

Tako zamenjamo registre, še vedno pa imamo za delo na voljo staro vrednost BC.

Naslednji ukaz iz te skupine ne vključuje zamenljivih registrov, temveč para DE in HL

EX DE, HL .

Z njim damo DE vrednost HL in obratno. Ukaz je zelo uporaben. Kot veste, je HL prednostni registrski par, vrednost, s katero želimo računati, pa je pogosto v DE.

Zadnji v tej skupini so ukazi

EX (SP), HL  
EX (SP), IX  
EX (SP), IY ,

.ki smo jih že spoznali v poglavju o skladu.

## UKAZI SET, RESET IN BIT

Ukazi v teh skupinah so, razen logičnih ukazov, edini, ki omogočajo spreminjanje ter preverjanje vrednosti posameznih bitov v registrih ali pomnilniških celicah. Ker je brskanje po bitih precej puščobno delo, so ti ukazi redkeje v rabi. Poleg tega porabi CP pogosto več časa za spreminjanje vrednosti posameznih bitov kot za spremembo celotnega zloga.

Pridejo pa trenutki, ko moramo vedeti, kakšna je vrednost nekega bita, ali jo moramo celo spremeniti. Le obdržati je treba v mislih, da je marsikdaj mogoče doseči isto z logičnimi operacijami.

Skupina "Set, Reset in Bit" omogoča, da bite "prižigamo" (jim damo vrednost 1), jih "ugašamo" (jim damo vrednost 0) ali preverjamo njihovo vrednost.

Prvi niz so ukazi "prižgi"

SET b, r  
SET b, (HL)  
SET b, (IX+ dis)  
SET b, (IY+ dis) .

Spomnite se, da bite v zlogu oštevilčimo od 0 do 7. Ukaz SET da bitu b v registru ali pomnilniški celici vrednost 1. Ob tem ne vpliva na nobeno zastavico.

Podobno velja za ukaze "ugasni" (RESET), ki uporabljajo iste operande, le da dajo bitu b vrednost 0

RES b, r  
RES b, (HL)  
RES b, (IX+ dis)  
RES b, (IY+ dis) .

Ukaz BIT bi moral pravzaprav imeti obliko "BIT ?", ker z njim preverjamo vrednost posameznega bita:

BIT b, r  
BIT b, (HL)  
BIT b, (IX+ dis)  
BIT b, (IY+ dis) .