



IPARI INFORMATIKAI KÖZPONT

Z 80-as sorozat  
VIII. rész:  
CPU utasításkészlet

Tervezési segédlet

"A mikroszámítógépek és alkalmazási rendszereik kutatása - fejlesztése" c.

OMFB Tárcaprogram

"A mikrogépes technika terjesztésének át-, illetve továbbképzési,  
tanácsadási, tájékoztatási szolgálatának működtetése" c. 4/b. alprogramja  
keretében készült

## TERVEZÉSI SEGÉDLET

**Készült az IPARI INFORMATIKAI KÖZPONT**

**Műszaki Információs és Elemző Osztályán**

**/ BUDAPEST V., Arany János u.24. /**

**Témafelelős: dr.Makra Ernőné**

**Munkatársak: Fehér Gyula, Fné Torma Magdolna**

**Zsombok Istvánné**

**Lektorálta: Benesóczky Dezső,**

**Dr. Boromisza Tamás**

Budapest, 1982.



**IPARI INFORMATIKAI KÖZPONT**

**Z 80-as sorozat**

**VIII.rész:**

**CPU utasításkészlet**

**Tervezési segédlet**

**Ezen tervezési segédlet**

része annak a Z-80 -as sorozatnak, amely a - "uP-k alkalmazása" c. témafigyelésünk 1981/4. és 1982/1., 2-3. sz. kérdőivein történő felmérés alapján - katalógus-hiánnyal küzdő kollegák részére készült.

A felhasználók további kérésére a sorozat egy részletes CPU utasításkészlettel bővült / Forrás: Zilog - 1982.május /.

A Z-80 -as sorozatunk így az alábbi részekből áll:

- I. rész: CPU
- II-III. rész: PIO, CTC
- IV. rész: SIO
- V. rész: COMBO
- VI. rész: DMA, ÓRA/RAM, VCU, A/D konverter
- VII. rész: Alkalmazástechnika példatár
- VIII. rész: CPU utasításkészlet

Forrás: ZILOG - MOSTEK /1981./

Z-80 CPU  
UTASÍTÁSKÉSZLETE

Forrás: ZILOG — 1982. május

## Bevezetés

A kézikönyv a Z80 CPU teljes utasításkészletének részletes ismertetését tartalmazza.

Az egyes utasításokat a műveleti kód, funkció, az assembler mnemonikus formátum és a műveleti kód /OP kód/ leírásával ismertetjük. Az utasítások funkciójának ismertetését az utasítás végrehajtási idejének, M ciklusainak, T órajelperiódusainak és a jelzőbit /flag/ regiszter érintett bitjeinek megadásával is szemléltetjük. Amennyiben a jelzőbit regiszter állapotát az utasítás nem befolyásolja, úgy ezt a leírásban külön nem tüntettük fel. Végül az utasítás működését egy példán keresztül illusztráljuk.

### Megjegyzés:

A végrehajtási idő /továbbiakban: V.i./ mindenegyus utasítás esetén 4 MHz-es CPU órajel frekvenciát feltételezve  $\mu$ s-ban értelmezendő.

Az utasítás végrehajtásához szükséges gépi ciklusok /M/ számát a teljes órajelperiódussal /T állapot/ együtt adjuk meg. Emellett az egyes M gépi ciklusokon belüli T órajelperiódusok számát is jelezzük. Pl.:

M ciklusok: 2      T állapotok: 7/4,3/      4 MHz V.i.: 1.75

Ez úgy értelmezendő, hogy az utasítás két gépi ciklusból áll. Az első gépi ciklus 4 órajelperiódusból /T állapotból/ áll, míg a második gépi ciklus 3 órajelperiódust tartalmaz. Így az utasítás összesen 7 órajelperiódus hosszúságú, ezért 1.75  $\mu$ s végrehajtási időt igényel.

A regiszter-formátumokban minden egyes utasítás esetén a legnagyobb helyiértékű a bal szélső bit, míg a legkisebb helyiértékű a jobb szélső bit.

# ADC HL,ss

Művelet: HL ← HL + ss + CY

Formátum: OP kód          Operandusok  
                  ADC                                  HL,ss

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | s | s | 1 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|

Leírás:

Az ss regiszterpár tartalma /a BC, DE, HL vagy SP regiszterpárok bármelyike/ az átvitel bittel (Carry Flag) együtt /a C jelzőbit az F regiszterben/ hozzáadódik a HL regiszterpár tartalmához, és az eredmény a HL regiszterbe kerül. Az ss operandus az assemblerrel fordított tárgykódban a következő:

| <u>Regiszterpár</u> | <u>ss</u> |    |
|---------------------|-----------|----|
| BC                  | 00        | 4A |
| DE                  | 01        | 5A |
| HL                  | 10        | 6A |
| SP                  | 11        | 7A |

M ciklusok: 4    T állapotok: 15 /4,4,4,3/    4 MHz V.i.: 3.75

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 11-es biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsoordulás áll fenn; egyébként törlődik
- N: törlődik
- C: 1-es állapotba kerül, ha a 15-ös biten átvitel keletkezik; egyébként törlődik

Példa:

Ha a BC regiszterpár 2222H-t, a HL regiszterpár 5437H-t tartalmaz, és az átvitel bit 1-be van állítva, az

ADC HL,BC

végrehajtása után a HL tartalma 765AH lesz.

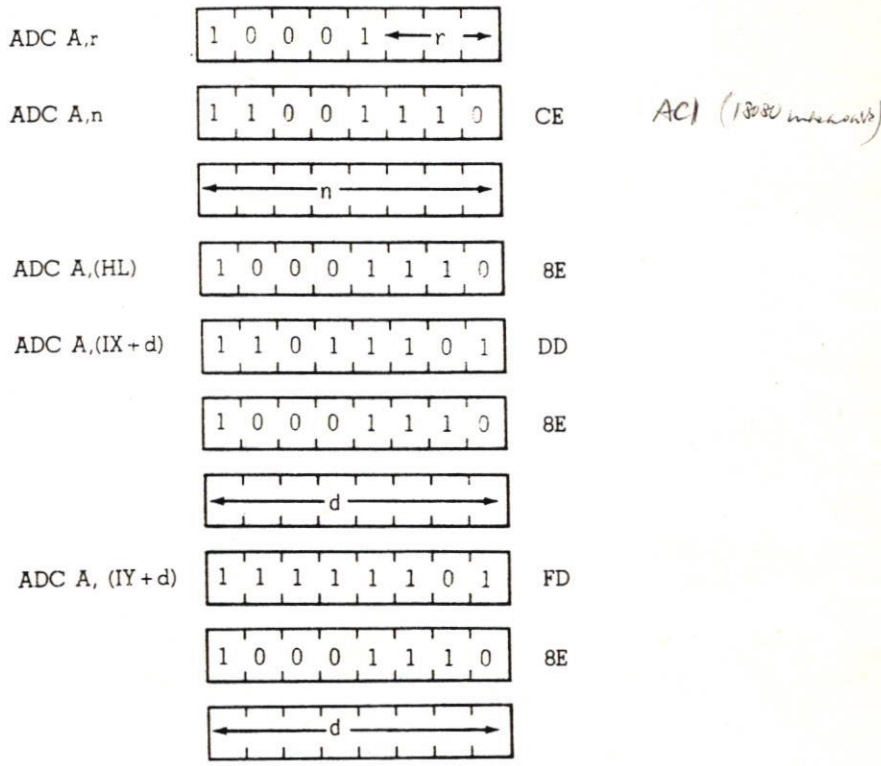
# ADC A,s

13080 ADC

Művelet:  $A \leftarrow A + s + CY$

Formátum:           OP kód           Operandusok  
   ADC                    A,s

Az s operandus r, n, HL, (IX+d) vagy (IY+d) bármelyike lehet, ahogy ez az ADD utasításnál is szerepel. A különböző OP kód-operandus kombinációkat tárgykódban a következők lehetnek:



az r a B,C,D,E,H,L vagy A regisztereket a fenti tárgykódban a következőképpen azonosítja:

| Regiszter | r   |    |
|-----------|-----|----|
| B         | 000 | 88 |
| C         | 001 | 89 |
| D         | 010 | 8A |
| E         | 011 | 8B |
| H         | 100 | 8C |
| L         | 101 | 8D |
| A         | 111 | 8F |



## ADC A,s

### Leírás:

Az s operandus az átvitel bittel együtt /C bit az F regiszterben/ hozzáadódik az akkumulátor tartalmához és az eredmény az akkumulátorban kerül tárolásra.

| UTASÍTÁS     | M CIKLUSOK | T ÁLLAPOTOK    | 4 MHz V.i. |
|--------------|------------|----------------|------------|
| ADC A,r      | 1          | 4              | 1.00       |
| ADC A,n      | 2          | 7 /4,3/        | 1.75       |
| ADC A,(HL)   | 2          | 7 /4,3/        | 1.75       |
| ADC A,(IX+d) | 5          | 19 /4,4,3,5,3/ | 4.75       |
| ADC A,(IY+d) | 5          | 19 /4,4,3,5,3/ | 4.75       |

### A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsondulás áll fenn; egyébként törlődik
- N: törlődik
- C: 1-es állapotba kerül, ha a 7-es biten átvitel keletkezik; egyébként törlődik

### Példa:

Abban az esetben, ha az akkumulátor 16H-t tartalmaz, az átvitel bit 1-es állapotban van, a HL regiszterpár 6666H-t és a 6666H című memóriarekesz 10H-t tartalmaz, az

ADC A,(HL)

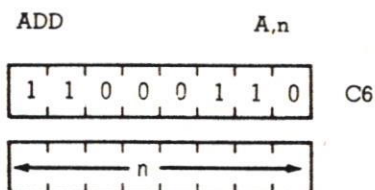
végrehajtása után az akkumulátor tartalma 27H lesz.

# ADD A,n

18080 ADI n

Művelet:  $A \leftarrow A + n$

Formátum: OP kód            Operandusok



## Leírás:

Az n egész szám hozzáadódik az akkumulátor tartalmához és az eredmény az akkumulátorban kerül tárolásra.

M ciklusok: 2      T állapotok: 7/4,3/      4 MHz V.i.: 1.75

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsordulás áll fenn; egyébként törlődik
- N: törlődik
- C: 1-es állapotba kerül, ha a 7-es biten átvitel keletkezik; egyébként törlődik

## Példa:

Ha az akkumulátor tartalma 23H, az

ADD A,33H

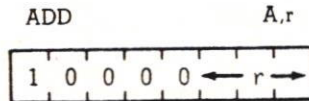
végrehajtásának eredményeként az akkumulátor tartalma 56H lesz.

# ADD A,r

18080 ADD,r

Művelet:  $A \leftarrow A + r$

Formátum: OP kód Operandumok



Leírás:

Az r regiszter tartalma hozzáadódik az akkumulátor tartalmához, és az eredmény az akkumulátorban tárolódik. Az r szimbólum tárgykódban a kódolt A,B,C,D,E,H vagy L regisztereket azonosítja:

| <u>Regiszter</u> | <u>r</u> |    |
|------------------|----------|----|
| A                | 111      | 87 |
| B                | 000      | 84 |
| C                | 001      | 81 |
| D                | 010      | 82 |
| E                | 011      | 83 |
| H                | 100      | 86 |
| L                | 101      | 85 |

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsondulás áll fenn; egyébként törlődik
- N: törlődik
- C: 1-es állapotba kerül, ha a 7-es biten átvitel keletkezik; egyébként törlődik

Példa:

Ha az akkumulátor tartalma 44H és a C regiszteré 11H, az

ADD A,C

végrehajtásának eredményeként az akkumulátor tartalma 55H lesz.

# ADD A,(HL)

19030 ADD M

Művelet:  $A \leftarrow A + (HL)$

Formátum:           OP kód           Operandusok  
                  ADD                A,(HL)

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 86

Leírás:

A HL regiszterpár által tartalmazott memóriacímen lévő byte hozzáadódik az akkumulátor tartalmához és az eredmény az akkumulátorban tárolódik.

M ciklusok: 2       T állapotok: 7/4,3/       4 MHz V.i.: 1.75

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsoordulás áll fenn; egyébként törlődik
- N: törlődik
- C: 1-es állapotba kerül, ha a 7-es biten átvitel keletkezik; egyébként törlődik

Példa:

Ha az akkumulátor tartalma A0H, a HL regiszterpár tartalma 2323H, és a 2323H című memóriarekeszben lévő byte 08H, az

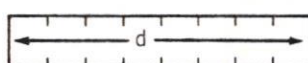
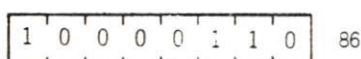
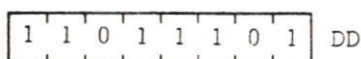
ADD A,(HL)

végrehajtásának eredményeként az akkumulátor tartalma A8H lesz.

# ADD A,(IX+d)

Művelet:  $A \leftarrow A + (IX+d)$

Formátum:      OP kód              Operandumok  
                  ADD                    A,(IX+d)



## Leírás:

Az index regiszter /IX regiszterpár/ tartalma hozzáadódik a d eltolási címhez. Az így kapott eredmény egy memóriacímre mutat, melynek tartalma hozzáadódik az akkumulátor tartalmához és az eredmény az akkumulátorban kerül tárolásra.

M ciklusok: 5      T állapotok: 19/4,4,3,5,3/    4 MHz V.i.: 4.75

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsondulás áll fenn; egyébként törlődik
- N: törlődik
- C: 1-es állapotba kerül, ha a 7-es biten átvitel keletkezik; egyébként törlődik

## Példa:

Ha az akkumulátor tartalma 11H, az IX index regiszter tartalma 1000H, és az 1005H című memóriarekesz tartalma 22H, az

ADD A,(IX+5H)

végrehajtásának eredményeként az akkumulátor tartalma 33H lesz.

# ADD A,(IY+d)

Művelet:  $A \leftarrow A + (IY+d)$

Formátum: OP kód Operandumok

ADD A,(IY+d)

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 FD

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 86

|       |  |  |  |  |  |  |  |
|-------|--|--|--|--|--|--|--|
| ← d → |  |  |  |  |  |  |  |
|-------|--|--|--|--|--|--|--|

Leírás:

Az index regiszter /IY regiszterpár/ tartalma hozzáadódik a d eltolási címhez. Az így kapott eredmény egy memóriacímre mutat, melynek a tartalma hozzáadódik az akkumulátor tartalmához, és az eredmény az akkumulátorban kerül tárolásra.

M ciklusok: 5 T állapotok: 19/4,4,3,5,3/ 4 MHz V.i.: 4.75

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsondulás áll fenn; egyébként törlődik
- N: törlődik
- C: 1-es állapotba kerül, ha a 7-es biten átvitel keletkezik; egyébként törlődik

Példa:

Ha az akkumulátor tartalma 11H, az IY index regiszterpár tartalma 1000H, és az 1005H című memóriarekesz tartalma 22H, az

ADD A,(IY+5H)

végrehajtásának eredményeként az akkumulátor tartalma 33H lesz.

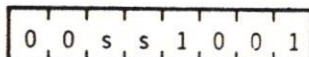
# ADD HL,ss

13040 DAD ss

Művelet: HL ← HL + ss

Formátum: OP kód Operandumok

ADD HL,ss



## Leírás:

Az ss regiszterpár /a BC, DE, HL vagy SP regiszterpárok bármelyike/ tartalma hozzáadódik a HL regiszterpár tartalmához, és az eredmény a HL regiszterbe kerül. Az ss operandus jelentése:

| <u>Regiszterpár</u> | <u>ss</u> |    |
|---------------------|-----------|----|
| BC                  | 00        | 09 |
| DE                  | 01        | 19 |
| HL                  | 10        | 29 |
| SP                  | 11        | 39 |

M ciklusok: 3 T állapotok: 11/4,4,3/ 4 MHz V.i.: 2.75

A jelzőbit regiszter érintett bitjei:

- S: változatlan marad
- Z: változatlan marad
- H: 1-es állapotba kerül, ha a 11-es biten átvitel keletkezik; egyébként törlődik
- P/V: változatlan marad
- N: törlődik
- C: 1-es állapotba kerül, ha a 15-ös biten átvitel keletkezik; egyébként törlődik

## Példa:

Ha a HL regiszterpár tartalmazza a 4242H egész számot és a DE regiszterpár 1111H-t, az

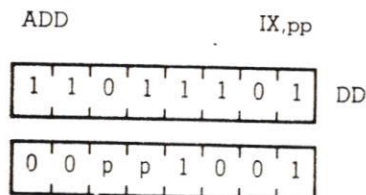
ADD HL,DE

végrehajtása után a HL regiszterpár tartalma 5353H lesz.

# ADD IX,pp

Művelet:  $IX \leftarrow IX + pp$

Formátum: OP kód            Operandumok



## Leírás:

A pp regiszterpár /a BC, DE, IX vagy SP regiszterpárok bármelyike/ tartalma hozzáadódik az IX index regiszter tartalmához, és az eredmény az IX regiszterbe kerül. A pp operandus a következőket jelenti:

| <u>Regiszterpár</u> | <u>pp</u> |    |
|---------------------|-----------|----|
| BC                  | 00        | 09 |
| DE                  | 01        | 19 |
| IX                  | 10        | 29 |
| SP                  | 11        | 39 |

M ciklusok: 4      T állapotok: 15/4,4,4,3/      4 MHz V.i.: 3.75

## A jelzőbit regiszter érintett bitjei:

- S: változatlan marad
- Z: változatlan marad
- H: 1-es állapotba kerül, ha a 11-es biten átvitel keletkezik; egyébként törlődik
- P/V: változatlan marad
- N: törlődik
- C: 1-es állapotba kerül, ha a 15-ös biten átvitel keletkezik; egyébként törlődik

## Példa:

Ha az IX index regiszter tartalma 3333H és a BC regiszterpár tartalma 5555H, az

ADD IX,BC

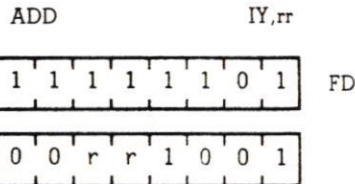
végrehajtásának eredményeként az IX tartalma 8888H lesz.



# ADD IY,rr

Művelet:  $IY \leftarrow IY + rr$

Formátum: OP kód            Operandusok



## Leírás:

Az rr regiszterpár /a BC, DE, IY vagy SP regiszterpárok bármelyike/ tartalma hozzáadódik az IY index regiszter tartalmához, és az eredmény az IY regiszterbe kerül. Az rr operandus jelentése a következő:

| <u>Regiszterpár</u> | <u>rr</u> |    |
|---------------------|-----------|----|
| BC                  | 00        | 0j |
| DE                  | 01        | 1j |
| IY                  | 10        | 2j |
| SP                  | 11        | 3j |

M ciklusok: 4      T állapotok: 15/4,4,4,3/      4 MHz V.i.: 3.75

A jelzőbit regiszter érintett bitjei:

- S: változatlan marad
- Z: változatlan marad
- H: 1-es állapotba kerül, ha a 11-es biten átvitel keletkezik; egyébként törlődik
- P/V: változatlan marad
- N: törlődik
- C: 1-es állapotba kerül, ha a 15-ös biten átvitel keletkezik; egyébként törlődik

## Példa:

Ha az IY index regiszter tartalma 3333H és a BC regiszterpár tartalma 5555H, az

ADD IY,BC

végrehajtása után az IY tartalma 8888H lesz.

# AND s

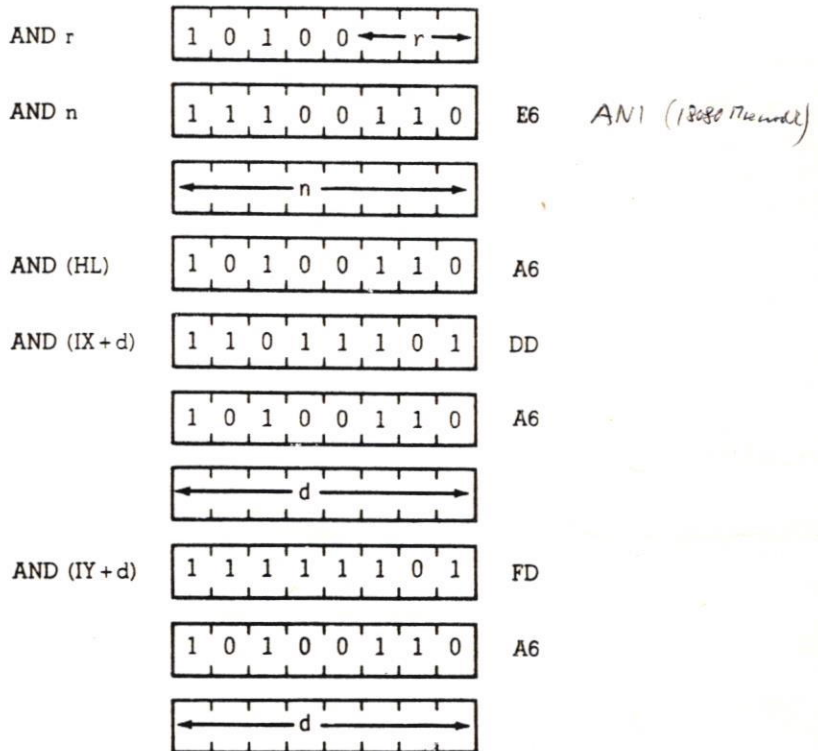
18D3D ANA s

Művelet:  $A \leftarrow A \wedge s$

Formátum: OP kód            Operandusok

AND                            s

Az s operandus az r, n, (HL), (IX+d) vagy (IY+d) bármelyike, amint azt az ezzel az utasítással analóg ADD utasításoknál meghatároztuk. Ezek a különböző lehetséges OP kód-operandus kombinációk a tárgykódban a következők lehetnek:



r B, C, D, E, H, L vagy A regisztereket azonosítja az alábbiak szerint:

| <u>Regiszter</u> | <u>r</u> |    |
|------------------|----------|----|
| B                | 000      | A0 |
| C                | 001      | A1 |
| D                | 010      | A2 |
| E                | 011      | A3 |
| H                | 100      | A4 |
| L                | 101      | A5 |
| A                | 111      | A7 |

# AND s

## Leírás:

Olyan bitenkénti logikai ÉS művelet, amely az s operandus által meghatározott byte, és az akkumulátor által tartalmazott byte között valósul meg. Az eredmény az akkumulátorban tárolódik.

| UTASÍTÁS   | M CIKLUSOK | T ÁLLAPOTOK   | 4 MHz V.i. |
|------------|------------|---------------|------------|
| AND r      | 1          | 4             | 1.00       |
| AND n      | 2          | 7/4,3/        | 1.75       |
| AND (HL)   | 2          | 7/4,3/        | 1.75       |
| AND (IX+d) | 5          | 19/4,4,3,5,3/ | 4.75       |
| AND (IY+d) | 5          | 19/4,4,3,5,3/ | 4.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: törlődik

## Példa:

Ha a B regiszter 7BH-t /01111011/, és az akkumulátor C3H-t /11000011/ tartalmaz, az

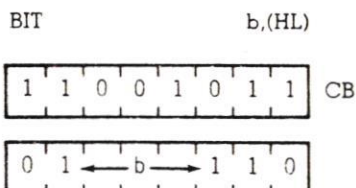
AND B

végrehajtásának eredményeként az akkumulátor tartalma 43H /01000011/ lesz.

# BIT b,(HL)

Művelet:  $Z \leftarrow (\overline{HL})_b$

Formátum: OP kód            Operandumok



## Leírás:

Ennek az utasításnak a végrehajtása után a jelzőbit regiszterben lévő Z bit a HL regiszterpár tartalma által megcímzett memóriarekeszen belül a b operandus által kijelölt bit komplementjét tartalmazza. A b operandus jelentése:

| <u>Vonatkozó bit</u> | <u>b</u> |    |
|----------------------|----------|----|
| 0                    | 000      | 4C |
| 1                    | 001      | 5E |
| 2                    | 010      | 58 |
| 3                    | 011      | 5F |
| 4                    | 100      | 66 |
| 5                    | 101      | 6E |
| 6                    | 110      | 76 |
| 7                    | 111      | 7E |

M ciklusok: 3    T állapotok: 12/4,4,4/    4 MHz V.i.: 3.00

A jelzőbit regiszter érintett bitjei:

- S: ismeretlen
- Z: 1-es állapotba kerül, ha a specifikált bit 0; egyébként törlődik
- H: 1-es állapotba kerül
- P/V: ismeretlen
- N:** törlődik
- C: változatlan marad

## Példa:

Ha a HL regiszterpár 4444H-t, és a 4-es bit a 4444H című memóriarekeszben 1-et tartalmaz, a

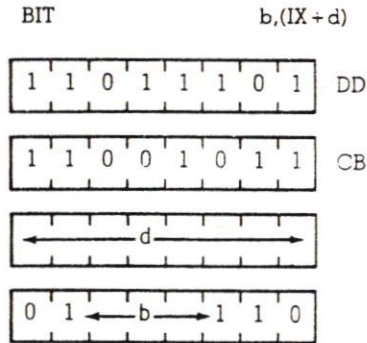
BIT 4,(HL)

végrehajtásának eredményeként a Z bit 0-t, a 4-es bit pedig a 4444H című memóriarekeszben változatlanul 1-et fog tartalmazni.

