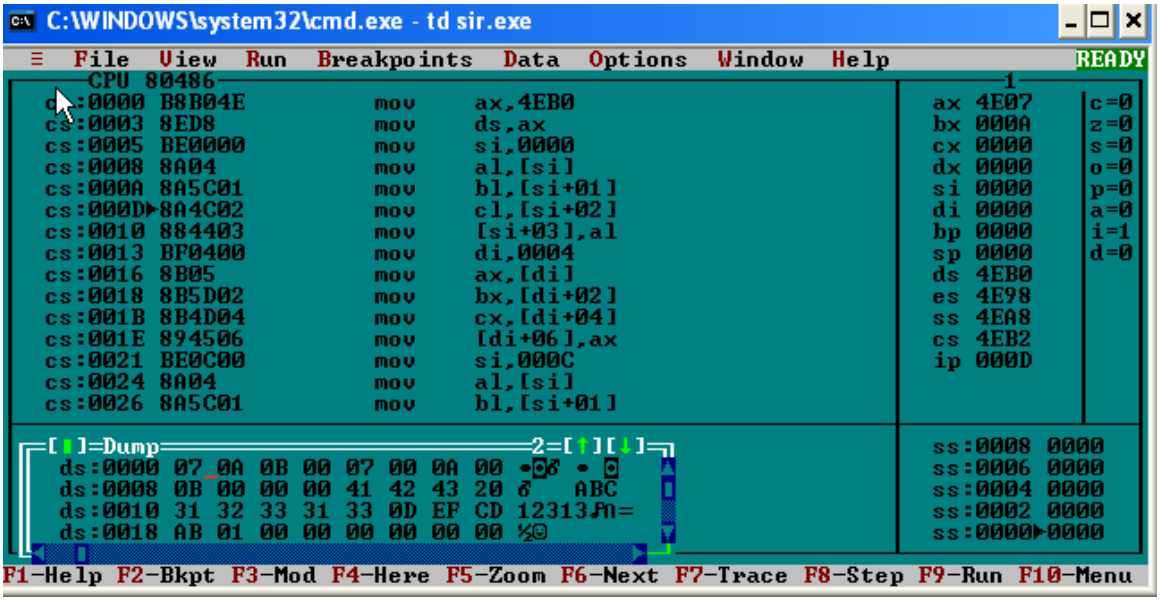
EXEMPLU DE UTILIZARE ŞI EXPLOATARE A MEMORIEI INTERNE

Să se determine harta memoriei pentru datele de mai jos şi offset-urile (deplasările) şirurilor

```
segm_stiva  SEGMENT
        stiva     DB 128 DUP (?)
segm_stiva  ENDS

segm_date  SEGMENT
        sir_date   DB      7,10,11, ?
        sir_date2  DW       7,10,11, ?
        sir_car    DB      'ABC 123'
        car        DB      '13'
        nr         DB       13
        nr_mare    DD       1abcdefh
segm_date  ENDS

segm_cod   SEGMENT
        ASSUME CS:segm_cod, DS:segm_date, SS:segm_stiva

        et:     MOV AX, segm_date
                MOV DS, AX

                MOV  SI, offset sir_date        ;  SI = 00 00
                MOV  AL, [SI]                   ; AX= ....07
                MOV  BL, [SI+1]                 ; BX= ....0A
                MOV  CL, [SI+2]                 ; CX= ....0B
                MOV  [SI+3], AL                 ; valoarea 07 se mută la adresa 0003

                MOV  DI, OFFSET sir_date2       ;  DI = 00 04
                MOV  AX, [DI]                   ; AX= 00 07
                MOV  BX, [DI+2]                 ; BX= 00 0A
                MOV  CX, [DI+4]                 ; CX= 00 0B
                MOV  [DI+6], AX                 ; valoarea 00 07 se mută la adresa
000A

                MOV  SI, OFFSET sir_car         ;  SI = 00 0C
                MOV  AL, [SI]                   ; AX= ....41
                MOV  BL, [SI+1]                 ; BX= ....42
                MOV  CL, [SI+2]                 ; CX= ....43

                MOV  DI, OFFSET car             ;  DI = 00 13
                MOV  AL,car                     ; AX= .... 31
                MOV  AH,nr                      ; AX= 0D ...

                MOV SI, OFFSET nr_mare                  ; SI = 00 16
                MOV AX, WORD PTR [nr_mare]              ; AX= CD EF
                MOV BX, WORD PTR [nr_mare+2]            ; BX= 01  AB
```

```
        MOV AX, 4C00H
        INT 21H

segm_cod   ENDS

END et
```

