

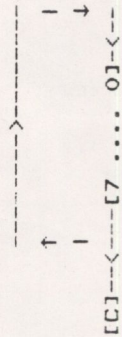
Vsebine registrov ali pomnilniških celic se ob tem ne spremeni. Edini rezultat tega niza ukazov je stanje ničelne zastavice: - če je vrednost preiskovanega bita 0, je zastavica dvignjena; - če je vrednost preiskovanega bita 1, je zastavica spuščena. To človeka nekoliko zbega, a si boste zlahka zapomnili takole: če ima bit vrednost nič, je ničelna zastavica dvignjena, sicer ne.

Primer uporabe teh ukazov boste našli na koncu naslednjega poglavja.

## KROŽNI IN DRUGI POMIKI

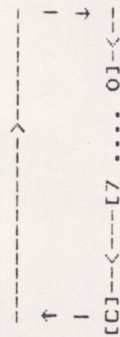
Pomiki so naslednja skupina ukazov. Na voljo jih je kar nekaj vrst, ki pa jih morate dobro razlikovati, da jih boste znali pravilno uporabljati. Kot operande lahko povsod pri teh ukazih uporabimo vse registre ter (HL), (IX+ dis) in (IY+ dis), tj. vsebine naslovov HL, IX+ dis in IY+ dis. Krožni pomik je možen v levo ali v desno ter lahko vključuje 8 ali 9 bitov. Pri slednjem je v premikanje vključen tudi bit prenosa Carry). Za primer si oglejmo skici 8- in 9-bitnega krožnega pomika v levo (v desno je povsem enak, razlikuje se le po smeri):

8-bitni krožni pomik (brez prenosa)



bit prenosa operand

9-bitni krožni pomik



Če označimo operand z "s", zapišemo ukaze za krožni pomik

```

RR s  (ROTATE RIGHT s) - 9-bitni v desno;
RL s  (ROTATE LEFT s)  - 9-bitni v levo;
RRC s (ROTATE RIGHT WITHOUT CARRY s) -
      8-bitni (brez prenosa) v desno;
RLC s (ROTATE LEFT WITHOUT CARRY s) -
      8-bitni (brez prenosa) v levo.
  
```

Kot ste videli, se pri 9-bitnem krožnem premiku prenese najvišji bit (bit 7) operanda v bit prenosa, bit prenosa v bit 0 operanda itn. Pri 8-bitnem krožnem pomiku se bit 7 operanda prenese v bit prenosa in v bit 0 operanda. Vsi omenjeni ukazi