# BIT b,(IX+d)

Művelet:  $Z \leftarrow (\overline{IX+d})_b$

Formátum: OP kód            Operandusok



## Leírás:

Ennek az utasításnak a végrehajtása után a Z bit tartalmazni fogja az IX regiszterpár /IX index regiszter/ és a d 2-es komplementű egész számként értelmezett címkiegészítés összege által meghatározott memóriarekesz azon bitjének negáltját, melyet a b operandus jelöl ki. A b operandus jelentése:

| Vonatkozó bit | b   |    |
|---------------|-----|----|
| 0             | 000 | 4C |
| 1             | 001 | 4E |
| 2             | 010 | 5C |
| 3             | 011 | 5E |
| 4             | 100 | 6C |
| 5             | 101 | 6E |
| 6             | 110 | 7C |
| 7             | 111 | 7E |

M ciklusok: 5    T állapotok: 20/4,4,3,5,4/    4 MHz V.i.: 5.

A jelzőbit regiszter érintett bitjei:

- S: ismeretlen
- Z: 1-es állapotba kerül, ha a b operandus által specifikált bit 0; egyébként törlődik
- H: 1-es állapotba kerül
- P/V: ismeretlen
- N: törlődik
- C: változatlan marad

## **BIT b,(IX+d)**

---

Példa:

Ha az IX index regiszter 2000H-t tartalmaz, és a 6-os bit a 2004H című memóriarekeszben 1-et tartalmaz, a

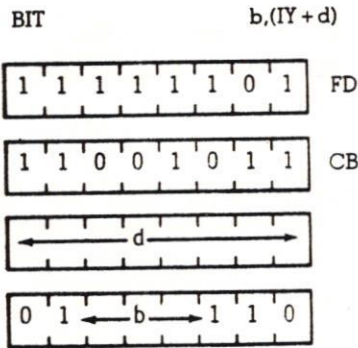
BIT 6 (IX+4H)

végrehajtásának eredményeként a Z bit 0-t fog tartalmazni, a 2004H című memóriarekeszben a 6-os bit pedig változatlanul 1-et.

## BIT $b,(IY+d)$

Művelet:  $Z \leftarrow (\overline{IY+d})_b$

Formátum:            OP kód            Operandumok



Leírás:

Ennek az utasításnak a végrehajtása után a Z bit tartalmazni fogja az IY regiszterpár /IY index regiszter/ és a d 2-es komplementű egész számként értelmezett címkiegészítés összege által meghatározott memóriarekesz azon bitjének negáltját, melyet a b operandus jelöl ki. A b operandus jelentése:

| Vonatkozó bit | b   |    |
|---------------|-----|----|
| 0             | 000 | 36 |
| 1             | 001 | 4E |
| 2             | 010 | 56 |
| 3             | 011 | 5E |
| 4             | 100 | 66 |
| 5             | 101 | 6E |
| 6             | 110 | 76 |
| 7             | 111 | 7E |

M ciklusok: 5    T állapotok: 20/4,4,3,5,4/    4 MHz V.i.: 5.00

A jelzőbit regiszter érintett bitjei:

- S: ismeretlen
- Z: 1-es állapotba kerül, ha a b operandus által specifikált bit 0; egyébként törlődik
- H: 1-es állapotba kerül
- P/V: ismeretlen

## BIT b,(IY+d)

---

N: törlődik  
C: változatlan marad

### Példa:

Ha az IY index regiszter tartalma 2000H, és a 6-os bit a 2004H című memóriarekeszben 1-et tartalmaz, a

BIT 6,(IY+4H)

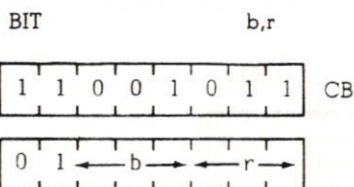
végrehajtásának eredményeként a Z bit változatlanul 0-t, a 6-os bit a 2004H című memóriarekeszben továbbra is 1-et tartalmaz.



# BIT b,r

Művelet:  $Z \leftarrow \bar{r}_b$

Formátum: OP kód            Operandusok



## Leírás:

Az utasítás az r operandus által meghatározott regiszternek a b operandus által kijelölt bitjét invertálva a jelzőbit regiszter Z bitjébe tölti. Az r és b operandusok tárgykódban a következőket jelenthetik:

| <u>Vonatkozó bit</u> | <u>b</u> | <u>Regiszter</u> | <u>r</u> |
|----------------------|----------|------------------|----------|
| 0                    | 000      | B                | 000      |
| 1                    | 001      | C                | 001      |
| 2                    | 010      | D                | 010      |
| 3                    | 011      | E                | 011      |
| 4                    | 100      | H                | 100      |
| 5                    | 101      | L                | 101      |
| 6                    | 110      | A                | 111      |
| 7                    | 111      |                  |          |

M ciklusok: 2      T állapotok: 8/4,4/      4 MHz V.i.: 2.00

## A jelzőbit regiszter érintett bitjei:

- S: ismeretlen
- Z: 1-es állapotba kerül, ha a b operandus által specifikált bit 0; egyébként törlődik
- H: 1-es állapotba kerül
- P/V: ismeretlen
- N: törlődik
- C: változatlan marad

## Példa:

Ha a 2-es bit a B regiszterben 0-t tartalmaz, a

BIT 2,B

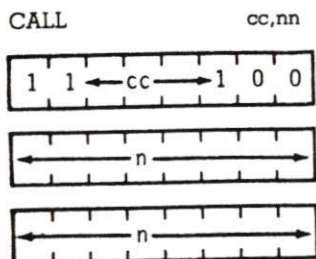
végrehajtásának eredményeként a Z bit 1-et fog tartalmazni, a 2-es bit a B regiszterben pedig 0 marad.

|               |    |    |    |    |    |    |    |
|---------------|----|----|----|----|----|----|----|
| gépi kódok r→ | B  | C  | D  | E  | H  | L  | A  |
| <u>CB</u> b   |    |    |    |    |    |    |    |
| ∅             | 4∅ | 41 | 42 | 43 | 44 | 45 | 47 |
| 1             | 48 | 49 | 4A | 4B | 4C | 4D | 4F |
| 2             | 5∅ | 51 | 52 | 53 | 54 | 55 | 57 |

# CALL cc,nn

Művelet: Ha cc igaz:  $(SP-1) \leftarrow PC_H$   
 $(SP-2) \leftarrow PC_L, PC \leftarrow nn$

Formátum: OP kód                      Operandusok



Megjegyzés: A fenti tárgy kód két n operandusa közül az első a két byte-os memóriacím alacsonyabb helyiértékű byte-ja.

## Leírás:

Ha a cc feltétel teljesül /logikailag igaz/, ez az utasítás tárolja a programszámláló /PC/ pillanatnyi tartalmát a külső veremtár tetejére, majd az nn operandust tölti a PC-be, és az így kapott PC tartalom által kijelölt memóriarekeszből egy szubrutin első OP kódja kerül kiolvasásra. /A szubrutin végén az eredeti programhoz való visszatérés egy a veremtár tetejét a PC-be visszahelyező RET utasítással biztosítható./ Ha a cc feltétel nem teljesül /hamis/, akkor a programszámláló tartalma a szokásosnak megfelelően lesz inkrementálva, és a program végrehajtása a sorrendben következő utasítással folytatódik. A művelet során először a veremtár mutató /SP/ pillanatnyi tartalmának 1-el való dekrementálása, majd a PC tartalom magasabb helyiértékű byte-jának az így kapott címre való kihelyezése történik meg. Ezután az SP ismét dekrementálódik és a PC tartalom alacsonyabb helyiértékű byte-ja töltődik az így kapott címre, azaz a veremtár tetejére.

Megjegyzés: Mivel a CALL cc,nn 3 byte-os utasítás, a programszámláló a tárolási művelet végrehajtása előtt már hárommal inkrementálódik. A cc feltétel nyolc olyan állapot valamelyikéként programozott, amely a jelzőbit regiszter /F regiszter/ feltételbitjeinek felel meg. Ez a nyolc állapot a következő táblázat szerint definiálható, amely ugyanakkor a tárgy kód megfelelő cc bit mezőjét is specifikálja:

# CALL cc,nn

|    | <u>cc</u> | <u>Feltétel</u>     | <u>A vizsgált bit</u> |        |
|----|-----------|---------------------|-----------------------|--------|
| C4 | 000       | NZ nem zérus        | Z                     | CNZ nn |
| CC | 001       | Z zérus             | Z                     | CZ nn  |
| D4 | 010       | NC nincs átvitel    | C                     | CNC nn |
| DC | 011       | C átvitel           | C                     | CC nn  |
| E4 | 100       | PO páratlan paritás | P/V                   | CPO nn |
| EC | 101       | PE páros paritás    | P/V                   | CPE nn |
| F4 | 110       | P pozitív előjel    | S                     | CP nn  |
| FC | 111       | M negatív előjel    | S                     | CM nn  |

Ha cc teljesül:

M ciklusok: 5 T állapotok: 17/4,3,4,3,3/ 4 MHz V.i.: 4.25

Ha cc nem teljesül:

M ciklusok: 3 T állapotok: 10/4,3,3/ 4 MHz V.i.: 2.50

Példa:

Ha a C bit törölt állapotban van, a programszámláló tartalma 1A47H, a veremtár mutatóé 3002H, a memóriarekeszeké pedig az alábbi:

| <u>Rekesz</u> | <u>Tartalom</u> |
|---------------|-----------------|
| 1A47H         | D4H             |
| 1A48H         | 35H             |
| 1A49H         | 21H             |

és ha az 1A47H-val kezdődő utasítás hozzáférési szekvencia megkezdődik, akkor a D43521H 3 byte-os utasítás kerül a CPU-hoz végrehajtásra. Ennek a mnemonikus ekvivalense:

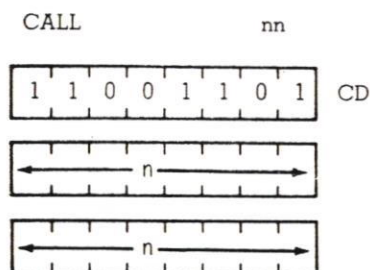
CALL NC,2135H

Ennek az utasításnak a végrehajtása után a 3001H című memóriarekesz tartalma 1AH, a 3000H című memóriarekesz tartalma 4AH, a veremtár mutató /SP/ tartalma 3000H, és a programszámláló tartalma 2135H lesz, amely a most végrehajtandó szubrutin első OP kódjának címét jelöli ki.

# CALL nn

Művelet:  $(SP-1) \leftarrow PC_H$ ,  $(SP-2) \leftarrow PC_L$ ,  $PC \leftarrow nn$

Formátum: OP kód                      Operandumok



Megjegyzés: A fenti tárgykód két n operandusa közül az első a két byte-os memóriacím alacsonyabb helyiértékű byte-ja.

## Leírás:

A programszámláló regiszter pillanatnyi tartalmának a külső memória veremtár tetejére történő mentése után az nn operandus töltődik a programszámláló regiszterbe /PC-be/, amely így arra a memóriacímre mutat, ahonnan egy szubrutin első OP kódja kerül kiemelésre. /A szubrutin végén az eredeti programhoz való visszatérés egy a veremtár tetejét a PC-be visszahelyező RET utasítással biztosítható./ A mentési művelet során először a veremtár mutató /SP regiszterpár/ pillanatnyi tartalmának 1-el való dekrementálása, majd a PC tartalom magasabb helyiértékű byte-jának az így kapott címre való kihelyezése történik meg. Ezután az SP ismét dekrementálódik, és a PC tartalom alacsonyabb helyiértékű byte-ja töltődik az így kapott címre, azaz a veremtár tetejére.

Megjegyzés: Mivel a CALL nn 3 byte-os utasítás, a programszámláló a mentési művelet végrehajtása előtt már 3-mal inkrementálódik.

M ciklusok: 5      T állapotok: 17/4,3,4,3,3/    4 MHz V.i.: 4.2

## Példa:

Abban az esetben, ha a programszámláló tartalma 1A47H, a veremtár mutatóé 3002H, a memóriarekeszeké pedig a következők szerint alakul:

## CALL nn

---

| <u>Rekesz</u> | <u>Tartalom</u> |
|---------------|-----------------|
| 1A47H         | CDH             |
| 1A48H         | 35H             |
| 1A49H         | 21H             |

és ha az 1A47H-val kezdődő utasítás hozzáférési szekvencia megkezdődik, akkor a CD3521H 3 byte-os utasítás kerül a CPU-hoz végrehajtásra. Ennek mnemonikus ekvivalense:

CALL 2135H

Ennek az utasításnak a végrehajtása után a 3001H memóriacím tartalma 1AH, a 3000H című memóriarekesz tartalma 4AH, az SP veremtár mutató /stack pointer/ tartalma 3000H, és a programszámláló tartalma 2135H lesz, amely a most végrehajtandó szubrutin első OP kódjának címét jelöli ki.

# CCF

18030 CMC

---

Művelet:  $CY \leftarrow \overline{CY}$

Formátum: OP kód

CCF

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 3F

Leírás:

A C bit a jelzőbit regiszterben invertálódik.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

A jelzőbit regiszter érintett bitjei:

- S: változatlan marad
- Z: változatlan marad
- H: az előző átvitel értékét veszi fel
- P/V: változatlan marad
- N: törlődik
- C: 1-es állapotba kerül, ha az utasítás végrehajtását megelőzően az átvitel bit /CY/ 0 volt; egyébként törlődik

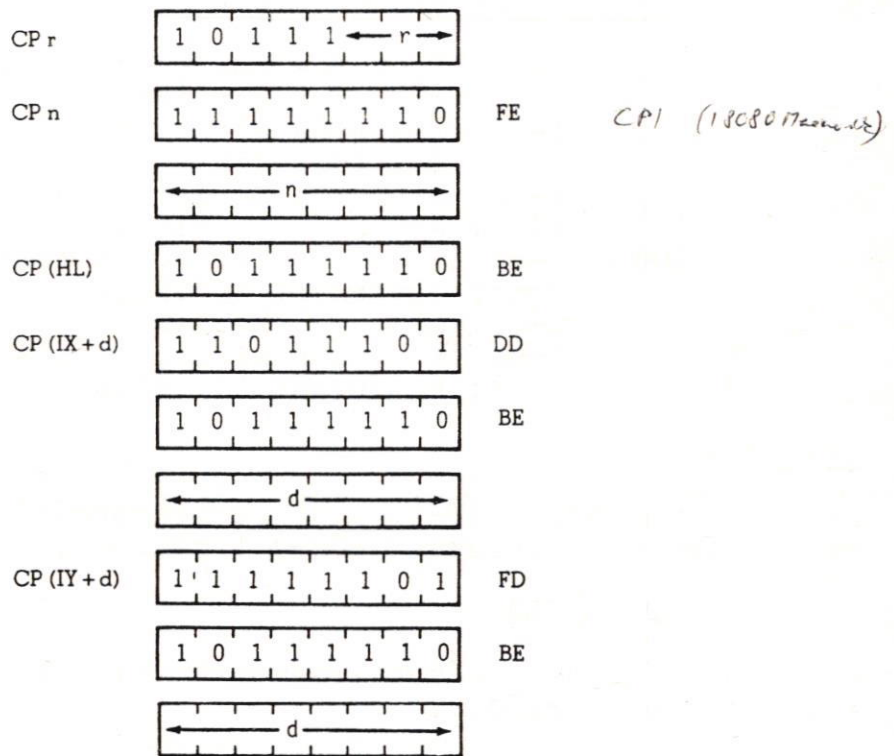
# CP s

18080 CMP

Művelet: A -s

Formátum: OP kód                      Operandusok  
                                CP    s

Az s operandus r, n, (HL), (IX+d) vagy (IY+d) bármelyike lehet, úgy, mint az ADD utasításoknál. Ezek a különböző lehetséges OP kód-operandus-kombinációk a tárgykódban a következők lehetnek:



r a fenti tárgykódban a következőképpen kódolt B,C,D,E,H,L vagy A regisztereket azonosítja:

| <u>Regiszter</u> | <u>r</u> |    |
|------------------|----------|----|
| B                | 000      | 88 |
| C                | 001      | 89 |
| D                | 010      | 8A |
| E                | 011      | 8B |
| H                | 100      | 8C |
| L                | 101      | 8D |
| A                | 111      | 8F |

## Leírás:

Az s operandus tartalma az akkumulátor tartalmával kerül összehasonlításra. Az összehasonlítás eredményét a jelzőbit

# CP s

---

regiszter megfelelő bitjének beállítása jelzi.

| UTASÍTÁS  | M CIKLUSOK | T ÁLLAPOTOK   | 4 MHz V.i. |
|-----------|------------|---------------|------------|
| CP r      | 1          | 4             | 1.00       |
| CP n      | 2          | 7/4,3/        | 1.75       |
| CP (HL)   | 2          | 7/4,3/        | 1.75       |
| CP (IX+d) | 5          | 19/4,4,3,5,3/ | 4.75       |
| CP (IY+d) | 5          | 19/4,4,3,5,3/ | 4.75       |

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsordulás áll fenn; egyébként törlődik
- N: 1-es állapotba kerül
- C: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik

Példa:

Ha az akkumulátor 63H-t, a HL regiszterpár 6000H-t, és a 6000H című memóriarekesz 60H-t tartalmaz, a

CP (HL)

utasítás végrehajtásának eredményeként a P/V bit a jelzőbit regiszterben törlődik.



# CPD

Művelet: A  $\leftarrow$ (HL), HL  $\leftarrow$  HL-1, BC  $\leftarrow$  BC-1

Formátum: OP kód

CPD

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 A9

## Leírás:

A HL regiszterpár által megcímezett memóriarekesz tartalma az akkumulátor regiszter tartalmával kerül összehasonlításra. Az összehasonlítás eredményeként egy feltétel bit fog 1-es állapotba kerülni. A HL és a byte számláló /BC regiszterpár/ tartalma dekrementálódik.

M ciklusok: 4    T állapotok: 16/4,4,3,5/    4 MHz V.i.: 4.00

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha  $A=(HL)$ ; egyébként törlődik
- H: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha  $BC-1 \neq 0$ ; egyébként törlődik
- N: 1-es állapotba kerül
- C: változatlan marad

## Példa:

Ha a HL regiszterpár 1111H-t, az 1111H című memóriarekesz 3BH-t, az akkumulátor 3BH-t, és a byte számláló 0001H-t tartalmaz, a

CPD

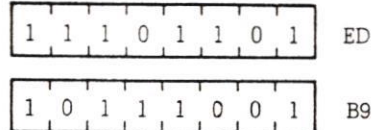
utasítás végrehajtása után a byte számláló 0000H-t, a HL regiszterpár 1110H-t fog tartalmazni, a Z bit a jelzőbit regiszterben 1-es állapotba kerül, és a P/V bit törlődik. Az utasítás az akkumulátor, vagy az 1111H című memóriarekesz tartalmára nem lesz hatással.

# CPDR

Művelet:  $A-(HL)$ ,  $HL \leftarrow HL-1$ ,  $BC \leftarrow BC-1$

Formátum: OP kód

CPDR



## Leírás:

A HL regiszterpár által megcímezett memóriarekesz tartalma az akkumulátor regiszter tartalmával kerül összehasonlításra. Az összehasonlítás eredményeként egy feltétel bit fog 1-es állapotba kerülni. A HL és a BC /Byte számláló/ regiszter dekrementálódik. Ha a dekrementálás eredményeként a BC tartalma 0-vá válik, vagy ha  $A=(HL)$ , az utasítás befejeződik. Ha BC tartalma nem zérus és  $A \neq (HL)$ , a programszámláló 2-vel dekrementálódik, és az utasítás megismétlődik.

Megjegyzés: ha BC tartalma az utasítás végrehajtása előtt már 0-ra van beállítva, az utasítás 64Kbyte-on keresztül fog megismétlődni, feltéve, hogy az egymást követő összehasonlítások során nem történt megegyezés. Minden memóriához fordulást követően két frissítési ciklus /refresh cycle/ zajlik le, és ilyenkor a megszakítások /interruptok/ is érvényesülnek.

## BC $\neq$ 0 és A $\neq$ (HL) esetén:

M ciklusok: 5    T állapotok: 21/4,4,3,5,5/    4 MHz V.i.: 5.25

## BC=0 vagy A=(HL) esetén:

M ciklusok: 4    T állapotok: 16/4,4,3,5/    4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha  $A=(HL)$ ; egyébként törlődik
- H: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha  $BC-1 \neq 0$ ; egyébként törlődik

## CPDR

---

N: 1-es állapotba kerül  
C: változatlan marad

### Példa:

Ha a HL regiszterpár 1118H-t, az akkumulátor F3H-t, és a byte számláló 0007H-t tartalmaz, a memóriarekeszek tartalma pedig az alábbiak szerint alakul:

(1118H) : 52H  
(1117H) : 00H  
(1116H) : F3H

akkor a

CPDR

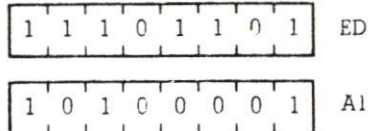
utasítás végrehajtása után a HL regiszterpár tartalma 1115H, a byte számláló tartalma 0004H lesz, a P/V bit 1-es állapotba kerül, és a Z bit is 1-es állapotba kerül.

# CPI

Művelet:  $A-(HL)$ ,  $HL \leftarrow HL+1$ ,  $BC \leftarrow BC-1$

Formátum: OP kód

CPI



## Leírás:

A HL regiszterpár által megcímezett memóriarekesz tartalma az akkumulátor regiszter tartalmával kerül összehasonlításra. Az összehasonlítás eredményeként egy feltétel bit fog 1-es állapotba kerülni. Azután a HL tartalma inkrementálódik, és a byte számláló /BC regiszterpár/ tartalma dekrementálódik.

M ciklusok: 4    T állapotok: 16/4,4,3,5/    4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha  $A=(HL)$ ; egyébként törlődik
- H: 1-es állapotba kerül, ha elulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha  $BC-1 \neq 0$ ; egyébként törlődik
- N: 1-es állapotba kerül
- C: változatlan marad

## Példa:

Ha a HL regiszterpár 1111H-t tartalmaz, az 1111H című memóriarekesz 3BH-t, az akkumulátor 3BH-t, és a byte számláló 0001H-t, a

CPI

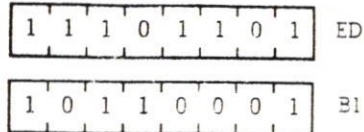
végrehajtásának eredményeként a byte számláló 0000H-t, a HL regiszterpár 1112H-t fog tartalmazni, a Z bit 1-es állapotba kerül, és a P/V bit törlődik. Az utasítás az akkumulátor, vagy az 1111H című memóriarekesz tartalmára nem lesz hatással.

# CPIR

Művelet:  $A-(HL)$ ,  $HL \leftarrow HL+1$ ,  $BC \leftarrow BC-1$

Formátum: OP kód

CPIR



## Leírás:

A HL regiszterpár által megcímezett memóriarekesz tartalma az akkumulátor regiszter tartalmával kerül összehasonlításra. Az összehasonlítás eredményeként egy feltétel bit fog 1-es állapotba kerülni. A HL inkrementálódik, és a byte számláló /BC regiszterpár/ tartalma dekrementálódik. Ha a dekrementálás eredményeként a BC tartalma 0-vá válik, vagy ha  $A=(HL)$ , az utasítás befejeződik. Ha BC tartalma nem zérus, és  $A \neq (HL)$ , a programszámláló 2-vel dekrementálódik, és az utasítás megismétlődik. Megjegyezzük, hogy akkor, ha BC tartalma az utasítás végrehajtása előtt már 0-ra van beállítva, az utasítás 64Kbyte-on keresztül fog megismétlődni, feltéve, hogy az egymást követő összehasonlítások során nem történt megegyezés. Minden egyes memóriához fordulást követően két frissítési ciklus zajlik le, és ilyenkor a megszakítások is érvényesülnek.