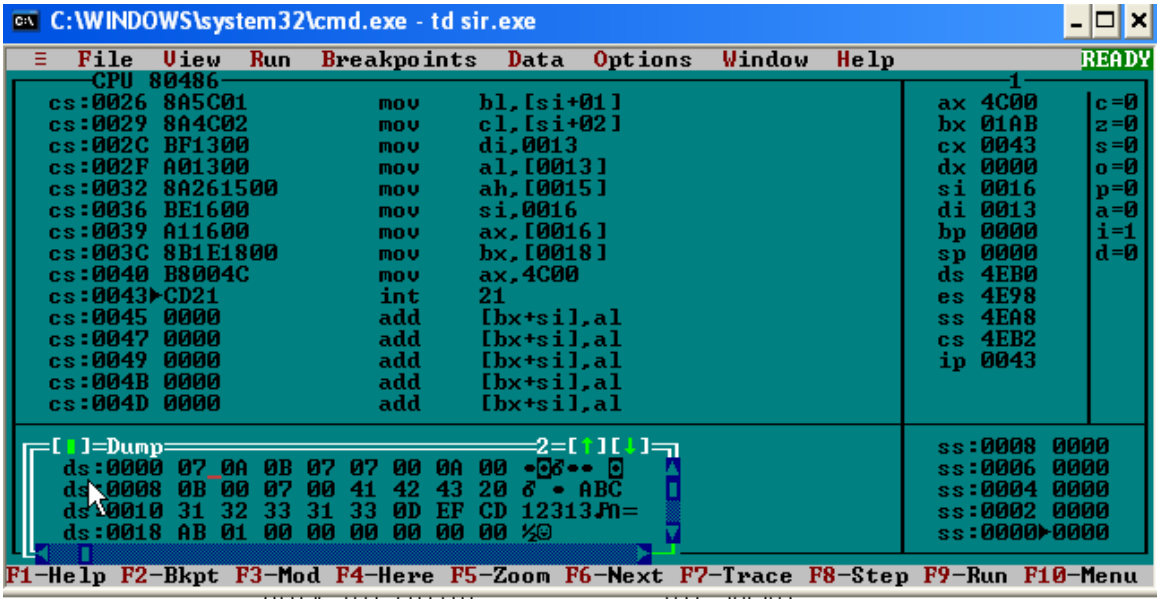


Harta memoriei la începutul programului:

sir_date	*sir_date2*	*sir_car*	*car*	*nr*
07 0A 0B 00	07 00 0A 00 0B 00 00 00	41 42 43 20 31 32 33	31 33	0D

EF CD AB 01
---
nr_mare

OBS.: În memorie, numerele la nivel de B (byte) şi caracterele (şirurile de caractere) sunt reprezentate în ordinea normală, iar numere la nivel de W (word) şi DD (double word) sunt reprezentate invers (se inversează byte-s între ei) – reprezentarea „little endian”



Harta memoriei la sfârşitul programului:

sir_date	*sir_date2*	*sir_car*	*car*	*nr*
07 0A 0B 07	07 00 0A 00 0B 00	07 00 41 42 43 20 31 32 33	31 33	0D

EF CD AB 01
nr_mare