

četrti številka

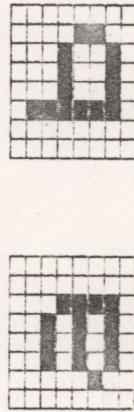
0 1 2 3 4 5 6 7 8 9 A B C D E F 0 1 2 3 4 5 6 7 8 9 A B C D E F

024508AUE	13579BDF	40 - prvi dve številki (58) naslova na zaslonu ter naslova prilastkov
024508AUE	13579BDF	- tretja številka
024508AUE	13579BDF	4B (59)
024508AUE	13579BDF	50 (5A)

SLIKA 1. ZASLONSKA DATOTEKA

Iz prej povedanega je že lažje razložiti ta dva programa in njun rezultat. S tem, zakaj smo v prvem uporabili FF in v drugem C7, pa se bomo ukvarjali malo kasneje.

Zelo zanimiv del ROM-a se skriva na njegovem koncu, med naslovoma 3D00 in 3FFF. V teh 3/4 kb niso programi, temveč dolga tabela. V njej so spravljene točkovni vzorci vseh znakov, razen grafičnih. Za vsak znak je potrebnih 8 zlogov; "a" in "b" npr. sta shranjena takole: 00 00 38 04 3C 00 in 00 20 3C 22 22 3C 00. Zapišite ta števila eno nad drugim v dvojiški obliki in nato pogledajte sliko 2! Zdaj vidite, da je vzorec črk enak razporedju enojk med ničlami v dvojiškem zapisu. Znaki imajo na zaslonu prav takšno obliko. Naslov znaka (se pravi: njegovega vzorca), ki ima kod X ("a" ima kod 61H), izračunamo takole: 8*X + 3C00 (naslov vzorca "a" je torej 8*61 + 3C00 = 3F08).



SLIKA 2. TOČKOVNA VZORCA CRK a IN b.

Spet je pred nami program. Poskusimo najprej ugotoviti, kaj naredi in ga šele potem praktično preizkusimo:

```
21A440 LD HL, 40A4 ;koordinata položaja 5,4
110B3F LD DE, 3F0B ;naslov vzorca znaka A
0608 LD B, 0B
1A Zanka LD A, (DE)
77 LD (HL), A
24 INC H ;pomembno:INC H, ne INC HL
13 INC DE
10FA DJNZ Zanka
C9 RET
```

Ste zadeli? Če je bilo vse v redu, se je na položaju 5,4 zapisala črka a.

Poglejmo še enkrat peto vrstico programa. Rekli smo INC H. Ta kratak in hiter ukaz je vzrok za na videz nesmiselno zapleteno predstavitve zaslona v pomnilniku. Če je naslov "PRINT" položaja HL, potem ima B zlogov, kamor zapišemo znak, naslovo HL, HL+0100, HL+0200, HL+0300 itd. do HL+0700. Z ukazom INC H dosežemo isto kot če bi imeli na voljo ukaz ADD HL, 0100. Zdaj pa si predstavljajte, kaj bi se zgodilo, če bi bil zaslon v pomnilniku predstavljen na naslovih, ki bi tekli lepo eden za drugim od levega zgornjega do desnega spodnjega kota. V tem primeru bi morale biti slike enega znaka na naslovih HL, HL+0020, HL+0040, HL+0060 itd. do HL+00E0. Ker ukaz ADD HL, nn ne obstaja, bi bil edini način za pomik na naslednji položaj PUSH DE ; LD DE, 0020 ; ADD HL, DE ; POP DE. To je precej daljše (in seveda veliko počasneje) kot INC H. Tako se razporeditev, ki jo je marsikdo imel za nespametno, izkaže kot zelo domiselna.

Kljub temu še ostane nekaj težav. Zaslon je razdeljen v tri tretjine in problem se pojavi, ko hočemo preiti iz ene tretjine v drugo. Primer: če je naslov nekega položaja HL, je je naslov naslednjega položaja običajno HL + 0001. Če pa ima HL slučajno vrednost 40FF, potem je naslednji položaj 4800 - ne 4100 (kar je druga vrstica prvega "PRINT" položaja na ekranu). Da bi se vedno pomaknili na naslednji položaj, moramo uporabiti tale program: