Excelu nebo do Accessu prázdný sloupec. Prázdný sloupec vzniká i na konci. Rutina špatně počítá počet položek záznamu: místo xor a dejte ld a,255 nebo při konci části odečtete jednotku. Dále by se dalo nějak zařídit, aby se bralo v úvahu i to, zda je záznam ve výběru nebo ne. Stačilo by třeba poslední sloupec dát "ano" (je vybrán) a "ne" (není vybrán). Identifikace vybrání je v ID věty (popsáno na začátku).

Ještě vám tady napíšu krátký basic, aby se to celé dalo nějak upotřebit:

```

10 CLEAR 24499: LOAD "*"tatorutina"CODE 24500
20 LOAD "*"soubor_z_datalogu"
30 OPEN #3,, "b:konvert.b":RANDOMIZE USR 24500
40 CLOSE #3

```

Pokud máte více souboru z jedné databáze, nemusíte zavírat kanál, stačí nahrát a spustit znova. Kanál zavřete, až je konverze celá (nebo plná disketa).

V roce 2000 konvertoval **SWEET FACTORY**