$BC \neq 0$  és  $A \neq (HL)$  esetén:

M ciklusok: 5    T állapotok: 21/4,4,3,5,5/    4 MHz V.i.: 5.25

$BC=0$  vagy  $A=(HL)$  esetén:

M ciklusok: 4    T állapotok: 16/4,4,3,5/    4 MHz V.i.: 4.00

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha  $A=HL$ ; egyébként törlődik
- H: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha  $BC-1 \neq 0$ ; egyébként törlődik
- N: 1-es állapotba kerül
- C: változatlan marad

# CPIR

---

## Példa:

Ha a HL regiszterpár 1111H-t, az akkumulátor F3H-t, a byte számláló 0007H-t tartalmaz, és a memóriarekeszek tartalma az alábbiak szerint alakul:

|         |   |     |
|---------|---|-----|
| (1111H) | : | 52H |
| (1112H) | : | 00H |
| (1113H) | : | F3H |

akkor a

## CPIR

végrehajtásának eredményeként a HL regiszterpár tartalma 1114H, a byte számláló tartalma 0004H lesz, a P/V bit 1-es állapotba kerül, és a Z bit is 1-es állapotba kerül.

# CPL

18080 CMA

---

Művelet:  $A \leftarrow \bar{A}$

Formátum: OP kód

CPL

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 2F

Leírás:

Az akkumulátor /A regiszter/ tartalma invertálódik /1-es komplementens képződik/.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

A jelzőbit regiszter érintett bitjei:

S: változatlan marad  
Z: változatlan marad  
H: 1-es állapotba kerül  
P/V: változatlan marad  
N: 1-es állapotba kerül  
C: változatlan marad

Példa:

Ha az akkumulátor tartalma 1011 0100, a

CPL

végrehajtásának eredményeként az akkumulátor tartalma 0100-1011 lesz.

# DAA

Művelet: az akkumulátor decimális korrekciója

Formátum: OP kód

DAA

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

27

Leírás:

Az utasítás BCD összeadáskor és kivonáskor szükségessé váló akkumulátor tartalom korrekciót végzi. Összeadás /ADD, ADC, INC/ vagy kivonás /SUB, SBC, DEC, NEG/ esetén az utasítás által megvalósított műveleteket az alábbi táblázat mutatja:

| Művelet | C<br>tartalom<br>DAA<br>előtt | Magasabb<br>helyiértékű<br>SZÁM<br>/7-4 bit/<br>HEXA<br>értéke | H<br>tartalom<br>DAA<br>előtt | Alacsonyabb<br>helyiértékű<br>SZÁM<br>/3-0 bit/<br>HEXA<br>értéke | BYTE-<br>hozzá-<br>adott<br>szám | C<br>tartalom<br>DAA<br>után |
|---------|-------------------------------|--|-------------------------------|---|----------------------------------|------------------------------|
|         | 0                             | 0-9  | 0                             | 0-9   | 00                               | 0                            |
|         | 0                             | 0-8  | 0                             | A-F   | 06                               | 0                            |
|         | 0                             | 0-9  | 1                             | 0-3   | 06                               | 0                            |
| ADD     | 0                             | A-F  | 0                             | 0-9   | 60                               | 1                            |
| ADC     | 0                             | 9-F  | 0                             | A-F   | 66                               | 1                            |
| INC     | 0                             | A-F  | 1                             | 0-3   | 66                               | 1                            |
|         | 1                             | 0-2  | 0                             | 0-9   | 60                               | 1                            |
|         | 1                             | 0-2  | 0                             | A-F   | 66                               | 1                            |
|         | 1                             | 0-3  | 1                             | 0-3   | 66                               | 1                            |
| SUB     | 0                             | 0-9  | 0                             | 0-9   | 00                               | 0                            |
| SBC     | 0                             | 0-8  | 1                             | 6-F   | FA                               | 0                            |
| DEC     | 1                             | 7-F  | 0                             | 0-9   | A0                               | 1                            |
| NEG     | 1                             | 6-F  | 1                             | 6-F   | 9A                               | 1                            |

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az akkumulátor tartalmának legmagasabb helyiértékű bitje 1 a művelet után; egyébként törlődik
- Z: 1-es állapotba kerül, ha a művelet után az akkumulátor tartalma zérus; egyébként törlődik
- L: lásd a táblázatot



# DAA

---

P/V: 1-es állapotba kerül, ha a művelet után az  
akkumulátor páros paritású; egyébként tör-  
lődik  
N: változatlan marad  
C: lásd a táblázatot

## Példa:

Ha a 15 /BCD/ és 27 /BCD/ értékeket kell összeadni, az elemi decimális aritmetika a következő eredményt adja:

$$\begin{array}{r} 15 \\ + 27 \\ \hline 42 \end{array}$$

Ha viszont a megfelelő bináris reprezentációk a normál bináris aritmetika szerint kerülnek összeadásra az akkumulátorban, ekkor

$$\begin{array}{r} 0001 \quad 0101 \\ + 0010 \quad 0111 \\ \hline 0011 \quad 1100 \quad 3C \end{array}$$

az eredmény nem egyértelmű. A DAA utasítás ezt az eredményt úgy korrigálja, hogy a helyes BCD reprezentációt kapjuk:

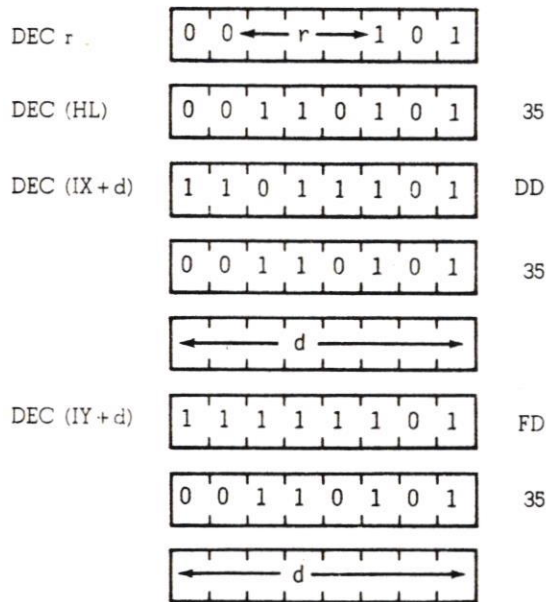
$$\begin{array}{r} 0011 \quad 1100 \\ + 0000 \quad 0110 \\ \hline 0100 \quad 0010 = 42 \end{array}$$

# DEC m

Művelet:  $m \leftarrow m-1$

Formátum:      OP kód              Operandusok  
                               DEC                                    m

Az m operandus r, (HL), (IX+d) vagy (IY+d) bármelyike, amint azt az ezzel az utasítással analóg INC utasításoknál meghatároztuk. Ezek a különböző lehetséges OP kód-operandus-kombinációk a tárgykódban a következők:



r azonosítja a B,C,D,E,H,L vagy A regisztereket, melyek a következők a fenti tárgykódban:

| <u>Regiszter</u> | <u>r</u> |    |
|------------------|----------|----|
| B                | 000      | 05 |
| C                | 001      | 0D |
| D                | 010      | 15 |
| E                | 011      | 1D |
| H                | 100      | 25 |
| L                | 101      | 2D |
| A                | 111      | 3D |

## Leírás:

Az m operandus által specifikált byte dekrementálódik.

| UTASÍTÁS   | M CIKLUSOK | T ÁLLAPOTOK     | 4 MHz V.i. |
|------------|------------|-----------------|------------|
| DEC r      | 1          | 4               | 1.00       |
| DEC (HL)   | 3          | 11/4,4,3/       | 2.75       |
| DEC (IX+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |
| DEC (IY+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha a művelet előtt m 80H volt; egyébként törlődik
- N: 1-es állapotba kerül
- C: változatlan marad

## Példa:

Ha a D regiszterben lévő byte 2AH, a

DEC D

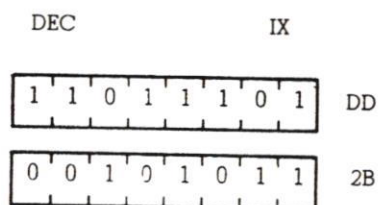
végrehajtásának eredményeként a D regiszter tartalma 29H lesz.

# DEC IX

---

Művelet:  $IX \leftarrow IX-1$

Formátum:      OP kód              Operandumok



Leírás:

Az IX index regiszter tartalma dekrementálódik.

M ciklusok: 2      T állapotok: 10/4,6/      4 MHz V.i.: 2.50

Példa:

Ha az IX index regiszter tartalma 2006H, a

DEC IX

végrehajtásának eredményeként az IX index regiszter tartalma 2005H lesz.

# DEC IY

---

Művelet: IY ← IY-1

Formátum: OP kód            Operandusok

DEC

IY

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 FD

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 2B

Leírás:

Az IY index regiszter tartalma dekrementálódik.

M ciklusok: 2      T állapotok: 10/4,6/      4 MHz V.i.: 2.50

Példa:

Ha az IY index regiszter tartalma 7649H, a

DEC IY

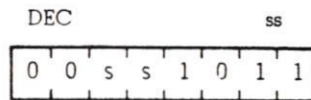
végrehajtásának eredményeként az IY index regiszter tartalma 7648H lesz.

# DEC ss

18080 DCX ss

Művelet: ss ← ss-1

Formátum: OP kód Operandumok



## Leírás:

Az ss regiszterpár /a BC, DE, HL vagy SP regiszterpárok bármelyike/ tartalma dekrementálódik. Az ss operandus tárgykódban a következő:

| <u>Regiszterpár</u> | <u>ss</u> |    |
|---------------------|-----------|----|
| BC                  | 00        | 4B |
| DE                  | 01        | 1B |
| HL                  | 10        | 2B |
| SP                  | 11        | 3B |

M ciklusok: 1      T állapotok: 6      4 MHz V.i.: 1.50

## Példa:

Ha a HL regiszterpár tartalma 1001H, a

DEC HL

végrehajtásának eredményeként a HL tartalma 1000H lesz.

# DI

---

Művelet: IFF ← 0

Formátum: OP kód

DI

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 F3

Leírás:

A megszakítás engedélyező flip-flopok /IFF1 és IFF2/ törlésével a DI tiltja a maszkolható megszakítások érvényre jutását. Megjegyezzük, hogy az utasítás már végrehajtása alatt tiltja a maszkolható megszakításokat.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

Példa:

Amikor a CPU a

DI

utasítást hajtja végre, a maszkolható megszakítás mindaddig letiltásra kerül, amíg azt egy később végrehajtott EI utasítás újra nem engedélyezi. A CPU tehát a normál megszakítás kérésre /Interrupt Request - INT/ ebben az esetben nem reagál.





## DJNZ e

---

```
LD      B,80          ; a számláló előkészítése
LD      HL,INBUF      ; a pointerek előkészítése
LD      DE,OUTBUF     ;

LOOP:   LD      A,(HL) ; a következő byte megszerzése
        ; az input pufferből
LD      (DE),A        ; tárolás az output pufferben
CP      ODH           ; CR volt ?
JR      Z,KÉSZ        ; Igen, vége
INC     HL            ; nem, pointer inkrementálás
INC     DE
DJNZ   LOOP           ; visszaugrás, ha még 80 byte
        ; áttöltése nem történt meg

KÉSZ:
```

# EI

---

Művelet: IFF ← 1

Formátum: OP kód

EI

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 FB

## Leírás:

Az EI a maszkolható megszakításokat engedélyezi, azáltal, hogy a megszakítás engedélyező flip-flopokat /IFF1 és IFF2/ 1-es állapotba állítja. Megjegyezzük, hogy az utasítás végrehajtása során a maszkolható megszakításokat tiltja.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

## Példa:

Ha a CPU befejezte az

EI

utasítás végrehajtását, a maszkolható megszakítás engedélyezett, és a CPU képes a megszakítás kérelem /Interrupt Request - INT/ elfogadására.

# EX (SP),HL

18080 XTHL

Művelet: H  $\leftrightarrow$  (SP+1) , L  $\leftrightarrow$  (SP)

Formátum: OP kód Operandusok

EX (SP),HL

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 E3

## Leírás:

A HL regiszterpárban lévő kisebb helyiértékű byte az SP regiszterpár tartalma által meghatározott memóriacímen lévő byte tartalmával cserélődik ki, a HL regiszter magasabb helyiértékű byte-ja pedig a következő memóriacímen /SP+1/ lévő byte-tal cserélődik ki.

M ciklusok: 5 T állapotok: 19/4,3,4,3,5/ 4 MHz V.i.: 4.75

## Példa:

Ha a HL regiszterpár 7012H-t, az SP regiszterpár 8856H-t tartalmaz, a 8856H című memóriarekeszben lévő byte 11H és a 8857H című memóriarekeszben lévő byte 22H, akkor az

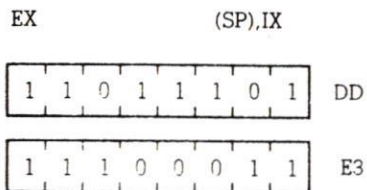
EX (SP), HL

utasítás végrehajtásának eredményeként a HL regiszterpár tartalma 2211H lesz, a 8856H című memóriarekeszben lévő byte 12H, a 8857H című memóriarekeszben lévő byte 70H lesz, és az SP regiszterpár 8856H-t fog tartalmazni.

# EX (SP),IX

Művelet:  $IX_H \leftrightarrow (SP+1), IX_L \leftrightarrow (SP)$

Formátum: OP kód            Operandusok



## Leírás:

Az IX index regiszterben lévő kisebb helyiértékű byte az SP regiszterpár tartalma által meghatározott memóriacímen lévő byte tartalmával cserélődik ki, az IX magasabb helyiértékű byte-ja pedig a következő memóriacímen /SP+1/ lévő byte-tal cserélődik ki.

M ciklusok: 6    T állapotok: 23/4,4,3,4,3,5/    4 MHz V.i.: 5.75

## Példa:

Ha az IX index regiszter 3988H-t, az SP regiszterpár 0100H-t tartalmaz, a 0100H című memóriarekeszben lévő byte 90H és a 0101H című memóriarekeszben lévő byte 48H, akkor ez

EX (SP),IX

utasítás végrehajtásának eredményeként az IX regiszterpár tartalma számértékben 4890H lesz, a 0100H című memóriarekeszben lévő byte 88H, a 0101H című memóriarekeszben lévő byte 39H lesz, és az SP regiszter 0100H-t fog tartalmazni.

## EX (SP),IY

Művelet:  $IY_H \leftrightarrow (SP+1)$ ,  $IY_L \leftrightarrow (SP)$

Formátum: OP kód Operandusok

EX (SP),IY

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 FD

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 E3

### Leírás:

Az IY index regiszterben lévő kisebb helyiértékű byte az SP regiszterpár tartalma által meghatározott memóriacímen lévő byte tartalmával cserélődik ki, az IY magasabb helyiértékű byte-ja pedig a következő memóriacímen /SP+1/ lévő byte-tal cserélődik ki.

M ciklusok: 6 T állapotok: 23/4,4,3,4,3,5/ 4 MHz V.i.: 5.75

### Példa:

Ha az IY index regiszter 3988H-t, az SP regiszterpár 0100H-t tartalmaz, a 0100H című memóriarekeszben lévő byte 90H és a 0101H című memóriarekeszben lévő byte 48H, akkor az

EX (SP), IY

utasítás végrehajtásának eredményeként az IY regiszterpár tartalma számértékben 4890H lesz, a 0100H című memóriarekeszben lévő byte 88H, a 0101H című memóriarekeszben lévő byte 39H lesz, és az SP regiszter 0100H-t fog tartalmazni.

# EX AF,AF'

Művelet: AF ↔ AF'

Formátum: OP kód            Operandusok

EX                            AF,AF'

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|

 08

Leírás:

Az AF és AF' 2 byte-os regiszterpárok tartalma felcserélődik.  
/Megjegyzés: az AF' regiszterpár az A' és F' regiszterekből áll/.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

Példa:

Ha az AF regiszterpár tartalma számértékben 9900H és az AF' regiszterpár tartalma számértékben 5944H, az

EX AF,AF'

utasítás végrehajtásának eredményeként az AF tartalma 5944H, és az AF' tartalma 9900H lesz.

# EX DE,HL

18080 XCHG

Művelet: DE ↔ HL

Formátum: OP kód Operandusok

EX DE,HL

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 EB

Leírás:

A DE és HL 2 byte-os regiszterpárak tartalma felcserélődik.

M ciklusok: 1 T állapotok: 4 4 MHz V.i.: 1.00

Példa:

Ha a DE regiszterpár tartalma számértékben 2822H és a HL regiszterpár tartalma számértékben 499AH, akkor az

EX DE,HL

utasítás végrehajtásának eredményeként a DE regiszterpár tartalma 499AH és a HL regiszterpár tartalma 2822H lesz.

# EXX

---

Művelet: (BC) ↔ (BC'), (DE) ↔ (DE'), (HL) ↔ (HL')

Formátum: OP kód

EXX

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 D9

Leírás:

A BC, DE és HL regiszterpárokban lévő minden egyes 2 byte-os érték rendre felcserélődik a BC', DE' és HL' regiszterpárokban lévő 2 byte-os értékkel.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

Példa:

Ha a BC, DE és HL regiszterpárok tartalma számértékben 445AH, 3DA2H és 8859H, és a BC', DE' és HL' regiszterpárok tartalma 0988H, 9300H és 00E7H, az

EXX

utasítás végrehajtásának eredményeként a regiszterpárok tartalma a következőképpen alakul: BC: 0988H; DE: 9300H; HL: 00E7H; BC': 445AH; DE': 3DA2H; és HL': 8859H.



# HALT

---

Művelet: CPU Halt

Formátum: OP kód

HALT

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 76

Leírás:

A HALT utasítás a következő megszakítás vagy törlési utasítás detektálásáig felfüggeszti a CPU működését. A „halt” állapotban a processzor a memória frissítő logika működtetéséhez NOP utasításokat hajt végre.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

# IM 0

---

Művelet: Interrupt Mode 0 beállítása

Formátum: OP kód            Operandusok

|    |   |    |
|----|---|----|
| IM | 0 |    |
| 1  | 1 | ED |
| 1  | 0 |    |
| 1  | 1 |    |
| 0  | 1 |    |
| 1  | 0 |    |
| 1  | 0 |    |
| 0  | 1 | 46 |
| 1  | 1 |    |
| 0  | 0 |    |

Leírás:

Az IM 0 utasítás a 0-ás megszakítási üzemmódot /mode 0/ állítja be. Ebben az üzemmódban a megszakítást kérő eszköz a megszakítás elfogadását követően egy tetszőleges utasítást helyezhet az adatbuszra, hogy azt a CPU végrehajtsa.

M ciklusok: 2      T állapotok: 8/4,4/      4 MHz V.i.: 2.00

# IM 1

---

Művelet: Interrupt Mode 1 beállítása

Formátum: OP kód            Operandusok

|   |   |   |   |   |   |   |   |   |    |  |
|---|---|---|---|---|---|---|---|---|----|--|
| IM  | 1 |   |   |   |   |   |   |   |    |  |
| <table border="1"><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td></tr></table> | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | ED |  |
| 1   | 1 | 1 | 0 | 1 | 1 | 0 | 1 |   |    |  |
| <table border="1"><tr><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td></tr></table> | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 56 |  |
| 0   | 1 | 0 | 1 | 0 | 1 | 1 | 0 |   |    |  |

Leírás:

Az IM utasítás az 1-es megszakítási üzemmódot /mode 1/ állítja be. Ebben az üzemmódban a processzor a megszakítási kérélemre egy 0038H-as rekeszre irányuló „Restart” végrehajtásával válaszol.

M ciklusok: 2      T állapotok: 8/4,4/      4 MHz V.i.: 2.00

## IM 2

---

Művelet: Interrupt Mode 2 beállítása

Formátum: OP kód            Operandusok

IM                            2

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 SE

### Leírás:

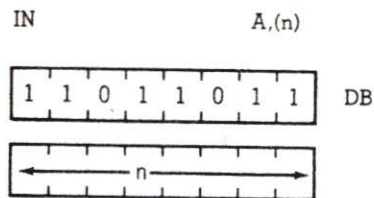
Az IM 2 utasítás a 2-es megszakítási üzemmódot /mode 2/ állítja be. Ez az üzemmód egy tetszőleges memóriarekeszre irányuló „Indirekt Call” végrehajtását teszi lehetővé. Ebben az üzemmódban a CPU egy 16 bites memóriacímet állít össze. A cím felső 8 bitjét az I „Interrupt Vector Register” tartalmazza, míg az alacsonyabb helyiértékű 8 bitet a megszakítás elfogadásakor a megszakítást kérő eszköz helyezi az adatbuszra.

M ciklusok: 2            T állapotok: 8/4,4/            4 MHz V.i.: 2.00

# IN A,(n)

Művelet:  $A \leftarrow (n)$

Formátum: OP kód          Operandusok



## Leírás:

A végrehajtás során az utasításnak a címbusz alsó felére /A0-tól A7-ig/ helyezett n operandusa a 256 lehetséges periféria regiszter /I/O port/ egyikét választja ki. Ugyanakkor az akkumulátor regiszter tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. A kiválasztott periféria regiszter tartalma az adatbuszra kerül, és a CPU akkumulátor regiszterébe /A regiszter/ íródik.

M ciklusok: 3    T állapotok: 11/4,3,4/    4 MHz V.i.: 2.75

## Példa:

Ha az akkumulátor tartalma 23H, és a 01H című periféria regiszterben 7BH van jelen, akkor az

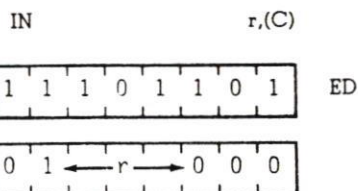
IN A,(01H)

utasítás végrehajtása után az akkumulátor 7BH-t fog tartalmazni.

# IN r,(C)

Művelet:  $r \leftarrow (C)$

Formátum: OP kód                      Operandusok



Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, amelynél a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során a C regiszter tartalma a címbusz alsó 8 bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanakkor a B regiszter tartalma a címbusz felső 8 bitjén /A8-tól A15-ig/ jelenik meg. A C regiszter tartalma által megcímzett periféria regiszterben lévő byte az adatbuszra kerül, majd a CPU-nak az utasítás r mezője által meghatározott regiszterébe íródik. Az r bármelyik CPU regisztert kijelölheti. Az egyes regiszterekhez hozzárendelt 3 bites r mezőket az alábbi táblázat mutatja. Az állapot bitek a beolvasott adat tartalmának megfelelően állnak be:

| <u>Regiszter</u> | <u>r</u> |  |
|------------------|----------|--|
| B                | 000      | 40H                                    |
| C                | 001      | 48H                                    |
| D                | 010      | 50H                                    |
| E                | 011      | 58H                                    |
| H                | 100      | 60H                                    |
| L                | 101      | 68H                                    |
| A                | 111      | 78H                                    |
|                  | 110      | 70H az a flagokba van kódszóval azadit |

M ciklusok: 3    T állapotok: 12/4,4,4/    4 MHz V.i.: 3.00

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha a beolvasott adat negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha a beolvasott adat nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül, páros paritás esetén; egyébként törlődik
- N: törlődik
- C: változatlan marad

Példa:

Ha a C regiszter tartalma 07H, a B regiszter tartalma 10H, és a 07H című periféria regiszterben 7BH van jelen, akkor az

IN D,(C)

végrehajtásának eredményeként a D regiszter tartalma 7BH lesz, míg a B és C regiszterek tartalma változatlan marad.

# INC (HL)

1980 INC M

Művelet: (HL) ← (HL) + 1

Formátum:        OP kód            Operandusok  
                  INC                    (HL)

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
|---|---|---|---|---|---|---|---|

 34

## Leírás:

A HL regiszter tartalma által megcímezett memóriarekesz tartalma inkrementálódik.

