

Teh 8 skupin lahko predstavimo kot bite registra A - skupini 0 pripada bit 0, skupini 1 bit 1 itn. Bitu skupine, ki jo želimo čitati, damo vrednost 0, vsem ostalim bitom pa vrednost 1. Če želimo brati skupino (1 2 3 4 5), se pravi tretjo skupino, mora imeti bit 3 registra A vrednost 0. Z drugimi besedami: pred branjem mora biti vrednost v registru A F7H

(A) = 1 1 1 0 1 1 = F7H (= 247).

Po izvršenem ukazu dobimo rezultat v registru A. Biti 5, 6 in 7 niso uporabljene, vrednost bitov od 0 do 4 pa je odvisna od tega, katera tipka je pritisnjena. Najnižji bit ustreza najbolj zunanji tipki; če je tipka pritisnjena, ima bit vrednost 0, sicer 1. V našem primeru vpliva

tipka "1" na bit 0,

tipka "2" na bit 1,

tipka "3" na bit 2 in tako naprej.

Če bi se odločili za branje skupine 4, bi morali dati registru A vrednost EFH (= 239), tipke pa bi bile predstavljene takole:

"0" - bit 0 rezultata,

"9" - bit 1 rezultata itn.

Fustimo za trenutek tipkovnico in pogledimo pomembno dejstvo. Rekli smo, da ima CP na voljo 65535 v/i naslovov. Če dobro pogledate, moramo v naših ukazih v resnici vedno določiti 16-bitni naslov! Pri ukazu

IN A, (n)

je njegov višji del v registru A, nižji pa je številka v/i vrat (dejanski v/i naslov = $256 * A + n$). Enako velja za niz

IN r, (C).

Tu je nižji zlog naslova (številka v/i vrat) v registru C, višji zlog pa moramo pred ukazom določiti v registru B (dejanski v/i naslov = $256 * (B) + (C)$). Če bi torej v našem prejšnjem primeru želeli uporabiti ta niz ukazov, bi dali registru C vrednost FEH, z registrom B bi pa navedli, katero skupino želimo čitati.

Res je, da višji del naslova v teh ukazih mnogokrat nima nobenega vpliva in je odločilna le številka v/i vrat (posledico smo že omenili: različni naslovi lahko pomenijo isto). Vendar kljub temu ne pozabite: v bistvu morate vedno določiti 16-bitni naslov.

Vrnimo se zdaj k tipkovnici. Marsikdaj bi (predvsem pri

igrah) želeli naenkrat prečitati celo zgornjo vrsto (skupini 3 in 4). To lahko storimo tako, da damo registru A vrednost

1 1 0 0 1 1 = E7H (= 231).

Kot vidite, sta tako bit 3 kot bit 4 enaka 0. Na ta način se podatki sicer pomešajo in ne morete vedeti, ali je bila pritisnjena tipka "1" ali "0", "2" ali "9" itn., ker ti pari vplivajo na iste bite. Uporaben pa je ta način, kot smo rekli, zlasti za igre, ker lahko npr. tipki "5" in "8" služita za pomikanje v levo in desno, čeprav pripadata različnim skupinama.

Uporaba ukazov IN ter OUT je najhitrejši način branja tipkovnice. Priznam pa, da se vam po pravicu zdi precej okoren. Zato za branje tipk pogosto uporabljamo podprograme iz ROM-a, ki vam bodo gotovo bolj všeč. O njih bomo govorili v poglavju "Uporaba tipkovnice".

Ukaza IN in OUT nam dajeta veliko moč. Z njima lahko pošiljate in/ali sprejemate tudi podatke (se pravi električne impulze) z robnega priključka. Na ta način lahko upravljate vse, kar je priključeno na vaš računalnik - najsi bo to električna železnica, gospodinjski aparati ali vaš domači disko klub.

V tabeli na začetku poglavja ste opazili še ukaze

INI (IN AND INCREASE = noter in povečaj),

IND (IN AND DECREASE = noter in zmanjšaj),

INIR (IN, INCREASE AND REPEAT = noter, povečaj in ponovi),

INDR (IN, DECREASE AND REPEAT = noter, zmanjšaj in ponovi),

OUTI (OUT AND INCREASE = ven in povečaj),

OUTD (OUT AND DECREASE = ven in zmanjšaj),

OTIR (OUT, INCREASE AND REPEAT = ven, povečaj in ponovi) ter

OTDR (OUT, DECREASE AND REPEAT = ven, zmanjšaj in ponovi).

To so skupinski ukazi za vhod in izhod, in sicer štirje navadni (INI, IND, OUTI, OUTD) in štirje avtomatični (INIR, INDR, OTIR, OTDR). Ti ukazi napravijo IN (HL), (C) oz. OUT (C), (HL), povečajo oz. zmanjšajo HL ter zmanjšajo B. Avtomatični ukazi se ponavljajo, dokler B ne doseže vrednosti 0. Skupinski ukazi za vhod in izhod so zelo malo v rabi, zato se jim ne bomo natančneje posvečali.

UPORABA: Z rabo v/i ukazov se bomo pozabavali v poglavju Uporaba tipkovnice ter v igri "Ključar Martin in vražji metulj".