M ciklusok: 3    T állapotok: 11/4,4,3/    4 MHz V.i.: 2.75

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha a működés előtt HL 7FH volt; egyébként törlődik
- N: törlődik
- C: változatlan marad

## Példa:

Ha a HL regiszterpár tartalma 3434H, és a 3434H című memóriarekeszben lévő byte 82H, akkor az

INC (HL)

végrehajtásának eredményeként a 3434H című memóriarekesz 83H-t fog tartalmazni.

# INC IX

---

Művelet: IX ← IX+1

Formátum:      OP kód            Operandusok

INC

IX

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 DD

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 23

## Leírás:

Az IX index regiszter tartalma inkrementálódik.

M ciklusok: 2      T állapotok: 10/4,6/      4 MHz V.i.: 2.50

## Példa:

Ha az IX index regiszter a 3300H egész számot tartalmazza, az

INC IX

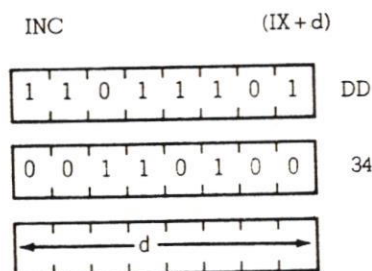
utasítás végrehajtásának eredményeként az IX index regiszter tartalma 3301H lesz.



## INC (IX+d)

Művelet:  $(IX+d) \leftarrow (IX+d) + 1$

Formátum: OP kód            Operandusok



### Leírás:

Az IX index regiszter /IX regiszterpár/ tartalmához hozzáadódik az utasítás d mezőjében szereplő 2-es komplementű egész számként ábrázolt címkiegészítés. Az így kapott memóriacím által meghatározott rekesz tartalma inkrementálódik.

M ciklusok: 6    T állapotok: 23/4,4,3,5,4,3/    4 MHz V.i.: 5.75

### A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha a működés előtt (IX+d) 7FH volt; egyébként törlődik
- N: törlődik
- C: változatlan marad

### Példa:

Ha az IX index regiszter tartalma 2020H, és a 2030 című memóriarekeszben lévő byte 34H, az

INC (IX+10H)

utasítás végrehajtásának eredményeként a 2030 című memóriarekesz tartalma 35H lesz.

# INC IY

Művelet:  $IY \leftarrow IY+1$

Formátum:      OP kód            Operandusok

INC

IY

|   |   |   |   |   |   |   |   |    |
|---|---|---|---|---|---|---|---|----|
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | FD |
|---|---|---|---|---|---|---|---|----|

|   |   |   |   |   |   |   |   |    |
|---|---|---|---|---|---|---|---|----|
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 23 |
|---|---|---|---|---|---|---|---|----|

Leírás:

Az IY index regiszter tartalma inkrementálódik.

M ciklusok: 2    T állapotok: 10/4,6/ 4 MHz V.i.: 2.50

Példa:

Ha az index regiszter tartalma 2977H, akkor az

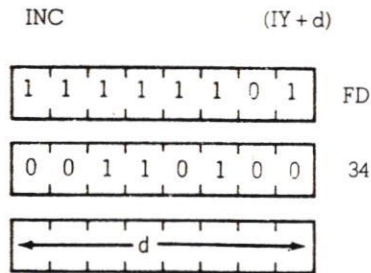
INC IY

utasítás végrehajtásának eredményeként az IY index regiszter tartalma 2978H lesz.

# INC (IY+d)

Művelet: (IY+d) (IY+d)+1

Formátum: OP kód Operandusok



## Leírás:

Az IY index regiszter /IY regiszterpár/ tartalmához hozzáadódik az utasítás d mezőjében szereplő 2-es komplementű egész számként ábrázolt címkiegészítés. Az így kapott memóriacím által meghatározott rekesz tartalma inkrementálódik.

M ciklusok:6 T állapotok: 23/4,4,3,5,4,3/ 4 MHz V.i.: 5.75

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha a működés előtt (IY+d) 7FH volt; egyébként törlődik
- N: törlődik
- C: változatlan marad

## Példa:

Ha az IY index regiszterpár tartalma 2020H, és a 2030H című memóriarekeszben lévő byte 34H, akkor az

INC (IY+10H)

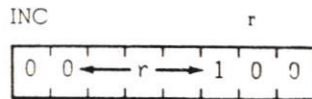
utasítás végrehajtásának eredményeként a 2030H című memóriarekesz tartalma 35H lesz.

# INC r

18080 INR r

Művelet:  $r \leftarrow r+1$

Formátum: OP kód Operandumok



## Leírás:

Az r regiszter inkrementálódik, r azonosítja az A,B,C,D,E,H vagy L regiszterek bármelyikét, amely tárgykódban a következő lehet:

| Regiszter | r   |    |
|-----------|-----|----|
| A         | 111 | 3C |
| B         | 000 | 04 |
| C         | 001 | 0C |
| D         | 010 | 14 |
| E         | 011 | 1C |
| H         | 100 | 24 |
| L         | 101 | 2C |

M ciklusok: 1 T állapotok: 4 4 MHz V.i.: 1.00

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha a 3-as biten átvitel keletkezik; egyébként törlődik
- P/V: 1-es állapotba kerül, ha a működés előtt r 7FH volt; egyébként törlődik
- N: törlődik
- C: változatlan marad

## Példa:

Ha a D regiszter tartalma 28H, az

INC D

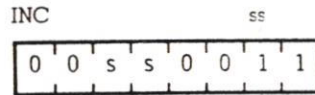
utasítás végrehajtásának eredményeként a D regiszter tartalma 29H lesz.

# INC ss

18080 INX

Művelet:  $ss \leftarrow ss+1$

Formátum: OP kód Operandumok



## Leírás:

Az ss regiszterpár /a BC, DE, HL vagy SP regiszterpárok bármelyike/ tartalma inkrementálódik. Az ss operandus tárgykódban a következő lehet:

| <u>Regiszterpár</u> | <u>ss</u> |    |
|---------------------|-----------|----|
| BC                  | 00        | 03 |
| DE                  | 01        | 13 |
| HL                  | 10        | 23 |
| SP                  | 11        | 33 |

M ciklusok: 1      T állapotok: 6      4 MHz V.i.: 1.50

## Példa:

Ha a HL regiszterpár 1000H-t tartalmaz, az

INC HL

utasítás végrehajtásának eredményeként a HL 1001H-t fog tartalmazni.

# IND

Művelet: (HL) ← (C), B ← B-1, HL ← HL-1

Formátum: OP kód

IND

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 AA

## Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, amelynél a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során a C regiszter tartalma a címbusz alsó 8 bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanekkor a byte számlálóként felhasználható B regiszter tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. A megcímzett periféria regiszter tartalma az adatbuszra kerül, majd a CPU-ba íródik. Ugyanekkor a HL regiszterpár tartalma a címbuszra kerül, és a periféria regiszterből beolvasott byte az ilyen módon megcímzett memóriarekeszbe íródik. Végül a byte számláló, és a HL regiszterpár dekrementálódik.

M ciklusok: 4 T állapotok: 16/4,5,3,4/ 4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

S: ismeretlen  
Z: 1-es állapotba kerül, ha B-1=0; egyébként törlődik  
H: ismeretlen  
P/V: ismeretlen  
N: 1-es állapotba kerül  
C: változatlan marad

## Példa:

Ha a C regiszter tartalma 07H, a B regiszter tartalma 10H, a HL regiszterpár tartalma 1000H, és a 07H című periféria regiszterben 7BH van jelen, akkor az

IND

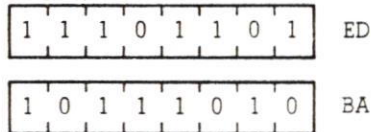
utasítás végrehajtásának eredményeként az 1000H című memóriarekesz 7BH-t, a HL regiszterpár OFFFH-t, és a B regiszter 0FH-t fog tartalmazni.

# INDR

Művelet: (HL) ← (C), B ← B-1, HL ← HL-1

Formátum: OP kód

INDR



## Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, amelynek a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során a C regiszter tartalma a címbusz alsó 8 bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanekkor a byte számlálóként felhasználható B regiszter tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. A megcímzett periféria regiszter tartalma az adatbuszra kerül, majd a CPU-ba íródik. Ugyanekkor a HL regiszterpár tartalma a címbuszra kerül, és a periféria regiszterből beolvasott byte az ilyen módon megcímzett memóriarekeszbe íródik. Végül a byte számláló, és a HL regiszterpár dekrementálódik. Ha a B /byte számláló/ dekrementálása zérus eredményez, az utasítás befejeződik. Ha a dekrementálás eredménye nem zérus, akkor a utasítás végrehajtása ismétlődik. Megjegyezzük, hogy ha az utasítás végrehajtását megelőzően B tartalma 0-ra lett beállítva, 256 byte adat olvasódik be. Minden adatbeolvasást követően két memória frissítési ciklus zajlik le, és ugyanekkor programmegszakítási kérelmek is érvényre juthatnak.

Ha B ≠ 0:

M ciklusok: 5 T állapotok: 21/4,5,3,4,5/ 4 MHz V.i.: 5.25

Ha B = 0:

M ciklusok: 4 T állapotok: 16/4,5,3,4/ 4 MHz V.i.: 4.00

A jelzőbit regiszter érintett bitjei:

S: ismeretlen  
Z: 1-es állapotba kerül  
H: ismeretlen

# INDR

---

P/V: ismeretlen  
N: 1-es állapotba kerül  
C: változatlan marad

## Példa:

Ha a C regiszter tartalma 07H, a B regiszter tartalma 03H, a HL regiszterpár tartalma 1000H, és a következő byte-sorozat áll rendelkezésre a 07H című periféria regiszterben:

51H  
A9H  
03H

akkor az

INDR

utasítás végrehajtásának eredményeként a HL regiszterpár OFFDH-t, a B regiszter zérust fog tartalmazni, és a memóriarekeszek tartalma a következőképpen alakul:

| <u>Rekesz</u> | <u>Tartalom</u> |
|---------------|-----------------|
| OFFEH         | 03H             |
| OFFFH         | A9H             |
| 1000H         | 51H             |



# INI

Művelet: (HL) ← (C), B ← B-1, HL ← HL+1

Formátum: OP kód

INI

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 A2

## Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, amelynél a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során a C regiszter tartalma a címbusz alsó 8 bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanekkor a byte számlálóként felhasználható B regiszter tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. A megcímzett periféria regiszter tartalma az adatbuszra kerül, majd a CPU-ba íródik. Ugyanekkor a HL regiszterpár tartalma a címbuszra kerül, és a periféria regiszterből beolvasott byte az ilyen módon megcímzett memóriarekeszbe íródik. Végül a byte számláló dekrementálódik, és a HL regiszterpár inkrementálódik.

M ciklusok: 4 T állapotok: 16/4,5,3,4/ 4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

- S: ismeretlen
- Z: 1-es állapotba kerül, ha B-1=0; egyébként törlődik
- H: ismeretlen
- P/V: ismeretlen
- N: 1-es állapotba kerül
- C: változatlan marad

## Példa:

Ha a C regiszter tartalma 07H, a B regiszter tartalma 10H, a HL regiszterpár tartalma 1000H, és a 07H című periféria regiszterben 7BH van jelen, akkor az

INI

utasítás végrehajtásának eredményeként az 1000H című memóriarekesz 7BH-t, a HL regiszterpár 1001H-t, és a B regiszter 0FH-t fog tartalmazni.

# INIR

Művelet: (HL) ← (C), B ← B-1, HL ← HL+1

Formátum: OP kód

INIR

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 B2

Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, emelynél a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során a C regiszter tartalma a címbusz alsó 8 bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanekkor a byte számlálóként felhasználható B regiszter tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. A megcímzett periféria regiszter tartalma az adatbuszra kerül, majd a CPU-ba íródik. Ugyanekkor a HL regiszterpár tartalma a címbuszra kerül, és a periféria regiszterből beolvasott byte az ilyen módon megcímzett memóriarekeszbe íródik. Azután a HL regiszterpár inkrementálódik, a byte számláló pedig dekrementálódik. Ha a B /byte számláló/ dekrementálása zérust eredményez, az utasítás befejeződik. Ha a dekrementálás eredménye nem zérus, akkor a programszámláló tartalma 2-vel dekrementálódik, és az utasítás végrehajtása ismétlődik. Megjegyezzük, hogy ha az utasítás végrehajtását megelőzően B tartalma 0-ra lett beállítva, 256 byte adat olvasódik be. Minden adatbeolvasást követően két memória frissítési ciklus zajlik le, és ugyanekkor programmegszakítási kérelmek is érvényre juthatnak.

Ha B≠0:

M ciklusok: 5 T állapotok: 21/4,5,3,4,5/ 4 MHz V.i.: 5.25

Ha B=0:

M ciklusok: 4 T állapotok: 16/4,5,3,4/ 4 MHz V.i.: 4.00

A jelzőbit regiszter érintett bitjei:

S: ismeretlen  
Z: 1-es állapotba kerül  
H: ismeretlen  
P/V: ismeretlen  
N: 1-es állapotba kerül  
C: változatlan marad

# INIR

---

Példa:

Ha a C regiszter tartalma 07H, a B regiszter tartalma 03H, a HL regiszterpár tartalma 1000H, és a következő byte-sorozat áll rendelkezésre a 07H című periféria regiszterben:

51H  
A9H  
03H

akkor az

INIR

utasítás végrehajtásának eredményeként a HL regiszterpár 1003H-t, a B regiszter zérust fog tartalmazni, és a memóriarekeszek tartalma a következőképpen alakul:

| <u>Rekesz</u> | <u>Tartalom</u> |
|---------------|-----------------|
| 1000H         | 51H             |
| 1001H         | A9H             |
| 1002H         | 03H             |

# JP (HL)

18080 PCHL

Művelet: PC ← HL

Formátum: OP kód Operandusok

|                 |      |
|-----------------|------|
| JP              | (HL) |
| 1 1 1 0 1 0 0 1 | E9   |

## Leírás:

A programszámláló regiszter /PC regiszterpár/ a HL regiszterpár tartalmával töltődik fel. A soronkövetkező utasítás a programszámláló új tartalma által meghatározott memóriarekeszből lesz kiolvasva.

M ciklusok: 1 T állapotok: 4 4 MHz V.i.: 1.00

## Példa:

Ha a programszámláló tartalma 1000H, és a HL regiszterpár tartalma 4800H, akkor a

JP (HL)

utasítás végrehajtásának eredményeként a programszámláló tartalma 4800H lesz.

# JP (IX)

Művelet: PC ← IX

Formátum: OP kód            Operandumok

JP

(IX)

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 DD

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 E9

## Leírás:

A programszámláló /PC regiszterpár/ az IX regiszterpár /IX index regiszter/ tartalmával töltődik fel. A soronkövetkező utasítás a programszámláló új tartalma által meghatározott memóriarekeszekből lesz kiolvasva.

M ciklusok: 2      T állapotok: 8/4,4/      4 MHz V.i.: 2.00

## Példa:

Ha a programszámláló tartalma 1000H, és az IX regiszterpár tartalma 4800H, akkor a

JP (IX)

utasítás végrehajtásának eredményeként a programszámláló tartalma 4800H lesz.

# JP (IY)

---

Művelet: PC ← IY

Formátum: OP kód            Operandumok

JP                            (IY)

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 FD

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 E9

## Leírás:

A programszámláló /PC regiszterpár/ az IY regiszterpár /IY index regiszter/ tartalmával töltődik fel. A soronkövetkező utasítás a programszámláló új tartalma által meghatározott memóriarekeszből lesz kiolvasva.

M ciklusok: 2    T állapotok: 8/4,4/    4 MHz V.i.: 2.00

## Példa:

Ha a programszámláló tartalma 1000H, és az IY regiszterpár tartalma 4800H, akkor a

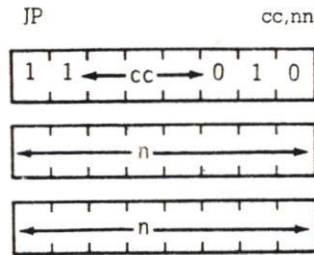
JP (IY)

utasítás végrehajtásának eredményeként a programszámláló tartalma 4800H lesz.

# JP cc,nn

Művelet: Ha cc igaz, PC ← nn

Formátum: OP kód            Operandumok



Megjegyzés: A fenti assemblerrel lefordított tárgykódban elsőként szereplő n operandus egy 2 byte-os memóriacím alacsonyabb helyiértékű byte-ja.

## Leírás:

Ha a cc feltétel teljesül, az utasítás az nn operandust a PC /programszámláló/ regiszterbe tölti, és a program az nn címmel kezdődő utasítás végrehajtásával folytatódik. Ha a cc feltétel nem teljesül, akkor a programszámláló a normál módon inkrementálódik, és a program a soronkövetkező utasítás végrehajtásával folytatódik. A cc feltétel programozása a jelzőbit regiszter /F regiszter/ valamelyik bitjének kijelölésével történhet. Az ilyen módon kijelölhető 8 feltételt tárgykódban az alábbi táblázat definiálja:

| cc  | Feltétel            | Aktuális bit |               |
|-----|---------------------|--------------|---------------|
| 000 | NZ nem zérus        | Z            | JNZ <i>na</i> |
| 001 | Z zérus             | Z            | JZ <i>na</i>  |
| 010 | NC nincs átvitel    | C            | JNC <i>na</i> |
| 011 | C átvitel           | C            | JC <i>na</i>  |
| 100 | PO páratlan paritás | P/V          | JPO <i>na</i> |
| 101 | PE páros paritás    | P/V          | JPE <i>na</i> |
| 110 | P pozitív előjel    | S            | JP <i>na</i>  |
| 111 | M negatív előjel    | S            | JM <i>na</i>  |

M ciklusok: 3    T állapotok: 10/4,3,3/    4 MHz V.i.: 2.50

## Példa:

Ha az átvitel bit /a C bit a jelzőbit regiszterben/ 1-es állapotban van, és a 1520H című memóriarekesz tartalma 03H, akkor a

JP C,1520H

végrehajtásának eredményeként a programszámláló 1520H-t fog tartalmazni, és a következő gépi ciklusban a CPU az 1520H-s címen elhelyezett 03H-s byte-ot fogja kiemelni.

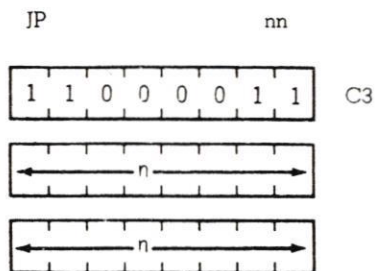
# JP nn

18080 JMP nn

---

Művelet: PC ← nn

Formátum: OP kód            Operandusok



Megjegyzés: A fenti assemblerrel lefordított tárgycodeban elsőként szereplő n operandus egy 2 byte-os memóriacím alacsonyabb helyiértékű byte-ja.

Leírás:

Feltétel nélküli vezérlés átadó utasítás. Végrehajtása során az nn operandus a programszámláló regiszterbe /PC/ töltődik, és így kijelöli azt a memóriarekeszt, ahonnan a CPU a következő utasítást kiemeli.

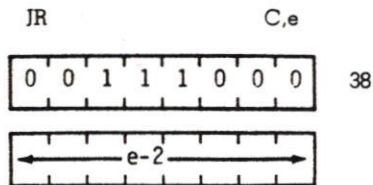
M ciklusok: 3    T állapotok: 10/4,3,3/    4 MHz V.i.: 2.50



## JR C,e

Művelet: Ha C=0, a PC a normál módon inkrementálódik  
Ha C=1,  $PC \leftarrow PC+e$

Formátum: OP kód            Operandusok



### Leírás:

Az utasítás az átvitel bit vizsgálatának eredményétől függően egy más programszegmensre történő relatív címzésű feltételes vezérlésátadást tesz lehetővé. Ha a vizsgálat során az „átvitel bit” tartalma 1, akkor az e 2-es komplementű egész számként értelmezett címkiegészítés értéke a programszámláló /PC/ tartalmához adódik, és a következő utasítást az így kapott címről fogja a CPU kiemelni. Az ugrás hossza az utasítás OP kódját tartalmazó memóriarekesz címétől értendő, és tartománya -126...+129 byte lehet. Az assembler kiszámolja az „e” címkiegészítés értékét, majd az utasítás végrehajtása során automatikusan 2-vel növelt PC tartalom miatt szükségessé váló korrekciót /2-vel való csökkentését/ is végrehajtja. /A PC tartalomnak ez a 2-vel történő növelése az utasítás 2 byte-os hosszának következménye/. Ha az átvitel bit 0-t tartalmaz, a következő utasítást a normál módon soron következő memóriarekeszből emeli ki a CPU.

### Ha a feltétel teljesül:

M ciklusok: 3    T állapotok: 12/4,3,5/    4 MHz V.i.: 3.00

### Ha a feltétel nem teljesül:

M ciklusok: 2    T állapotok: 7/4,3/    4 MHz V.i.: 1.75

### Példa:

Az átvitel bit 1-es állapotban van, és a 480H memóriacím-től 4 rekesznyi-t kell visszaugrani. Az assembler nyelvű utasítás:

JR C, CIMKE

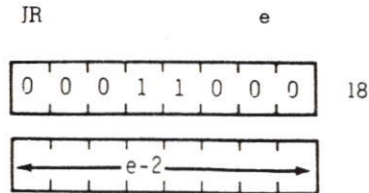
Az eredményként kapott tárgykód, és a PC végső értéke az alábbi táblázaton látható:

| Rekesz |               | OP kód                           |
|--------|---------------|----------------------------------|
| 47C    | CIMKE: -      | PC az ugrás után                 |
| 47D    | -             | -                                |
| 47E    | -             | -                                |
| 47F    | -             | -                                |
| 480    | JR    C,CIMKE | 38                               |
| 481    |               | FA /-6, 2-es komplement kód-ban/ |

# JR e

Művelet: PC ← PC+e

Formátum: OP kód Operandumok



## Leírás:

Az utasítás egy más programszegmensre történő, relatív címzésű, feltétel nélküli vezérlésátadást biztosít. Az e 2-es komponensű egész számként értelmezett címkiegészítés értéke a programszámláló /PC/ tartalmához adódik, és a következő utasítást az így kapott címről fogja a CPU kiemelni. Az ugrás hossza az utasítás OP kódját tartalmazó memóriarekesz címétől értendő, és tartománya -126...+129 byte lehet. Az assembler kiszámolja az „e” címkiegészítés értékét, majd az utasítás végrehajtása során automatikusan 2-vel növelt PC tartalom miatt szükségessé váló korrekciót /2-vel való csökkentést/ is végrehajtja. /A PC tartalomnak ez a 2-vel történő növelése az utasítás 2 byte-os hosszának következménye/.

M ciklusok: 3    T állapotok: 12/4,3,5/    4 MHz V.i.: 3.00

## Példa:

A 480-as címtől 5 rekeszre előre ugráskor a következő assembler nyelvű utasítást használjuk:

JR CIMKE

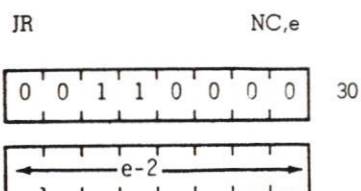
Az eredményként kapott tárgykód, és a PC végső értéke az alábbi táblázaton látható:

| <u>Rekesz</u> |        |       | <u>OP kód</u>    |
|---------------|--------|-------|------------------|
| 480           | JR     | CIMKE | 18               |
| 481           | -      |       | 03               |
| 482           | -      |       |                  |
| 483           | -      |       |                  |
| 484           | -      |       |                  |
| 485           | CIMKE: | -     | PC az ugrás után |

# JR NC,e

Művelet: Ha C=1, a PC a normál módon inkrementálódik  
Ha C=0, PC ← PC+e

Formátum: OP kód Operandumok



## Leírás:

Az utasítás az átvitel bit vizsgálatának eredményétől függően egy más programszegmensre történő relatív címzéssel feltételes vezérlésátadást tesz lehetővé. Ha a vizsgálat során az „átvitel bit” tartalma 0, akkor az e 2-es komplementű egész számként értelmezett címkiegészítés értéke a programszámláló /PC/ tartalmához adódik, és a következő utasítást az így kapott címről fogja a CPU kiemelni. Az ugrás hossza az utasítás OP kódját tartalmazó memóriarekesz címétől értendő, és tartománya -126...+129 byte lehet. Az assembler kiszámolja az „e” címkiegészítés értékét, majd az utasítás végrehajtása során automatikusan 2-vel növelt PC tartalom miatt szükségessé váló korrekciót /2-vel való csökkentést/ is végrehajtja. /A PC tartalomnak ez a 2-vel történő növelése az utasítás 2 byte-os hosszának következménye/.

Ha az átvitel bit 1-et tartalmaz, a következő utasítást a normál módon soron következő memóriarekeszből emeli ki a CPU.

Ha a feltétel teljesül:

M ciklusok: 3    T állapotok: 12/4,3,5/    4 MHz V.i.: 3.00

Ha a feltétel nem teljesül:

M ciklusok: 7    T állapotok: 7/4,3/    4 MHz V.i.: 1.75

## Példa:

Az átvitel bit törölt, és az ugró utasítást meg kell ismételni. Az assembler nyelvű utasítás:

CIMKE: JR NC,CIMKE

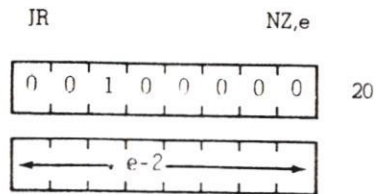
Az eredményként kapott tárgykód, és a PC végső értéke az alábbi táblázaton látható:

| <u>Rekesz</u> |                    | <u>OP kód</u>         |
|---------------|--------------------|-----------------------|
| 480           | CIMKE: JR NC,CIMKE | 30 - PC az ugrás után |
| 481           |                    | 00                    |

# JR NZ,e

Művelet: Ha Z=1, a PC a normál módon inkrementálódik  
 Ha Z=0, PC ← PC+e

Formátum: OP kód Operandusok



## Leírás:

Az utasítás a Z zérus bit vizsgálatának eredményétől függően egy más programszegmensre történő relatív címzésű feltételes vezérlésátadást tesz lehetővé. Ha a vizsgálat során a bit tartalma 0, akkor az e 2-es komplementű egész számként értelmezett címkiegészítés értéke a programszámláló /PC/ tartalmához adódik, és a következő utasítást az így kapott címről fogja a CPU kiemelni. Az ugrás hossza az utasítás OP kódját tartalmazó memóriarekesz címétől értendő, és tartománya -126...+129 byte lehet. Az assembler kiszámolja az „e” címkiegészítés értékét, majd az utasítás végrehajtása során automatikusan 2-vel növelt PC tartalom miatt szükségessé váló korrekciót /2-vel történő csökkentést/ is végrehajtja. /A PC tartalomnak ez a 2-vel való növelése az utasítás 2 byte-os hosszának következménye/. Ha a Z bit 1-et tartalmaz, a következő utasítást a normál módon soron következő memóriarekeszből emeli ki a CPU.

## Ha a feltétel teljesül:

M ciklusok: 3 T állapotok: 12/4,3,5/ 4 MHz V.i.: 3.00

## Ha a feltétel nem teljesül:

M ciklusok: 2 T állapotok: 7/4,3/ 4 MHz V.i.: 1.75

## Példa:

A Z zérus bit törölt, és a 480H memóriacímtől 4 rekesznyiit kell visszaugrani. Az assembler nyelvű utasítás:

JR NZ,CIMKE

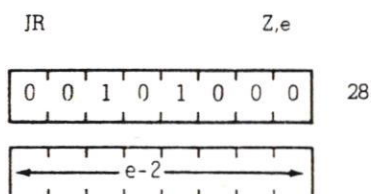
Az eredményként kapott tárgy kód, és a PC végső értéke az alábbi táblázaton látható:

| Rekesz | CIMKE:      | OP kód                           |
|--------|-------------|----------------------------------|
| 47C    | -           | PC az ugrás után                 |
| 47D    | -           |                                  |
| 47E    | -           |                                  |
| 47F    | -           |                                  |
| 480    | JR NZ,CIMKE | 20                               |
| 481    |             | FA /-6, 2-es komplement kód-ban/ |

## JR Z,e

Művelet: Ha Z=0, a PC a normál módon inkrementálódik  
Ha Z=1, PC ← PC+e

Formátum: OP kód Operandumok



### Leírás:

Az utasítás a Z zérus bit vizsgálatának eredményétől függően egy más programszegmensre történő relatív címzésű feltételes vezérlésátadást tesz lehetővé. Ha a vizsgálat során a bit tartalma 1, akkor az „e” 2-es komplementű egész számként értelmezett címkiegészítés értéke a programszámláló /PC/ tartalmához adódik, és a következő utasítást az így kapott címről fogja a CPU kiemelni. Az ugrás hossza az utasítás OP kódját tartalmazó memóriarekesz címétől értendő, és tartománya -126...+129 byte lehet. Az assembler kiszámolja az „e” címkiegészítés értékét, majd az utasítás végrehajtása során automatikusan 2-vel megnövelt PC tartalom miatt szükségessé váló korrekciót /2-vel való csökkentést/ is elvégzi. /A PC tartalomnak ez a 2-vel történő növelése az utasítás 2 byte-os hosszának következménye/. Ha a Z bit 0-t tartalmaz, a következő utasítást a normál módon soron következő memóriarekeszből emeli ki a CPU.

### Ha a feltétel teljesül:

M ciklusok: 3    T állapotok: 12/4,3,5/    4 MHz V.i.: 3.00

### Ha a feltétel nem teljesül:

M ciklusok: 2    T állapotok: 7/4,3/    4 MHz V.i.: 1.75

### Példa:

A Z zérus bit 1-es állapotban van, és a 300H memóriacímű 5 rekesznyi táblázatot kell előre ugrani. Az assembler nyelvű utasítás a következő:

JR Z,CIMKE

Az eredményként kapott tárgy kód, és a PC végső értéke az alábbi táblázaton látható:

| <u>Rekesz</u> |               | <u>OP kód</u>    |
|---------------|---------------|------------------|
| 300           | JR    Z,CIMKE | 28               |
| 301           | -             | 03               |
| 302           | -             |                  |
| 303           | -             |                  |
| 304           | -             |                  |
| 305           | CIMKE:    -   | PC az ugrás után |

# LD A,(BC)

18080 LDAX BC

Művelet:  $A \leftarrow (BC)$

Formátum:            OP kód            Operandusok  
                         LD                    A,(BC)

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 0A

## Leírás:

A BC regiszterpár tartalma által specifikált memóriarekesz tartalma az akkumulátorba töltődik.

M ciklusok: 2    T állapotok: 7/4,3/    4 MHz V.i.: 1.75

## Példa:

Ha a BC regiszterpár tartalma számértékben 4747H, és a 4747H című memóriarekeszben lévő byte 12H, akkor az

LD A,(BC)

utasítás végrehajtásának eredményeként a 12H byte az A regiszterbe töltődik.

# LD A,(DE)

18080 LDAX DE

---

Művelet: A ← (DE)

Formátum:            OP kód            Operandusok  
                         LD                    A,(DE)

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 1A

Leírás:

A DE regiszterpár tartalma által specifikált memóriarekesz tartalma az akkumulátorba töltődik.

M ciklusok: 2    T állapotok: 7/4,3/    4 MHz V.i.: 1.75

Példa:

Ha a DE regiszterpár tartalma számértékben 30A2H, és a 30A2H című memóriarekeszben lévő byte 22H, akkor az

LD A,(DE)

utasítás végrehajtásának eredményeként a 22H byte az A regiszterbe töltődik.

# LD A,I

Művelet:  $A \leftarrow I$

Formátum: OP kód Operandumok

LD A,I

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 57

Leírás:

Az I Interrupt Vector Register tartalma az akkumulátorba töltődik.

M ciklusok: 2 T állapotok: 9/4,5/ 4 MHz V.i.: 2.25

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az I regiszter negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az I regiszter nulla; egyébként törlődik
- H: törlődik
- P/V: IFF2 értékét veszi fel
- N: törlődik
- C: változatlan marad

Példa:

Ha az Interrupt Vector Register-ben lévő byte 4AH, akkor az

LD A,I

utasítás végrehajtásának eredményeként az akkumulátor is 4AH-t fog tartalmazni.

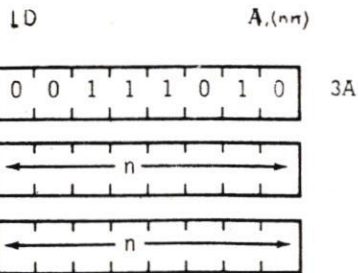


# LD A,(nn)

13030 LDA AA

Művelet:  $A \leftarrow (nn)$

Formátum: OP kód            Operandumok



## Leírás:

Az nn operandusok által specifikált memóriarekesz tartalma az akkumulátorba töltődik. Az első n operandus egy 2 byte-os memóriacím alacsonyabb helyiértékű byte-ja.

M ciklusok: 4      T állapotok: 13/4,3,3,3/      4 MHz V.i.: 3.25

## Példa:

Ha nn tartalma számértékben 8832H, és a 8832H című memóriarekeszben lévő byte 04H, akkor az

LD A,(nn)

utasítás végrehajtásának eredményeként a 04H byte az akkumulátorba töltődik.

# LD A,R

Művelet: A ← R

Formátum: OP kód Operandumok

LD

A,R

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 5F

Leírás:

Az R Memory Refresh Register tartalma az akkumulátorba töltődik.

M ciklusok: 2      T állapotok: 9/4,5/      4 MHz V.i.: 2.25

A jelzőbit regiszter bitjei:

- S: 1-es állapotba kerül, ha az R regiszter negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az R regiszter nulla; egyébként törlődik
- H: törlődik
- P/V: IFF2 értékét veszi fel
- N: törlődik
- C: változatlan marad

Példa:

Ha a Memory Refresh Register-ben lévő byte 4AH, akkor az

LD A,R

utasítás végrehajtásának eredményeként az akkumulátor is 4AH-t fog tartalmazni.

# LD (BC),A

1980 STAX BC

Művelet: (BC) ← A

Formátum:      OP kód            Operandusok  
                 LD                    (BC),A

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 02

## Leírás:

A BC regiszterpár tartalma által specifikált memóriarekeszben töltődik az akkumulátor tartalma.

M ciklusok: 2      T állapotok: 7/4,3/      4 MHz V.i.: 1.75

## Példa:

Ha az akkumulátor 7AH-t, és a BC regiszterpár 1212H-t tartalmaz, akkor az

LD (BC),A

utasítás végrehajtásának eredményeként a 7AH a 1212H című memóriarekeszbe töltődik.

# LD (DE),A

18030 STAX DE

Művelet: (DE) ← A

Formátum:            OP kód            Operandusok  
                         LD                    (DE),A

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 12

## Leírás:

Az akkumulátor tartalma a DE regiszterpár tartalma által specifikált memóriarekeszbe töltődik.

M ciklusok: 2    T állapotok: 7/4,3/    4 MHz V.i.: 1.75

## Példa:

Ha a DE regiszterpár tartalma 1128H, és az akkumulátorban lévő byte A0H, akkor az

LD (DE),A

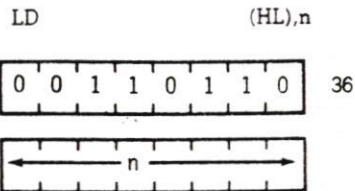
utasítás végrehajtásának eredményeként A0H a 1128H című memóriarekeszbe töltődik.

# LD (HL),n

18080 MVI M,n

Művelét: (HL) ← n

Formátum: OP kód Operandusok



## Leírás:

Az n egész szám a HL regiszterpár tartalma által specifikált memóriarekeszbe töltődik.

M ciklusok: 3    T állapotok: 10/4,3,3/    4 MHz V.i.: 2.50

## Példa:

Ha a HL regiszterpár 4444H-t tartalmaz, akkor az

LD (HL),28H

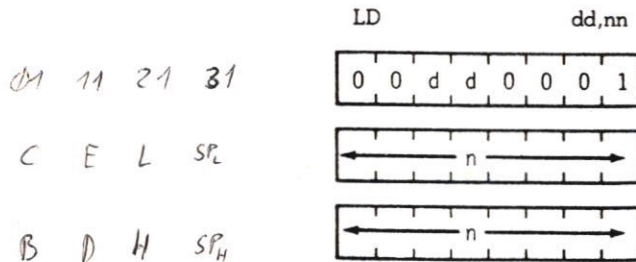
utasítás végrehajtásának eredményeként a 4444H című memóriarekeszbe 28H töltődik.

# LD dd,nn

13080 LX 1 dd,nn

Művelet: dd ← nn

Formátum: OP kód      Operandusok



## Leírás:

Az nn 2 byte-os egész szám a dd regiszterpárba töltődik, ahol a dd a BC, DE, HL vagy SP regiszterpárok egyikét jelöli, ezek tárgykódban a következők lehetnek:

| <u>Regiszterpár</u> | <u>dd</u> |
|---------------------|-----------|
| BC                  | 00        |
| DE                  | 01        |
| HL                  | 10        |
| SP                  | 11        |

A tárgykódban elsőként szereplő n operandus az alacsonyabb helyiértékű byte-ot képviseli.

M ciklusok: 3    T állapotok: 10/4,3,3/    4 MHz V.i.: 2.50

## Példa:

Az

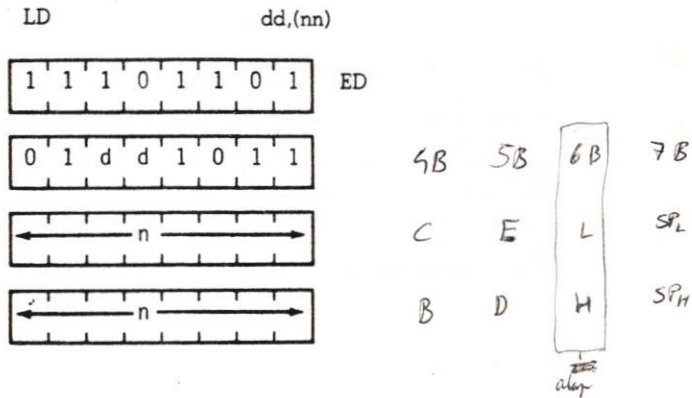
LD HL,5000H

utasítás végrehajtása után a HL regiszterpár tartalma 5000H lesz.

# LD dd,(nn)

Művelet:  $dd_H \leftarrow (nn+1), \quad dd_L \leftarrow (nn)$

Formátum:      OP kód      Operandusok



## Leírás:

Az nn című memóriarekesz tartalma a dd regiszterpár alacsonyabb helyiértékű regiszterébe töltődik, míg az nn+1 című memóriarekesz tartalma a dd regiszterpár magasabb helyiértékű byte-jába töltődik. A dd regiszterpár a BC, DE, HL vagy SP regiszterpárok egyikét jelöli, amelyek tárgykódban a következők lehetnek:

| <u>Regiszterpár</u> | <u>dd</u> |
|---------------------|-----------|
| BC                  | 00        |
| DE                  | 01        |
| HL                  | 10        |
| SP                  | 11        |

A fenti tárgykódban elsőként szereplő n operandus nn alacsonyabb helyiértékű byte-ja.

M ciklusok: 6    T állapotok: 20/4,4,3,3,3,3/ 4 MHz V.i.: 5.00

## Példa:

Ha a 2130H című memóriarekeszben lévő byte 65H, és a 2131H című memóriarekeszben lévő byte 78H, akkor az

LD BC,(2130H)

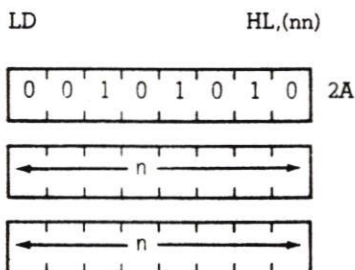
utasítás végrehajtásának eredményeként a BC regiszterpár 7865H-t fog tartalmazni.

# LD HL,(nn)

18030 LHLD nn

Művelet: H ← (nn+1), L ← (nn)

Formátum            OP kód            Operandusok



## Leírás:

Az nn című memóriarekesz tartalma a HL regiszterpár alacsonyabb helyiértékű regiszterébe /L regiszter/ töltődik, míg az nn+1 című memóriarekesz tartalma a HL regiszterpár magasabb helyiértékű byte-jába /H regiszter/ töltődik. A fenti tárgykódban elsőként szereplő n operandus nn alacsonyabb helyiértékű byte-ja.

M ciklusok: 5    T állapotok: 16/4,3,3,3,3/    4 MHz V.i.: 4.00

## Példa:

Ha a 4545H című memóriarekesz 37H-t, és a 4546H című memóriarekesz 13H-t tartalmaz, akkor az

LD HL,(4545H)

utasítás végrehajtásának eredményeként a HL regiszterpár A137H-t fog tartalmazni.

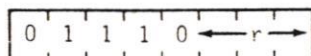


# LD (HL),r

Művelet: (HL) ← r

Formátum: OP kód      Operandusok

LD                      (HL),r



## Leírás:

Az r regiszter tartalma a HL regiszterpár tartalma által specifikált memóriarekeszbe töltődik. Az r az A, B, C, D, E, H vagy L regisztereket azonosítja, amelyek tárgykódban a következők lehetnek:

| <u>Regiszter</u> | <u>r</u> |    |
|------------------|----------|----|
| A                | 111      | 77 |
| B                | 000      | 70 |
| C                | 001      | 71 |
| D                | 010      | 72 |
| E                | 011      | 73 |
| H                | 100      | 74 |
| L                | 101      | 75 |

M ciklusok: 2      T állapotok: 7/4,3/      4 MHz V.i.: 1.75

## Példa:

Ha a HL regiszterpár tartalma a 2146H memóriacímre mutat, és a B regiszterben lévő byte 29H, akkor az

LD (HL),B

utasítás végrehajtásának eredményeként a 2146H című memóriarekesz szintén 29H-t fog tartalmazni.

# LD I,A

---

Művelet: I ← A

Formátum: OP kód Operandumok

| LD              | I,A |
|-----------------|-----|
| 1 1 1 0 1 1 0 1 | ED  |
| 0 1 0 0 0 1 1 1 | 47  |

Leírás:

Az akkumulátor tartalma az I Interrupt Control Vector Register-be töltődik.

M ciklusok: 2 T állapotok: 9/4,5/ 4 MHz V.i.: 2.25

Példa:

Ha az akkumulátor tartalma számértékben 81H, akkor az

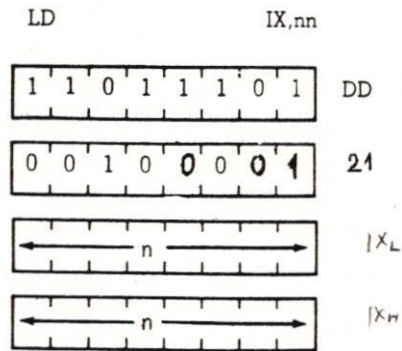
LD I,A

utasítás végrehajtása után az Interrupt Vector Register szintén 81H-t fog tartalmazni.

# LD IX,nn

Művelet: IX ← nn

Formátum: OP kód            Operandusok



## Leírás:

Az nn egész szám az IX index regiszterbe töltődik. A fenti assemblált tárgykódban elsőként szereplő n operandus az alacsonyabb helyiértékű byte.

M ciklusok: 4    T állapotok: 14/4,4,3,3/    4 MHz V.i.: 3.50

## Példa:

Az

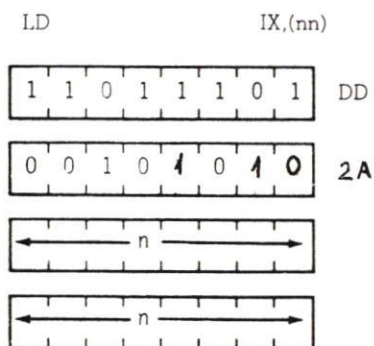
LD IX,45A2H

utasítás végrehajtásának eredményeként az index regiszter a 45A2H egész számot fogja tartalmazni.

## LD IX,(nn)

Művelet:  $IX_H \leftarrow (nn+1), IX_L \leftarrow (nn)$

Formátum: OP kód            Operandusok



### Leírás:

Az nn című memóriarekesz tartalma az IX index regiszter alacsonyabb helyiértékű byte-jába töltődik, míg az nn+1 című memóriarekesz tartalma az IX index regiszter magasabb helyiértékű byte-jába töltődik. A fenti tárgykódban elsőként szereplő n operandus nn alacsonyabb helyiértékű byte-ja.

M ciklusok: 6    T állapotok: 20/4,4,3,3,3,3/ 4 MHz V.i.: 5.00

### Példa:

Ha a 6666H című memóriarekesz 92H-t, és a 6667H című memóriarekesz DAH-t tartalmaz, akkor az

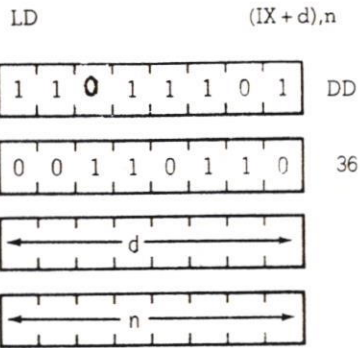
LD IX,(6666H)

utasítás végrehajtásának eredményeként az IX index regiszter DA92H-t fog tartalmazni.

# LD (IX+d),n

Művelet: (IX+d) ← n

Formátum: OP kód            Operandusok



## Leírás:

Az n operandus abba a memóriarekeszbe töltődik, amelynek címét az IX index regiszter, és a d 2-es komplementű kódban adott címkiegészítés operandus összege jelöli ki.

M ciklusok: 5    T állapotok: 19/4,4,3,5,3/    4 MHz V.i.: 4.75

## Példa:

Ha az IX index regiszter tartalma számértékben 219AH, az

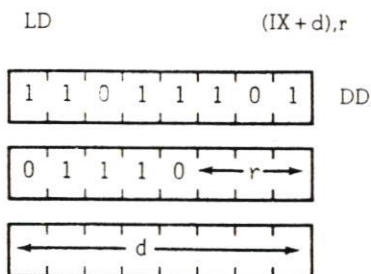
LD (IX+5H),5AH

utasítás végrehajtásának eredményeként az 5AH byte a 219FH című memóriarekeszbe töltődik.

# LD (IX+d),r

Művelet: (IX+d) ← r

Formátum:      OP kód              Operandumok



## Leírás:

Az r regiszter tartalma abba a memóriarekeszbe töltődik, melynek címét az IX index regiszter, és a d 2-es komplement egész számként értelmezett címkiegészítés összege határozza meg. Az r szimbólum az A, B, C, D, E, H vagy L regisztereket azonosítja, amelyek tárgykódban a következők:

| <u>Regiszter</u> | <u>r</u> |
|------------------|----------|
| A                | 111      |
| B                | 000      |
| C                | 001      |
| D                | 010      |
| E                | 011      |
| H                | 100      |
| L                | 101      |

M ciklusok: 5    T állapotok: 19/4,4,3,5,3/    4 MHz V.i.:

## Példa:

Ha a C regiszterben lévő byte 1CH, és az IX index regiszter 3100H-t tartalmaz, akkor az

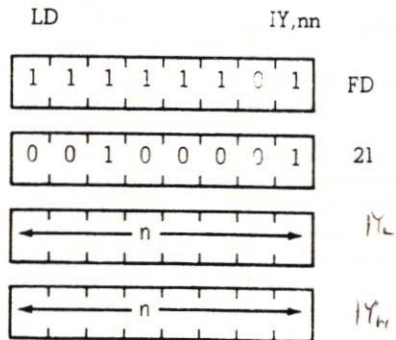
LD (IX+6H),C

utasítás elvégzi a 3100H+6H összegzést, és 1CH-t a 3106H memóriarekeszbe tölti.

# LD IY,nn

t: IY ← nn

um: OP kód Operandusok



:

egész szám az IY index regiszterbe töltődik. A fenti ódban elsőként szereplő n operandus az alacsonyabb értékű byte-ot jelenti.

usok: 4 T állapotok: 14/4,4,3,3/ 4 MHz V.i.: 3.50

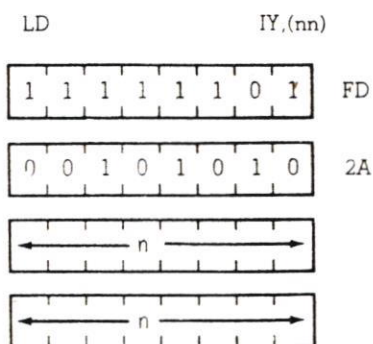
LD IY,7733H

ás végrehajtásának eredményeként az IY index regiszter H egész számot fogja tartalmazni.

## LD IY,(nn)

Művelet:  $IY_H \leftarrow (nn+1), \quad IY_L \leftarrow (nn)$

Formátum:            OP kód            Operandusok



### Leírás:

Az nn című memóriarekesz tartalma az IY index regiszter alacsonyabb helyiértékű byte-jába töltődik, míg az nn+1 című memóriarekesz tartalma az IY index regiszter magasabb helyiértékű byte-jába töltődik. A fenti tárgykódban elsőként szereplő n operandus az alacsonyabb helyiértékű byte.

M ciklusok: 6    T állapotok: 20/4,4,3,3,3,3/ 4 MHz V.i.: 5

### Példa:

Ha a 6666H című memóriarekesz 92H-t, és a 6667H című memóriarekesz DAH-t tartalmaz, akkor az

LD IY,(6666H)

utasítás végrehajtásának eredményeként az IY index regiszter tartalma DA92H lesz.

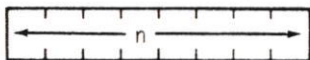
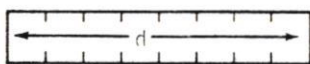
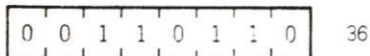
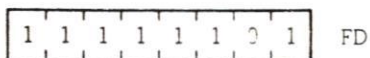


# LD (IY+d),n

Ut: (IY+d) ← n

um: OP kód                      Operandusok

LD                                      (IY+d),n



egész szám abba a memóriarekeszbe töltődik, amelynek az index regiszter, és a d egész számként értelmezett egészítés összege határozza meg.

usok: 5      T állapotok: 19/4,4,3,5,3/      4 MHz V.i.: 4.75

IY index regiszter tartalma számértékben A940H, az

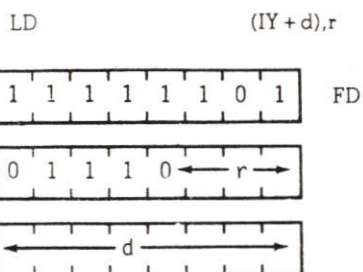
LD (IY+10H),97H

ás végrehajtásának eredményeként a 97H byte az A950H emóriarekeszbe töltődik.

# LD (IY+d),r

Művelet:  $(IY+d) \leftarrow r$

Formátum: OP kód            Operandusok



## Leírás:

Az r regiszter tartalma abba a memóriarekeszbe töltődik, melynek címét az IY index regiszter, és a d 2-es komplement egész számként értelmezett címkiegészítés összege határozza meg. Az r szimbólum az alábbi táblázat alapján specifikálható:

| <u>Regiszter</u> | <u>r</u> |
|------------------|----------|
| A                | 111      |
| B                | 000      |
| C                | 001      |
| D                | 010      |
| E                | 011      |
| H                | 100      |
| L                | 101      |

M ciklusok: 5    T állapotok: 19/4,4,3,5,3/    4 MHz V.i.: 4

## Példa:

Ha a C regiszterben lévő byte 48H, és az IY index regiszter 2A11H-t tartalmaz, akkor az

LD (IY+4H),C

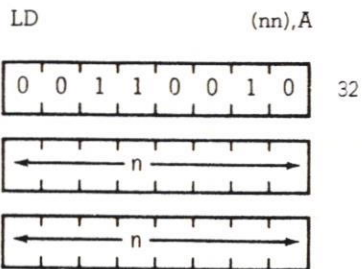
utasítás elvégzi a 2A11H+4H összegzést, és 48H-t a 2A15H című memóriarekeszbe tölti.

# LD (nn),A

1 feje STA nn

t: (nn) ← A

um: OP kód Operandusok



:

akkumulátor tartalma az nn operandusok által specifikált memóriarekeszbe töltődik. A fenti tárgykódban elsőként szereplő operandus nn alacsonyabb helyiértékű byte-ja.

Operandusok: 4 T állapotok: 13/4,3,3,3/ 4 MHz V.i.: 3.25

akkumulátor tartalma D7H, akkor az

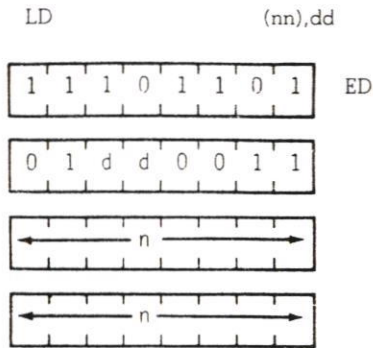
LD (3141H),A

instrukció végrehajtása után D7H a 3141H című memóriarekeszbe kerül.

# LD (nn),dd

Művelet:  $(nn+1) \leftarrow dd_H, (nn) \leftarrow dd_L$

Formátum:            OP kód            Operandusok



## Leírás:

A dd regiszterpár alacsonyabb helyiértékű byte-ja az nn című memóriarekeszbe töltődik, míg a magasabb helyiértékű byte az nn+1 című memóriarekeszbe töltődik. A dd regiszterpár a BC, DE, HL vagy SP regiszterpárok valamelyikét azonosítja, melyek tárgykódban a következők:

| <u>Regiszterpár</u> | <u>dd</u> |    |
|---------------------|-----------|----|
| BC                  | 00        | 43 |
| DE                  | 01        | 53 |
| HL                  | 10        |    |
| SP                  | 11        | 73 |

A fenti tárgykódban elsőként szereplő n operandus egy 2 byte-os memóriacím alacsonyabb helyiértékű byte-ja.

M ciklusok: 6    T állapotok: 20/4,4,3,3,3,3/    4 MHz    V.i.: 5.00

## Példa:

Ha a BC regiszterpár tartalma számértékben 4644H, akkor az

LD (1000 H),BC

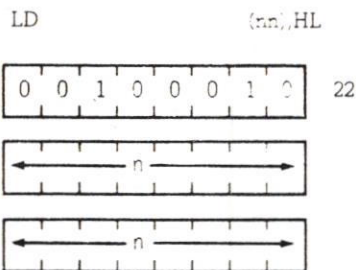
utasítás végrehajtásának eredményeként 44H az 1000H című memóriarekeszbe, 46H pedig az 1001H című memóriarekeszbe töltődik.

# LD (nn),HL

1880 SHLD nn

efekt:  $(nn+1) \leftarrow H, (nn) \leftarrow L$

mátum: OP kód Operandusok



rás:

L regiszterpár tartalmának alacsonyabb helyiértékű byte-ja regiszter/ az nn című memóriarekeszbe töltődik, és HL tartalmának magasabb helyiértékű byte-ja pedig /H regiszter/ az l című memóriarekeszbe töltődik. A fenti assemblált tárgyban elsőként szereplő n operandus nn alacsonyabb helyiértékű byte-ja.

iklusok: 5 T állapotok: 16/4,3,3,3,3/ 4 MHz V.i.: 4.00

da:

a HL regiszterpár tartalma 483AH, akkor az

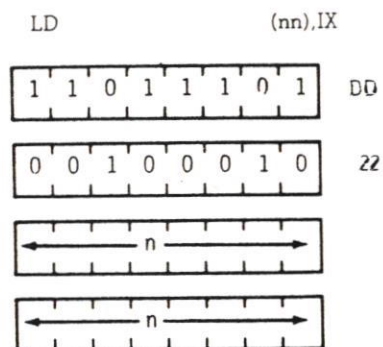
LD (B229H),HL

sítás végrehajtásának eredményeként a B229H című memóriarekesz tartalma 3AH, a B22AH című memóriarekesz pedig 48H lesz.

# LD (nn),IX

Művelet:  $(nn+1) \leftarrow IX_H, (nn) \leftarrow IX_L$

Formátum: OP kód            Operandusok



## Leírás:

Az IX index regiszter tartalmának alacsonyabb helyiértékű byte-ja az nn című memóriarekeszbe töltődik, míg a magasabb helyiértékű byte az nn+1 című memóriarekeszbe töltődik. A fenti tárgykódban elsőként szereplő n operandus nn alacsonyabb helyiértékű byte-ja.

M ciklusok: 6    T állapotok: 20/4,4,3,3,3,3/    4 MHz V.i.: 5.00

## Példa:

Ha az IX index regiszter 5A30H-t tartalmaz, akkor az

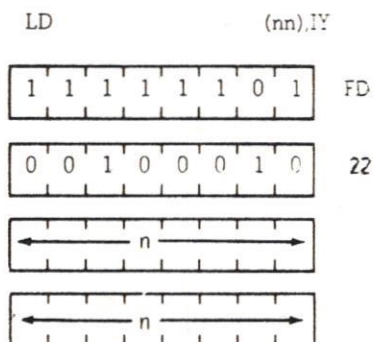
LD (4392H),IX

utasítás végrehajtásának eredményeként a 4392H című memóriarekesz tartalma számértékben 30H lesz, a 4393H című memóriarekesz tartalma pedig 5AH lesz.

# LD (nn),IY

Operandum:  $(nn+1) \leftarrow IY_H, (nn) \leftarrow IY_L$

Operandum: OP kód Operandumok



Operandum:

IY index regiszter tartalmának alacsonyabb helyiértékű byte-ja az nn című memóriarekeszbe, míg a magasabb helyiértékű byte az nn+1 című memóriarekeszbe töltődik. A fenti példában elsőként szereplő n operandus nn alacsonyabb helyiértékű byte-ja.

Ciklusok: 6 T állapotok: 20/4,4,3,3,3,3/ 4 MHz V.i.: 5.00

Operandum:

Ha az IY index regiszter 4174H-t tartalmaz, akkor az

LD (8838H),IY

Operandum végrehajtásának eredményeként a 8838H című memóriarekesz tartalma számértékben 74H lesz, a 8839H című memóriarekesz tartalma pedig 41H lesz.

# LD R,A

---

Művelet: R ← A

Formátum: OP kód            Operandusok

LD

R,A

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 4F

Leírás:

Az akkumulátor tartalma az R Memory Refresh regiszterbe töltődik.

M ciklusok: 2    T állapotok: 9/4,5/    4 MHz V.i.: 2.25

Példa:

Ha az akkumulátor tartalma számértékben B4H, akkor az

LD R,A

utasítás végrehajtása után az R Memory Refresh regiszter szintén B4H-t fog tartalmazni.

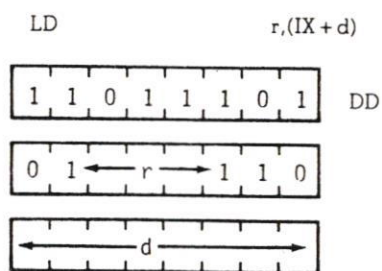




# LD r,(IX+d)

Művelet:  $r \leftarrow (IX+d)$

Formátum: OP kód          Operandusok



## Leírás:

Az IX+d operandus /az IX index regiszter tartalma és egész számmal adott címkiegészítés által kijelölt memóriaszám tartalmának összege/ az r regiszterbe töltődik, a az A, B, C, D, E, H vagy L regisztereket azonosítja, a tárgykódban a következők:

| <u>Regiszter</u> | <u>r</u> |
|------------------|----------|
| A                | 111      |
| B                | 000      |
| C                | 001      |
| D                | 010      |
| E                | 011      |
| H                | 100      |
| L                | 101      |

M ciklusok: 5    T állapotok: 19/4,4,3,5,3/    4 MHz V.i.:

## Példa:

Ha az IX index regiszter tartalma számértékben 25AFH, akkor az

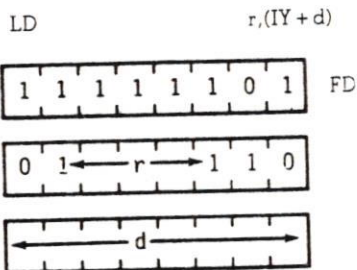
LD B,(IX+19H)

utasítás a 25AFH+19H összeadást eredményezi, ami a 25C00H memóriarekeszre mutat. Ha ez a memóriarekesz 39H-t tartalmaz, az utasítás végrehajtásának eredményeként a B regiszter szintén 39H-t fog tartalmazni.

# LD r,(IY+d)

: r ← (IY+d)

OP kód                      Operandusok



d operandus /az IY index regiszter tartalma, és a d számmal adott címkiegészítés által kijelölt memóriaregiszter tartalmának összege/ az r regiszterbe töltődik, ahol r B, C, D, E, H vagy L regisztereket azonosítja, amelyek kódokban a következők:

| Regiszter | r   |
|-----------|-----|
| A         | 111 |
| B         | 000 |
| C         | 001 |
| D         | 010 |
| E         | 011 |
| H         | 100 |
| L         | 101 |

Asok: 5    T állapotok: 19/4,4,3,5,3/    4 MHz V.i.: 4.75

IY index regiszter tartalma számértékben 25AFH, az

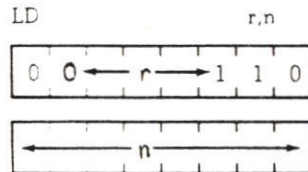
LD B, (IY+19H)

és a 25AFH+19H összeadást eredményezi, amely a 25C8H memóriarekeszre mutat. Ha ez a memóriarekesz 39H-t tartalmaz, az utasítás végrehajtásának eredményeként a B regiszterintén 39H-t fog tartalmazni.

# LD r,n

Művelet:  $r \leftarrow n$

Formátum: OP kód      Operandusok



Leírás:

Az n 8 bites egész szám az r által meghatározott regiszterek egyikébe töltődik. Az r az A, B, C, D, E, H vagy L regisztereket azonosítja, amelyek tárgykódban a következők:

| <u>Regiszter</u> | <u>r</u> |
|------------------|----------|
| A                | 111      |
| B                | 000      |
| C                | 001      |
| D                | 010      |
| E                | 011      |
| H                | 100      |
| L                | 101      |

M ciklusok: 2    T állapotok: 7/4,3/    4 MHz V.i.: 1.75

Példa:

Az

LD E,A5H

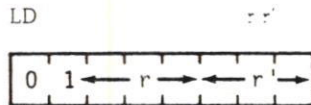
utasítás végrehajtása után az E regiszter tartalma A5H lesz.

# LD r,r'

18020 MOV r,r'

Művelet:  $r \leftarrow r'$

Formátum: OP kód            Operandusok



Leírás:

Az r' operandus által meghatározott háttérregiszter tartalma az r operandus által meghatározott regiszterbe töltődik. Megjegyzés: r,r' az A, B, C, D, E, H vagy L regiszterek bármelyikét azonosítja, amelyek tárgykódban a következők:

| <u>Regiszter</u> | <u>r,r'</u> |
|------------------|-------------|
| A                | 111         |
| B                | 000         |
| C                | 001         |
| D                | 010         |
| E                | 011         |
| H                | 100         |
| L                | 101         |

M ciklusok: 1    T állapotok: 4    4 MHz V.i.: 1.00

Példa:

Ha a H regiszter tartalma számértékben 8AH, és az E regiszter 10H-t tartalmaz, akkor az

LD H,E

utasítás végrehajtása után mindkét regiszter 10H-t fog tartalmazni.

# LD SP,HL

19030 SPHL

---

Művelet: SP ← HL

Formátum: OP kód Operandusok

LD SP,HL

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 F9

Leírás:

A HL regiszterpár tartalma az SP veremtár-mutató regiszterbe töltődik.

M ciklusok: 1 T állapotok: 6 4 MHz V.i.: 1.50

Példa:

Ha a HL regiszterpár 442EH-t tartalmaz, akkor az

LD SP,HL

utasítás végrehajtása után az SP regiszter is 442EH-t fog tartalmazni.

# LD SP,IX

Művelet: SP ← IX

Formátum: OP kód Operandusok

| LD  | SP,IX |   |   |   |   |   |   |   |    |
|---|-------|---|---|---|---|---|---|---|----|
| <table border="1"><tr><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td></tr></table> | 1     | 1 | 0 | 1 | 1 | 1 | 0 | 1 | DD |
| 1   | 1     | 0 | 1 | 1 | 1 | 0 | 1 |   |    |
| <table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td></tr></table> | 1     | 1 | 1 | 1 | 1 | 0 | 0 | 1 | F9 |
| 1   | 1     | 1 | 1 | 1 | 0 | 0 | 1 |   |    |

Leírás:

Az IX index regiszter 2 byte-os tartalma az SP regiszterbe töltődik.

M ciklusok: 2 T állapotok: 10/4,6/ 4 MHz V.i.: 2.50

Példa:

Ha az IX index regiszter tartalma 98DAH, akkor az

LD SP,IX

utasítás végrehajtása után az SP regiszter tartalma szintén 98DAH lesz.

# LD SP,IY

---

Művelet: SP ← IY

Formátum: OP kód          Operandumok

| LD              | SP,IY |
|-----------------|-------|
| 1 1 1 1 1 1 0 1 | FD    |
| 1 1 1 1 1 0 0 1 | F9    |

Leírás:

Az IY index regiszter 2 byte-os tartalma az SP regiszterbe töltődik.

M ciklusok: 2    T állapotok: 10/4,6/    4 MHz V.i.: 2.50

Példa:

Ha az IY index regiszter tartalmazza az A227H egész számot, akkor az

LD SP,IY

utasítás végrehajtásának eredményeként az SP regiszter is A227H-t fog tartalmazni.

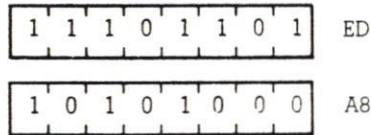


# LDD

Művelet: (DE) ← (HL), DE ← DE-1, HL ← HL-1, BC ← BC-1

Formátum: OP kód

LDD



## Leírás:

Ez a két byte hosszúságú utasítás a HL regiszterpár tartalma által megcímzett memóriarekeszben lévő byte-ot a DE regiszterpár tartalma által megcímzett memóriarekeszbe tölti, majd mindkét regiszterpár, és a BC /byte számláló/ regiszterpár is dekrementálódik.

M ciklusok: 4    T állapotok: 16/4,4,3,5/    4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

S: változatlan marad  
Z: változatlan marad  
H: törlődik  
P/V: 1-es állapotba kerül, ha BC-1≠0; egyébként törlődik  
N: törlődik  
C: változatlan marad

## Példa:

Ha a HL regiszterpár 1111H-t tartalmaz, az 1111H című memóriarekeszben lévő byte 88H, a DE regiszterpár tartalma 2222H, a 2222H című memóriarekeszben lévő byte 66H, és a BC regiszterpár 7H-t tartalmaz, akkor az

LDD

utasítás végrehajtásának eredményeként a regiszterpárok és a memóriarekeszek tartalma a következőképpen alakul:

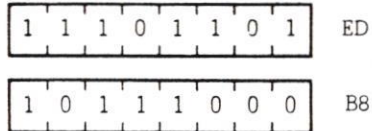
|         |   |       |
|---------|---|-------|
| HL      | : | 1110H |
| (1111H) | : | 88H   |
| DE      | : | 2221H |
| (2222H) | : | 88H   |
| BC      | : | 6H    |

# LDDR

Művelet: (DE) ← (HL), DE ← DE-1, HL ← HL-1, BC ← BC-1

Formátum: OP kód

LDDR



## Leírás:

A két byte hosszúságú utasítás a HL regiszterpár tartalma által megcímzett memóriarekeszben lévő byte-ot a DE regiszterpár tartalma által megcímzett memóriarekeszbe tölti, majd mindkét regiszterpár, és a BC regiszterpár /byte számláló/ is dekrementálódik. Ha a dekrementálás eredményeként BC tartalma zérus lesz, akkor az utasítás befejeződik. Ha viszont a BC tartalma nem zérus, akkor a programszámláló tartalma 2-vel dekrementálódik, és az utasítás megismétlődik. Megjegyezzük, hogy ha BC tartalma az utasítás végrehajtását megelőzően zérusra lett beállítva, az utasítás 64K-szor ciklikusan ismétlődik. Minden egyes byte átvitelét követően 2 felfrissítési ciklus zajlik le, és ilyenkor megszakítási kérelmek is érvényre juthatnak.

## Ha BC≠0:

M ciklusok: 5 T állapotok: 21/4,4,3,5,5/ 4 MHz V.i.: 5.25

## Ha BC=0:

M ciklusok: 4 T állapotok: 16/4,4,3,5/ 4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

S: változatlan marad  
Z: változatlan marad  
H: törlődik  
P/V: törlődik  
N: törlődik  
C: változatlan marad

## Példa:

Ha a HL regiszterpár 1114H-t tartalmaz, a DE regiszterpár

# LDDR

---

2225H-t, a BC regiszterpár 0003H-t tartalmaz, és a memóriarekeszek tartalma a következőképpen alakul:

|         |   |     |         |   |     |
|---------|---|-----|---------|---|-----|
| (1114H) | : | A5H | (2225H) | : | C5H |
| (1113H) | : | 36H | (2224H) | : | 59H |
| (1112H) | : | 88H | (2223H) | : | 66H |

akkor az

LDDR

utasítás végrehajtása után a regiszterpárok és a memóriarekeszek tartalma a következő lesz:

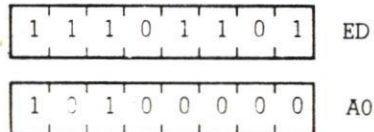
|         |   |       |         |   |     |
|---------|---|-------|---------|---|-----|
| HL      | : | 1111H |         |   |     |
| DE      | : | 2222H |         |   |     |
| BC      | : | 0000H |         |   |     |
| (1114H) | : | A5H   | (2225H) | : | A5H |
| (1113H) | : | 36H   | (2224H) | : | 36H |
| (1112H) | : | 88H   | (2223H) | : | 88H |

# LDI

Művelet: (DE) ← (HL), DE ← DE+1, HL ← HL+1, BC ← BC-1

Formátum: OP kód

LDI



## Leírás:

Az utasítás a HL regiszterpár tartalma által megcímzett memóriarekeszben lévő byte-ot a DE regiszterpár tartalma által megcímzett memóriarekeszbe tölti, majd mindkét regiszterpár inkrementálódik, és a BC /byte számláló/ regiszterpár dekrementálódik.

M ciklusok: 4    T állapotok: 16/4,4,3,5/    4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

S: változatlan marad  
Z: változatlan marad  
H: törlődik  
P/V: 1-es állapotba kerül, ha BC-1≠0; egyébként törlődik  
N: törlődik  
C: változatlan marad

## Példa:

Ha a HL regiszterpár 1111H-t tartalmaz, a 1111H című memóriarekeszben lévő byte 88H, a DE regiszterpár 2222H-t tartalmaz, és a 2222H című memóriarekeszben lévő byte 66H, a BC regiszterpár tartalma 7H, akkor az

LDI

utasítás végrehajtásának eredményeként a regiszterpárok és a memóriarekeszek tartalma a következőképpen alakul:

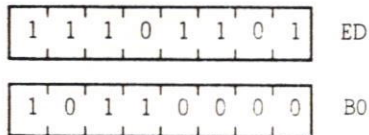
|         |   |       |
|---------|---|-------|
| HL      | : | 1112H |
| (1111H) | : | 88H   |
| DE      | : | 2223H |
| (2222H) | : | 88H   |
| BC      | : | 6H    |

# LDIR

Művelet: (DE) ← (HL), DE ← DE+1, HL ← HL+1, BC ← BC-1

Formátum: OP kód

LDIR



## Leírás:

A két byte hosszúságú utasítás a HL regiszterpár tartalma által megcímzett memóriarekeszben lévő byte-ot a DE regiszterpár tartalma által megcímzett memóriarekeszbe tölti, majd mindkét regiszterpár inkrementálódik, és a BC /byte számláló/ regiszterpár dekrementálódik. Ha a dekrementálás eredményeként BC tartalma zérus lesz, akkor az utasítás befejeződik. Ha viszont a BC tartalma nem zérus, akkor a programszámláló tartalma 2-vel dekrementálódik, és az utasítás megismétlődik. Megjegyezzük, hogy ha BC tartalma az utasítás végrehajtását megelőzően zérusra lett beállítva, az utasítás 64K-szor ciklikusan megismétlődik. Minden egyes byte átvitelét követően két felfrissítési ciklus zajlik le, és ilyenkor megszakítási kérések is érvényre juthatnak.

## BC≠0 esetén:

M ciklusok: 5 T állapotok: 21/4,4,3,5,5/ 4 MHz V.i.: 5.25

## BC=0 esetén:

M ciklusok: 4 T állapotok: 16/4,4,3,5/ 4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

S: változatlan marad  
Z: változatlan marad  
H: törlődik  
P/V: törlődik  
N: törlődik  
C: változatlan marad

## Példa:

Ha a HL regiszterpár 1111H-t, a DE regiszterpár 2222H-t, a

# LDIR

---

BC regiszterpár 0003H-t tartalmaz, és a memóriarekeszek tartalma a következő:

|         |   |     |         |   |     |
|---------|---|-----|---------|---|-----|
| (1111H) | : | 88H | (2222H) | : | 66H |
| (1112H) | : | 36H | (2223H) | : | 59H |
| (1113H) | : | A5H | (2224H) | : | C5H |

akkor az

LDIR

utasítás végrehajtásának eredményeként a regiszterpárok, és a memóriarekeszek tartalma a következőképpen alakul:

|         |   |       |         |   |     |
|---------|---|-------|---------|---|-----|
| HL      | : | 1114H |         |   |     |
| DE      | : | 2225H |         |   |     |
| BC      | : | 0000H |         |   |     |
| (1111H) | : | 88H   | (2222H) | : | 88H |
| (1112H) | : | 36H   | (2223H) | : | 36H |
| (1113H) | : | A5H   | (2224H) | : | A5H |

# NEG

Művelet: A O-A  
Formátum: OP kód

NEG

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
|---|---|---|---|---|---|---|---|

 44

## Leírás:

Az utasítás az akkumulátor tartalmának 2-es komplementjét képezi, majd ezt az eredményt az akkumulátorba tölti. Ez a művelet azt jelenti, hogy az akkumulátor tartalma 0-ból kivonódik. Megjegyezzük, hogy 80H eredeti akkumulátor tartalom esetén az utasítás végrehajtása után az akkumulátor tartalma értelemszerűen változatlan marad.

M ciklusok: 2 T állapotok: 8/4,4/ 4 MHz V.i.: 2.00

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha a működés előtt az akkumulátor tartalma 80H volt; egyébként törlődik
- N: 1-es állapotba kerül
- C: 1-es állapotba kerül, ha a működés előtt az akkumulátor tartalma nem 00H volt; egyébként törlődik

## Példa:

Ha az akkumulátor tartalma

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|

a NEG

utasítás végrehajtásának eredményeként az akkumulátor tartalma

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|

lesz.

# NOP

---

Művelet: nincsen

Formátum: OP kód

NOP

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|

 00

Leírás:

Az utasítás végrehajtása során a CPU semmiféle aritmetikai vagy logikai műveletet nem végez.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00



# OR s

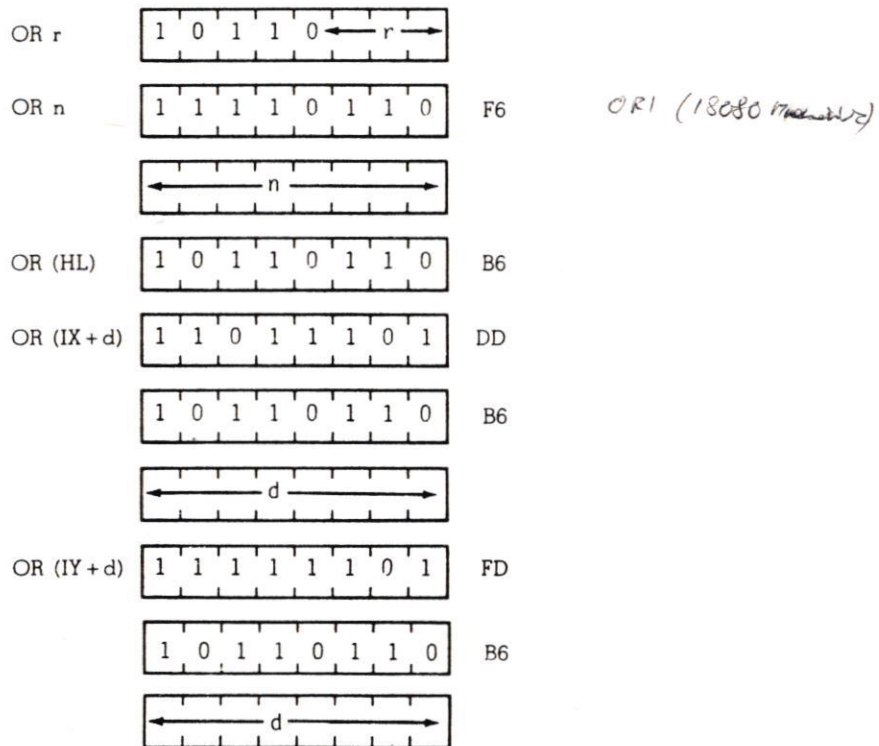
18080 ORA

Művelet:  $A \leftarrow A \vee s$

Formátum:

| Opcode | Operands |
|--------|----------|
| OR     | s        |

Az s operandus r, n, (HL), (IX+d) vagy (IY+d) bármelyike lehet, amint azt az ezzel az utasítással analóg ADD utasításoknál meghatároztuk. A különböző lehetséges OP kód-operandus kombinációk tárgykódban a következők lehetnek:



Az r a B, C, D, E, H, L vagy A regisztereket azonosítja, amelyek a fenti tárgykódban a következők:

# OR s

---

| <u>Regiszter</u> | <u>r</u> |    |
|------------------|----------|----|
| B                | 000      | B0 |
| C                | 001      | B1 |
| D                | 010      | B2 |
| E                | 011      | B3 |
| H                | 100      | B4 |
| L                | 101      | B5 |
| A                | 111      | B7 |

## Leírás:

Az utasítás az s operandus által specifikált byte, és az akkumulátor tartalma között bitenkénti logikai VAGY kapcsolatot hoz létre, és az eredmény az akkumulátorba kerül.

| UTASÍTÁS  | M | CIKLUSOK | T             | ÁLLAPOTOK | 4 MHz V.i. |
|-----------|---|----------|---------------|-----------|------------|
| OR r      |   | 1        | 4             |           | 1.00       |
| OR n      |   | 2        | 7/4,3/        |           | 1.75       |
| OR (HL)   |   | 2        | 7/4,3/        |           | 1.75       |
| OR (IX+d) |   | 5        | 19/4,4,3,5,3/ |           | 4.75       |
| OR (IY+d) |   | 5        | 19/4,4,3,5,3/ |           | 4.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: törlődik

## Példa:

Ha a H regiszter tartalma 48H 01001000 , és az akkumulátoré 12H (00010010) , akkor az

OR H

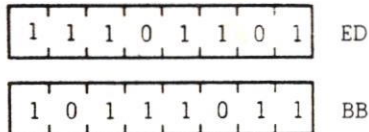
utasítás végrehajtásának eredményeként az akkumulátor tartalma 5AH (01011010) lesz.

# OTDR

Művelet: (C) ← (HL) , B ← B-1, HL ← HL-1

Formátum: OP kód

OTDR



## Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, amelynél a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során először a HL regiszterpár tartalma kerül a címbuszra. Az ilyen módon megcímzett memóriarekeszben lévő byte-ot a CPU átmenetileg tárolja, majd a byte számláló /B regiszter/ tartalma dekrementálódik. Ezután a C regiszter tartalma a címbusz alsó nyolc bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanekkor a byte számlálóként felhasználható B regiszter dekrementált tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. Ezután az előzőleg a HL tartalma által megcímzett memóriarekeszből kiolvasott, majd átmenetileg a CPU-ban tárolt byte a C regiszter tartalma által kijelölt periféria regiszterbe töltődik. Ezt követően a HL regiszterpár tartalma dekrementálódik, és ha a már korábban dekrementált B regiszter tartalom nem zérus, akkor a programszámláló tartalma 2-vel dekrementálódik, és az utasítás megismétlődik. Ha viszont B tartalma zérus, az utasítás befejeződik. Megjegyezzük, hogy ha az utasítás végrehajtását megelőzően B tartalma 0-ra lett beállítva, 256 byte adat kitöltését végzi az utasítás. Minden egyes byte-átvitelt követően két memóriafrissítési ciklus zajlik le, és ugyanakkor programmegszakítási kérelmek is érvényre juthatnak.

## Ha B≠0:

M ciklusok: 5    T állapotok: 21/4,5,3,4,5/    4 MHz V.i.: 5.25

## Ha B=0:

M ciklusok: 4    T állapotok: 16/4,5,3,4/    4 MHz V.i.: 4.00

# OTDR

---

A jelzőbit regiszter érintett bitjei:

S: ismeretlen  
Z: 1-es állapotba kerül  
E: ismeretlen  
P/V: ismeretlen  
N: 1-es állapotba kerül  
C: változatlan marad

Példa:

Ha a C regiszter tartalma 07H, a B regiszteré 03H, a HL regiszterpár 1000H-t tartalmaz, és a memóriarekeszek tartalma a következő:

| <u>Rekesz</u> | <u>Tartalom</u> |
|---------------|-----------------|
| OFFEH         | 51H             |
| OFFFH         | A9H             |
| 1000H         | 03H             |

akkor az

OTDR

utasítás végrehajtása után a HL regiszterpár tartalma OFFDH lesz, a B regiszter 0-t fog tartalmazni, és az eredetileg a memóriában lévő byte sorozat a következő sorrendben a 07H című perifériaregiszterbe töltődik:

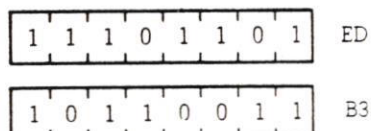
03H  
A9H  
51H

# OTIR

Művelet: (C) ← (HL) , B ← B-1, HL ← HL+1

Formátum: OP kód

OTIR



## Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, amelynél a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során először a HL regiszterpár tartalma kerül a címbuszra. Az ilyen módon megcímezett memóriarekeszben lévő byte-ot a CPU átmenetileg tárolja, majd a byte számláló /B regiszter/ tartalma dekrementálódik. Ezután a C regiszter tartalma a címbusz alsó 8 bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanekkor a byte számlálóként felhasználható B regiszter dekrementált tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. Ezután az előzőleg a HL regiszterpár tartalma által megcímezett memóriarekeszből kiolvasott, majd átmenetileg a CPU-ban tárolt byte a C regiszter tartalma által kijelölt periféria regiszterbe töltődik. Ezt követően a HL regiszterpár tartalma inkrementálódik, és ha a már korábban dekrementált B regiszter tartalom nem zérus, akkor a programszámláló tartalma 2-vel dekrementálódik, és az utasítás megismétlődik. Ha viszont B tartalma zérus, az utasítás befejeződik. Megjegyezzük, hogy ha az utasítás végrehajtását megelőzően B tartalma 0-ra lett beállítva, 256 byte adat kitöltését végzi az utasítás. Minden egyes byte-átvitelt követően két memóriafrissítési ciklus zajlik le, és ugyanekkor programmegszakítási kérelmek is érvényre juthatnak.

## Ha B≠0:

M ciklusok: 5    T állapotok: 21/4,5,3,4,5/    4 MHz V.i.: 5.25

## Ha B=0:

M ciklusok: 4    T állapotok: 16/4,5,3,4/    4 MHz V.i.: 4.00

# OTIR

---

A jelzőbit regiszter érintett bitjei:

S: ismeretlen  
Z: 1-es állapotba kerül  
H: ismeretlen  
P/V: ismeretlen  
N: 1-es állapotba kerül  
C: változatlan marad

Példa:

Ha a C regiszter tartalma 07H, a B regiszteré 03H, a HL regiszterpár 1000H-t tartalmaz, és a memóriarekeszek tartalma a következő:

| <u>Rekesz</u> | <u>Tartalom</u> |
|---------------|-----------------|
| 1000H         | 51H             |
| 1001H         | A9H             |
| 1002H         | 03H             |

akkor az

OTIR

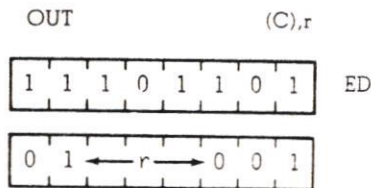
utasítás végrehajtása után a HL regiszterpár tartalma 1003H lesz, a B regiszter 0-t fog tartalmazni, és az eredetileg a memóriában lévő byte-sorozat a következő sorrendben a 07H című periféria regiszterbe töltődik:

51H  
A9H  
03H

# OUT (C),r

Művelet: (C) ← r

Formátum: OP kód Operandumok



## Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, amelynél a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során a C regiszter tartalma a címbusz alsó 8 bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanekkor a B regiszter tartalma a címbusz felső 8 bitjén /A8-tól A15-ig/ van jelen. Ezután az r regiszterben lévő byte az adatbuszra kerül, és a C regiszter tartalma által kiválasztott periféria regiszterbe töltődik. Az r regiszter kódja a CPU bármely regiszterét az alábbi táblázat szerint azonosítja:

| Regiszter | r   |     |
|-----------|-----|-----|
| B         | 000 | 49H |
| C         | 001 | 49H |
| D         | 010 | 59H |
| E         | 011 | 59H |
| H         | 100 | 69H |
| L         | 101 | 69H |
| A         | 111 | 79H |

M ciklusok: 3 T állapotok: 12/4,4,4/ 4 MHz V.i.: 3.00

## Példa:

Ha a C regiszter tartalma 01H, és a D regiszteré 5AH, akkor az

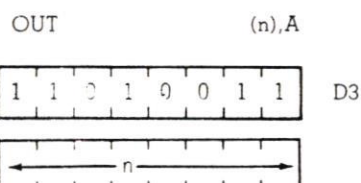
OUT (C),D

utasítás végrehajtásának eredményeként az 5AH byte a 01H című periféria regiszterbe töltődik.

# OUT (n),A

Művelet: (n) ← A

Formátum: OP kód            Operandusok



## Leírás:

A végrehajtás során az utasításnak a címbusz alacsony helyiértékű bitjeire /A0-tól A7-ig/ helyezett n operandusa a 256 lehetséges periféria regiszter egyikét választja ki. Ugyanekkor az akkumulátor /A regiszter/ tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. Az akkumulátorban lévő byte az adatbuszra kerül, és a kiválasztott periféria regiszterbe töltődik.

M ciklusok: 3    T állapotok: 11/4,3,4/    4 MHz V.i.: 2.75

## Példa:

Ha az akkumulátor tartalma 23H, az

OUT (01H),A

utasítás végrehajtásának eredményeként a 23H byte töltődik a 01H című periféria regiszterbe.

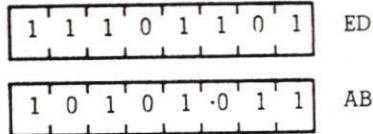


# OUTD

Művelet: (C) ← (HL), B ← B-1, HL ← HL-1

Formátum: OP kód

OUTD



## Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, amelynél a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során először a HL regiszterpár tartalma kerül a címbuszra. Az ilyen módon megcímezett memóriarekeszben lévő byte-ot a CPU átmenetileg tárolja, majd a byte számláló /B regiszter/ tartalma dekrementálódik. Ezután a C regiszter tartalma a címbusz alsó 8 bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanekkor a byte számlálóként felhasználható B regiszter dekrementált tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. Ezután az előzőleg a HL tartalma által megcímezett memóriarekeszekből kiolvasott, majd átmenetileg a CPU-ban tárolt byte a C regiszter tartalma által kijelölt periféria regiszterbe töltődik. Végül a HL regiszterpár tartalma dekrementálódik.

M ciklus: 4 T állapotok: 16/4,5,3,4/ 4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

- S: ismeretlen
- Z: 1-es állapotba kerül, ha B-1=0; egyébként törlődik
- H: ismeretlen
- P/V: ismeretlen
- N: 1-es állapotba kerül
- C: változatlan marad

## Példa:

Ha a C regiszter tartalma 07H, a B regiszteré 10H, a HL regiszterpár tartalma 1000H, és az 1000H című memóriarekesz tartalma 59H, akkor az

OUTD

utasítás végrehajtásának eredményeként a B regiszter 0FH-t, a HL regiszterpár 0FFFH-t fog tartalmazni, és az 59H töltődik a 07H című periféria regiszterbe.

# OUTI

Művelet: (C) ← (HL), B ← B-1, HL ← HL+1

Formátum: OP kód

OUTI

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 A3

## Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző I/O utasítás, amelynél a periféria regiszter címét a C regiszter tartalmazza. A végrehajtás során először a HL regiszterpár tartalma kerül a címbuszra. Az ilyen módon megcímezett memóriarekeszben lévő byte-ot a CPU átmenetileg tárolja, majd a byte számláló /B regiszter/ tartalma dekrementálódik. Ezután a C regiszter tartalma a címbusz alsó 8 bitjére /A0-tól A7-ig/ kerül, és kiválasztja a 256 lehetséges periféria regiszter egyikét. Ugyanekkor a byte számlálóként felhasználható B regiszter dekrementált tartalma a címbusz magasabb helyiértékű 8 bitjén /A8-tól A15-ig/ van jelen. Ezután az előzőleg a HL tartalma által megcímezett memóriarekeszből kiolvasott, majd átmenetileg a CPU-ban tárolt byte a C regiszter tartalma által kijelölt periféria regiszterbe töltődik. Végül a HL regiszterpár tartalma inkrementálódik.

M ciklusok: 4 T állapotok: 16/4,5,3,4/ 4 MHz V.i.: 4.00

## A jelzőbit regiszter érintett bitjei:

S: ismeretlen  
Z: 1-es állapotba kerül, ha B-1=0; egyébként törlődik  
H: ismeretlen  
P/V: ismeretlen  
N: 1-es állapotba kerül  
C: változatlan marad

## Példa:

Ha a C regiszter tartalma 07H, a B regiszteré 10H, a HL regiszterpár 1000H-t tartalmaz, és az 1000H című memóriarekesz tartalma 59H, akkor az

OUTI

utasítás végrehajtásának eredményeként a B regiszter 0FH-t, a HL regiszterpár 1001H-t fog tartalmazni, és az 59H töltődik a 07H című periféria regiszterbe.

# POP IX

Művelet:  $IX_H \leftarrow (SP+1)$ ,  $IX_L \leftarrow (SP)$

Formátum: OP kód            Operandumok

| POP             | IX |    |
|-----------------|----|----|
| 1 1 0 1 1 1 0 1 |    | DD |
| 1 1 1 0 0 0 0 1 |    | EI |

## Leírás:

Az utasítás az operatív memóriában elhelyezkedő LIFO verem-  
táron POP műveletet /a veremtár tetején elhelyezkedő, azaz  
oda utoljára betöltött 2 byte kiemelése/ végez, és ennek  
eredményét az IX index regiszterbe tölti. A veremtár aktuá-  
lis tetejére mutató címet az SP regiszterpár tartalmazza. Az  
utasítás először az SP tartalma által kijelölt memóriarekesz  
tartalmát tölti az index regiszter alacsonyabb helyiértékű  
byte-jába, majd az SP inkrementálódik, és az ilyen módon ki-  
jelölt következő memóriarekesz tartalma az IX index regisz-  
ter magasabb helyiértékű byte-jába másolódik. Ezután az SP  
ismét inkrementálódik.

M ciklusok: 4    T állapotok: 14/4,4,3,3/    4 MHz V.i.: 3.50

## Példa:

Ha a veremtár mutató /SP/ 1000H-t, az 1000H című memóriare-  
kesz 55H-t, és az 1001H című memóriarekesz 33H-t tartalmaz,  
a

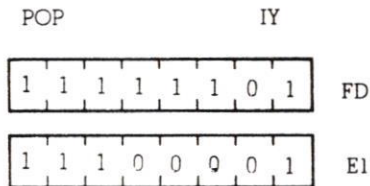
POP IX

utasítás végrehajtása után az IX index regiszter 3355H-t, és  
az SP 1002H-t fog tartalmazni.

# POP IY

Művelet:  $IY_H \leftarrow (SP+1), IY_L \leftarrow (SP)$

Formátum: OP kód          Operandusok



## Leírás:

Az utasítás az operatív memóriában elhelyezkedő LIFO verem-  
táron POP műveletet /a veremtár tetején elhelyezkedő, azaz az  
oda utoljára betöltött 2 byte kiemelése/ végez, és ennek ered-  
ményét az IY index regiszterbe tölti. A veremtár aktuális te-  
tejére mutató címet az SP regiszterpár tartalmazza. Az utasí-  
tás először az SP tartalma által kijelölt memóriarekesz tar-  
talmát tölti az index regiszter alacsonyabb helyiértékű byte-  
-jába, majd az SP inkrementálódik, és az ilyen módon kijelölt  
következő memóriarekesz tartalma az IY index regiszter maga-  
sabb helyiértékű byte-jába másolódik. Ezután az SP ismét  
inkrementálódik.

M ciklusok: 4    T állapotok: 14/4,4,3,3/    4 MHz V.i.: 3.50

## Példa:

Ha az SP regiszterpár 1000H-t, az 1000H című memóriarekesz  
55H-t, és az 1001H című memóriarekesz 33H-t tartalmaz, a

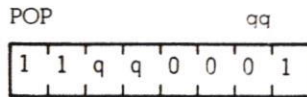
POP IY

utasítás végrehajtása után az IY index regiszter 3355H-t,  
és az SP 1002H-t fog tartalmazni.

# POP qq

Művelet:  $qq_H \leftarrow (SP+1), qq_L \leftarrow (SP)$

Formátum: OP kód Operandusok



## Leírás:

Az utasítás az operatív memóriában elhelyezkedő LIFO verem-  
táron POP műveletet végez, és ennek eredményét a qq regisz-  
terpárba tölti. A veremtár aktuális tetejére mutató címet az  
SP regiszterpár tartalmazza. Az utasítás először az SP tar-  
talma által kijelölt memóriarekesz tartalmát tölti qq ala-  
csonyabb helyiértékű byte-jába, majd az SP inkrementálódik,  
és az ilyen módon kijelölt következő memóriarekesz tartalma  
a qq regiszterpár magasabb helyiértékű byte-jába másolódik.  
Ezután az SP ismét inkrementálódik. A qq operandus a BC, DE,  
HL vagy AF regiszterpárokat definiálja, amelyek tárgykódban  
a következők:

| Regiszterpár | qq |    |                  |
|--------------|----|----|------------------|
| BC           | 00 | C1 |                  |
| DE           | 01 | D1 |                  |
| HL           | 10 | E1 |                  |
| AF           | 11 | F1 | 18080 m. POP PSW |

M ciklusok: 3 T állapotok: 10/4,3,3/ 4 MHz V.i.: 2.50

## Példa:

Ha az SP regiszterpár 1000H-t, az 1000H című memóriarekesz  
55H-t, és az 1001H című memóriarekesz 33H-t tartalmaz, a

POP HL

utasítás végrehajtásának eredményeként a HL regiszterpár tar-  
talma 3355H, az SP tartalma pedig 1002H lesz.

# PUSH IX

Művelet:  $(SP-2) \leftarrow IX_L, (SP-1) \leftarrow IX_H$

Formátum: OP kód            Operandusok

| PUSH            | IX |    |
|-----------------|----|----|
| 1 1 0 1 1 1 0 1 |    | DD |
| 1 1 1 0 0 1 0 1 |    | E5 |

## Leírás:

Az utasítás az IX index regiszter tartalmát PUSH művelettel /2 byte betöltése a veremtár tetejére/ az operatív memóriában elhelyezkedő LIFO veremtárba helyezi. A veremtár aktuális tetejére mutató címet az SP regiszterpár tartalmazza. Az utasítás először az SP tartalmát dekrementálja, és az index regiszter magasabb helyiértékű 8 bitjét az így nyert cím által kijelölt memóriarekeszbe tölti, majd az SP ismét dekrementálódik, és az index regiszter alacsonyabb helyiértékű byte-ja az SP regiszterpárban így kapott cím által definiált memóriarekeszbe másolódik.

M ciklusok: 3    T állapotok: 15/4,5,3,3/    4 MHz V.i.: 3.75

## Példa:

Ha az IX index regiszter 2233H-t, és az SP regiszterpár 1007H-t tartalmaz, a

PUSH IX

utasítás végrehajtásának eredményeként az 1006H című memóriarekesz 22H-t, az 1005H című 33H-t fog tartalmazni, és az SP tartalma 1005H lesz.

# PUSH IY

Művelet:  $(SP-2) \leftarrow IY_L, (SP-1) \leftarrow IY_H$

Formátum: OP kód      Operandusok

| PUSH            | IY |    |
|-----------------|----|----|
| 1 1 1 1 1 1 0 1 |    | FD |
| 1 1 1 0 0 1 0 1 |    | ES |

## Leírás:

Az utasítás az IY index regiszter tartalmát PUSH művelettel /2 byte betöltése a veremtár tetejére/ az operatív memóriában elhelyezkedő LIFO veremtárba helyezi. A veremtár aktuális tetejére mutató címet az SP regiszterpár tartalmazza. Az utasítás először az SP tartalmát dekrementálja, és az index regiszter magasabb helyiértékű 8 bitjét az így nyert cím által kijelölt memóriarekeszbe tölti, majd az SP ismét dekrementálódik, és az index regiszter alacsonyabb helyiértékű byte-ja az SP regiszterpárban így kapott cím által definiált memóriarekeszbe másolódik.

M ciklusok: 4      T állapotok: 15/4,5,3,3/      4 MHz V.i.: 3.75

## Példa:

Ha az IY index regiszter 2233H-t, és az SP regiszterpár 1007H-t tartalmaz, a

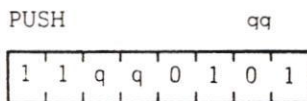
PUSH IY

utasítás végrehajtásának eredményeként az 1006H című memóriarekesz 22H-t, az 1005H című 33H-t fog tartalmazni, és az SP tartalma 1005H lesz.

# PUSH qq

Művelet:  $(SP-2) \leftarrow qq_L, (SP-1) \leftarrow qq_H$

Formátum: OP kód            Operandusok



## Leírás:

Az utasítás a qq regiszterpár tartalmát PUSH művelettel /2 byte betöltése a veremtár tetejére/ az operatív memóriában elhelyezkedő LIFO veremtárba helyezi. A veremtár aktuális tetejére mutató címet az SP regiszterpár tartalmazza. Az utasítás először az SP tartalmát dekrementálja, és a qq regiszterpár magasabb helyiértékű 8 bitjét az így nyert cím által kijelölt memóriarekeszbe tölti, majd az SP ismét dekrementálódik, és a qq regiszterpár alacsonyabb helyiértékű byte-ja az SP-ben így kapott cím által definiált memóriarekeszbe másolódik. A qq operandus a BC, DE, HL vagy AF regiszterpárokat jelenti, amelyek tárgykódban a következők:

| Regiszterpár | qq |                              |
|--------------|----|------------------------------|
| BC           | 00 | C5                           |
| DE           | 01 | D5                           |
| HL           | 10 | E5                           |
| AF           | 11 | F5 <i>ROSE memo PUSH PSW</i> |

M ciklusok: 3    T állapotok: 11/5,3,3/    4 MHz V.i.: 2.75

## Példa:

Ha az AF regiszterpár tartalma 2233H, és az SP regiszterpár 1007H-t tartalmaz, a

PUSH AF

utasítás végrehajtásának eredményeként az 1006H című memóriarekesz 22H-t, az 1005H című memóriarekesz 33H-t, és az SP regiszterpár 1005H-t fog tartalmazni.

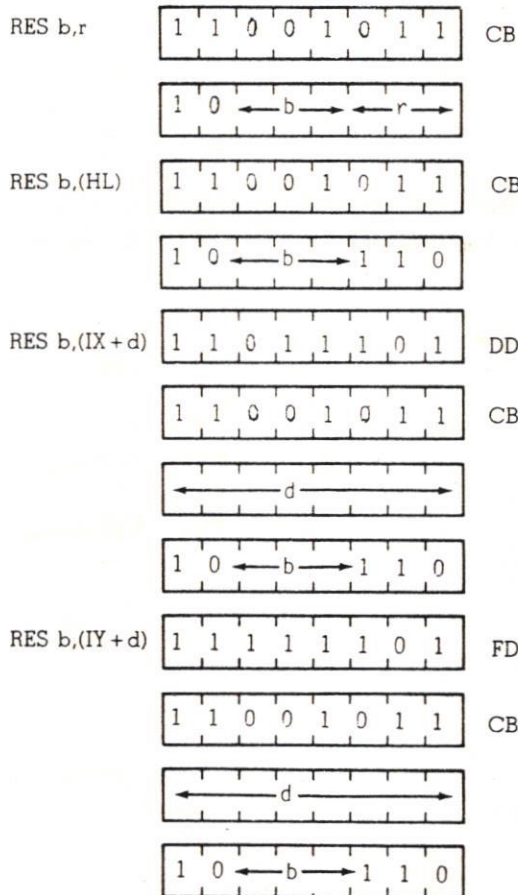


# RES b,m

Művelet:  $s_b \leftarrow 0$

Formátum:            OP kód            Operandusok  
                          RES                    b,m

A b operandus az m operandus /r, (HL), (IX+d) vagy (IY+d) bármelyike/ által meghatározott byte egy bitjét /7-től 0-ig/ definiálja, amint azt az ezzel az utasítással analóg SET utasításoknál meghatároztuk. Ezek a különböző lehetséges OP kód-operandus kombinációk a tárgyködben a következők lehetnek:



| Törölt bit | b   | Regiszter | r   |
|------------|-----|-----------|-----|
| 0          | 000 | B         | 000 |
| 1          | 001 | C         | 001 |
| 2          | 010 | D         | 010 |
| 3          | 011 | E         | 011 |
| 4          | 100 | H         | 100 |
| 5          | 101 | L         | 101 |
| 6          | 110 | L         | 110 |
| 7          | 111 | L         | 111 |

## RES b,m

---

### Leírás:

Az m operandus által meghatározott byte-nak a b operandus által meghatározott bitje törlődik.

| UTASÍTÁS   | M CIKLUSOK | T ÁLLAPOTOK     | 4 MHz V.i. |
|------------|------------|-----------------|------------|
| RES r      | 4          | 8/4,4/          | 2.00       |
| RES (HL)   | 4          | 15/4,4,4,3/     | 3.75       |
| RES (IX+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |
| RES (IY+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |

### Példa:

A

RES 6,D

utasítás végrehajtásának eredményeként a D regiszter 6-os bitje törlődik. /A D regiszter 0-ás bitje a legalacsonyabb helyiértékű bit/.

# RET

Művelet:  $PC_L \leftarrow (SP), PC_H \leftarrow (SP+1)$

Formátum: OP kód

RET

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 c9

## Leírás:

Az utasítás a vezérlést az eredeti programnak adja vissza. Végrehajtása során az előzőleg CALL utasítással a veremtár tetejére mentett eredeti programszámláló /PC/ tartalmat POP művelettel kiemeli, és visszatölti a PC-be. Végrehajtásakor először az SP tartalma által kijelölt memóriarekeszben lévő byte másolódik át a PC alacsonyabb helyiértékű 8 bitjébe, majd az SP inkrementálásával kapott memóriacímen elhelyezkedő byte töltődik a PC magasabb helyiértékű 8 bitjébe. /Az SP másodszor inkrementálódik/. A CPU következő gépi ciklusa a következő OP kódot már az ilyen módon a PC-ben jelenlévő cím által meghatározott memóriarekeszből kapja.

M ciklusok: 3    T állapotok: 10/4,3,3/    4 MHz V.i.: 2.50

## Példa:

Ha a programszámláló tartalma 3535H, az SP tartalma 2000H, a 2000H című memóriarekesz tartalma B5H, és a 2001H című memóriarekeszé 18H, akkor a

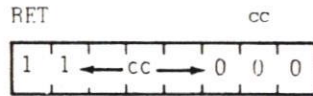
RET

utasítás végrehajtása után az SP tartalma 2002H, és a programszámláló tartalma 18B5H lesz, amely értelemszerűen a következő végrehajtandó OP kód címét jelenti.

# RET cc

Művelet: Ha cc igaz:  $PC_L \leftarrow (SP)$ ,  $PC_H \leftarrow (SP+1)$

Formátum: OP kód            Operandusok



## Leírás:

Ha a cc feltétel teljesül, az utasítás a vezérlést az eredeti programnak adja vissza. Végrehajtása során az előzőleg CALL utasítással a veremtár tetejére mentett eredeti programszám-láló /PC/ tartalmát POP művelettel kiemeli, és visszatölti a PC-be. Végrehajtásakor először az SP tartalma által kijelölt memóriarekeszben lévő byte másolódik át a PC alacsonyabb helyiértékű 8 bitjébe, majd az SP inkrementálásával kapott memóriacímen elhelyezkedő byte töltődik a PC magasabb helyi-értékű 8 bitjébe. /Az SP másodszor inkrementálódik/. A CPU következő gépi ciklusa a következő OP kódot már az ilyen módon a PC-ben jelenlévő cím által meghatározott memóriarekesz-ből kapja. Ha a cc feltétel nem teljesül, a PC a szokott módon inkrementálódik, és a program a sorrendben következő uta-sítás végrehajtásával folytatódik. A cc feltétel programozá-sa az F regiszter valamelyik bitjének kijelölésével történ-het. Az ilyen módon kijelölhető 8 feltételt az alábbi táblá-zat definiálja:

|    | <u>cc</u> | <u>Feltétel</u>     | <u>Aktuális bit</u> |     |
|----|-----------|---------------------|---------------------|-----|
| CC | 000       | NZ nem zérus        | Z                   | RNZ |
| CS | 001       | Z zérus             | Z                   | RZ  |
| DC | 010       | NC nincs átvitel    | C                   | RNC |
| DS | 011       | C átvitel           | C                   | RC  |
| ES | 100       | PO páratlan paritás | P/V                 | RPO |
| ES | 101       | PE páros paritás    | P/V                 | RPE |
| FS | 110       | P pozitív előjel    | S                   | RP  |
| FS | 111       | M negatív előjel    | S                   | RM  |

## Ha cc teljesül:

M ciklusok: 3    T állapotok: 11/5,3,3/    4 MHz V.i.: 2.75

## Ha cc nem teljesül:

M ciklusok: 1    T állapotok: 5    4 MHz V.i.: 1.25

## RET cc

---

### Példa:

Ha az S bit 1-es állapotban van, a programszámláló tartalma 3535H, az SP tartalma 2000H, a 2000H című memóriarekeszben lévő byte B5H, és a 2001H című memóriarekesz tartalma 18H, akkor a

RET M

utasítás végrehajtásának eredményeként az SP tartalma 2002H lesz, a programszámláló tartalma pedig 18B5H, amely értelem-szerűen a következő végrehajtandó OP kód címét jelenti.

# RETI

Művelet: Visszatérés programmegszakításból

Formátum: OP kód

RETI

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 4D

Leírás:

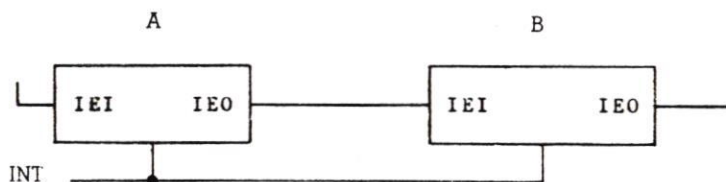
Az utasítást megszakítást kiszolgáló rutin végén használjuk azért, hogy

1. a programszámláló /PC/ eredeti tartalmát /a RET utasítás-hoz hasonlóan/ visszaállítsuk
2. egy I/O eszköz számára jelezzük, hogy a programmegszakítást kiszolgáló rutin befejeződött. A RETI utasítás lehetővé teszi a programmegszakítások egymásba ágyazását, biztosítva, hogy a magasabb prioritású programmegszakítási kérések felfüggesztik az alacsonyabb prioritású kiszolgáló rutinok futását. Az IFF2 tartalma visszatöltődik az IFF1 megszakítás engedélyező flip-flopba.

M ciklusok: 4 T állapotok: 14/4,4,3,3/ 4 MHz V.i.: 3.50

Példa:

Adott két megszakítást okozó eszköz, A és B, amelyek úgy vannak megszakítási láncba kapcsolva, hogy A-nak nagyobb a prioritása.



B programmegszakítást generál, ami elfogadásra is kerül. /B-nek IEO engedélyező kimenete törlődik, és ezáltal mindaddig, amíg B kiszolgálása tart, az összes alacsonyabb prioritású megszakításkérés blokkolódik/. Ezután A kér programmegszakítást, ezáltal felfüggesztve B kiszolgálását. /A-nak az IEO programmegszakítást engedélyező kimenete törlődik, jelezve, hogy egy magasabb prioritású eszköz kiszolgálása van folyamatban/. Az A rutin befejezésekor végrehajtott RETI utasítás ismét 1-es állapotba helyezi A-nak a programmegszakítást engedélyező IEO kimenetét, és így lehetővé teszi, hogy a B-t kiszolgáló rutin folytatódjék. A B rutin befejezésekor kiadott újabb RETI utasítás a B eszköz megszakítást engedélyező IEO kimenetét is 1-es állapotba teszi, és ilyen módon biztosítja az alacsonyabb prioritású programmegszakítások érvényre jutását is.

# RETN

Művelet: Visszatérés nem maszkolható programmegszakításból

Formátum: OP kód

RETN

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 45

## Leírás:

A nem maszkolható programmegszakítást kiszolgáló rutinok végén használt utasítás. Olyan feltétel nélküli vezérlésátadást eredményez, amely funkciójában a RET utasítással azonos. A programszámláló /PC/ korábbi veremtárba mentett tartalma POP művelettel kiemelődik, és a PC-be kerül vissza. Az SP tartalma által kijelölt memóriarekeszben lévő byte a PC alacsonyabb helyiértékű 8 bitjébe töltődik, majd az SP inkrementálásával kijelölt memóriarekesz 8 bitje a PC magasabb helyiértékű byte-jába másolódik, és az SP ismét inkrementálódik. A vezérlés ilyen módon visszaadódik az eredeti programhoz, és a CPU következő gépi ciklusa a következő OP kódot már a PC által kijelölt memóriarekeszből fogja kiemelni. A végrehajtás során IFF2 tartalma visszatöltődik IFF1-be, és így az NMI /nem maszkolható megszakítás/ elfogadását megelőző állapotot veszi fel.

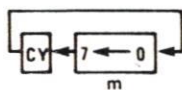
M ciklusok: 4    T állapotok: 14/4,4,3,3/    4 MHz V.i.: 3.50

## Példa:

Ha egy nem maszkolható megszakítás /NMI/ vételekor az SP tartalma 1000H, és a programszámláló 1A45H-t tartalmaz, a CPU a következő utasítást figyelmen kívül hagyja, és helyette egy, a 0066H címre irányuló „restart” műveletet hajt végre. Ekkor a programszámláló 1A45H-s aktuális tartalma a külső veremtár OFFFH és OFFEH című rekeszeibe kerül. Az első veremtár a PC magasabb helyiértékű byte-ját, míg a második az alacsonyabb helyiértékű 8 bitet tartalmazza, és a programszámlálóba a 0066H cím töltődik. Ezen a címen egy olyan megszakítás kiszolgáló rutin kezdődik, amely RETN utasítással fejeződik be. A RETN végrehajtásakor az eredeti programszámláló tartalom a veremtárból kiemelődik /elsőként az alacsonyabb helyiértékű byte/, és ezáltal az SP tartalma újra az eredeti 1000H értéket veszi fel. A program futása ott folytatódik, ahol abba maradt, vagyis az 1A45H címen elhelyezett OP kód kiemelésével.

# RL m

Művelet:



Formatum:

OP kód

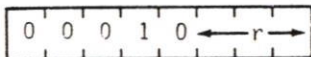
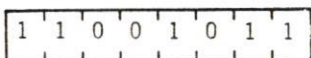
Operandusok

RL

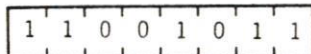
m

Az m operandus r, (HL), (IX+d) vagy (IY+d) bármelyike lehet, hasonlóan, mint az RLC utasításoknál. A lehetséges OP kód-operandus kombinációk tárgykódban a következők:

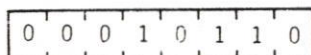
RL r CB



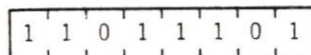
RL (HL) CB



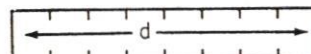
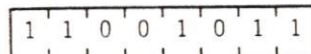
16



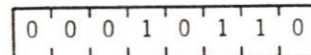
RL (IX+d) DD



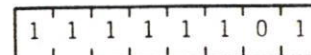
CB



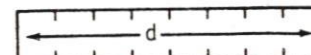
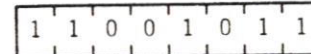
16



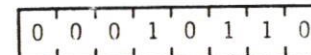
RL (IY+d) FD



CB



16



Az r a P, C, D, E, H, L vagy A regisztereket azonosítja, amelyek tárgykódban a következők:

Regiszter:

| Regiszter | r   |    |
|-----------|-----|----|
| B         | 000 | 10 |
| C         | 001 | 11 |
| D         | 010 | 12 |
| E         | 011 | 13 |
| H         | 100 | 14 |
| L         | 101 | 15 |
| A         | 111 | 17 |



# RL m

## Leírás:

Az m operandus tartalma a CY átvitel biten /a jelzőbit regiszter C bitje/ keresztül egy helyiértékkel balra tolódik. A 0-ás bit tartalma az 1-es bitbe másolódik; az 1-es bit előző tartalma pedig a 2-es bitbe, stb. A 7-es bit az átvitel bitbe, és az átvitel bit előző tartalma pedig a 0-ás bitbe másolódik. /A 0-ás bit az m operandus legkisebb helyiértékű bitje/.

| UTASÍTÁS  | M CIKLUSOK | T ÁLLAPOTOK     | 4 MHz V.i. |
|-----------|------------|-----------------|------------|
| RL r      | 2          | 8/4,4/          | 2.00       |
| RL (HL)   | 4          | 15/4,4,4,3/     | 3.75       |
| RL (IX+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |
| RL (IY+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: az m operandus által meghatározott byte 7-es bitjének tartalmát veszi fel

## Példa:

Ha a D regiszter, és az átvitel bit tartalma

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| C | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

az

RL D

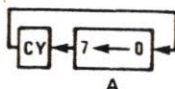
utasítás végrehajtásának eredményeként a D regiszter, és az átvitel bit tartalma a következő lesz:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| C | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |

# RLA

18080 RAL

Művelet:



Formátum: OP kód

RLA

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 17

Leírás:

Az akkumulátor /A regiszter/ tartalma a CY átvitel biten /C bit/ keresztül egy helyiértékkal balra tolódik. A 0-ás bit tartalma az 1-es bitbe kerül, az 1-es bit előző tartalma pedig a 2-es bitbe, stb. A 7-es bit az átvitel bitbe, az átvitel bit előző tartalma pedig a 0-ás bitbe másolódik. A 0-ás bit a legalacsonyabb helyiértékű bit.

M ciklusok: 1 T állapotok: 4 4 MHz V.i.: 1.00

A jelzőbit regiszter érintett bitjei:

S: változatlan marad  
Z: változatlan marad  
H: törlődik  
P/V: változatlan marad  
N: törlődik  
C: az akkumulátor 7-es bitjének tartalmát veszi fel

Példa:

Ha az akkumulátor, és az átvitel bit tartalma

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| C | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |

akkor az

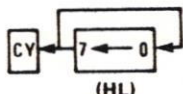
RLA

utasítás végrehajtásának eredményeként az akkumulátor, és az átvitel bit tartalma a következő lesz:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| C | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |

## RLC (HL)

Művelet:



Formátum:            OP kód            Operandusok

                         RLC                            (HL)

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 CB

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|

 06

Leírás:

A HL regiszterpár tartalma által specifikált memóriarekesz tartalma a CY átvitel bit /C bit/ megkerülésével egy helyiértékkel balra tolódik. A 0-ás bit tartalma az 1-es bitbe kerül, az 1-es bit előző tartalma pedig a 2-es bitbe, stb. A 7-es bit tartalma az átvitel bitbe, és a 0-ás bitbe másolódik. A 0-ás bit a byte legkisebb helyiértékű bitje.

M ciklusok: 4    T állapotok: 15/4,4,4,3/    4 MHz V.i.: 3.75

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: a (HL) operandus által meghatározott byte 7-es bitjének tartalmát veszi fel

Példa:

Ha a HL regiszterpár tartalma 2828H, és a 2828H című memóriarekesz tartalma

## RLC (HL)

---

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

akkor az

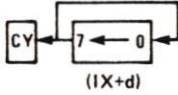
RLC (HL)

utasítás végrehajtásának eredményeként a 2828H című memóriarekesz, és az átvitel bit tartalma a következő lesz:

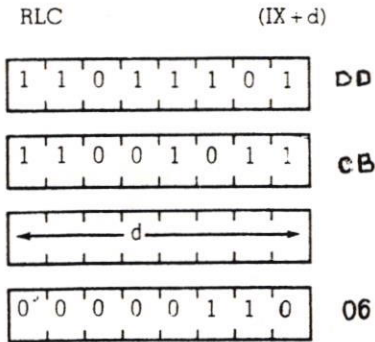
|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| C | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

# RLC (IX+d)

Művelet:



Formátum:            OP kód            Operandumok



Leírás:

Az IX index regiszter tartalma, és a 2-es komplementű kódban értelmezett d egész szám összege által specifikált memóriarekesz tartalma a CY átvitel bit /C bit/ megkerülésével egy helyiértékkel balra tolódik. A 0-ás bit tartalma az 1-es bitbe kerül, az 1-es bit előző tartalma pedig a 2-es bitbe, stb. A 7-es bit tartalma az átvitel bitbe, és a 0-ás bitbe másolódik. A 0-ás bit a byte legkisebb helyiértékű bitje.

M ciklusok:6    T állapotok: 23/4,4,3,5,4,3/    4 MHz V.i.: 5.75

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: az (IX+d) által kijelölt byte 7-es bitjének tartalmát veszi fel

## RLC (IX+d)

---

Példa:

Ha az IX index regiszter tartalma 1000H, és az 1002H című memóriarekesz tartalma

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

akkor az

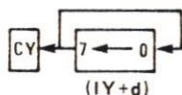
RLC (IX+2H)

utasítás végrehajtása után az 1002H című memóriarekesz, és az átvitel bit tartalma a következő lesz:

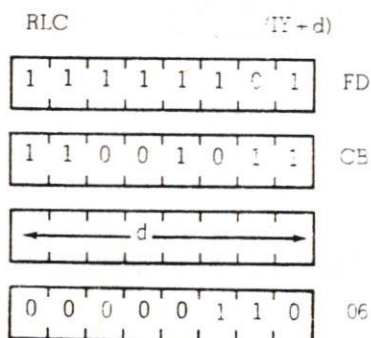
|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| C | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

## RLC (IY+d)

Művelet:



Formátum: OP kód      Operandusok



Leírás:

Az IY index regiszter tartalma, és a 2-es komplementű kódban értelmezett d egész szám összege által specifikált memóriarekesz tartalma a CY átvitel bit /C bit/ megkerülésével egy helyiértékkel balra tolódik. A 0-ás bit tartalma az 1-es bitbe kerül, az 1-es bit előző tartalma pedig a 2-es bitbe, stb. A 7-es bit tartalma az átvitel bitbe, és a 0-ás bitbe másolódik. A 0-ás bit a byte legkisebb helyiértékű bitje.

M ciklusok: 6    T állapotok: 23/4,4,3,5,4,3/    4 MHz V.i.: 5.75

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik

## RLC (IY+d)

---

C: az (IY+d) által kijelölt byte 7-es bitjének tartalmát veszi fel

Példa:

Ha az IY index regiszter tartalma 1000H, és az 1002H című memóriarekesz tartalma

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

akkor az

RLC (IY+2H)

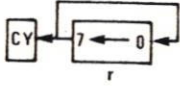
utasítás végrehajtásának eredményeként az 1002H című memóriarekesz, és az átvitel bit tartalma a következő lesz:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| C | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

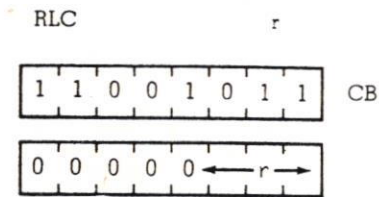


# RLC r

Művelet:



Formátum: OP kód Operandumok



Leírás:

Az r operandus által meghatározott regiszter tartalma a CY átvitel bit megkerülésével egy helyiértékkel balra tolódik. A 0-ás bit tartalma az 1-es bitbe kerül, az 1-es bit előző tartalma pedig a 2-es bitbe, stb. A 7-es bit tartalma az átvitel bitbe, és a 0-ás bitbe kerül. Az r operandus tárgykódban a következő lehet:

| <u>Regiszter</u> | <u>r</u> |    |
|------------------|----------|----|
| B                | 000      | 04 |
| C                | 001      | 01 |
| D                | 010      | 02 |
| E                | 011      | 03 |
| H                | 100      | 04 |
| L                | 101      | 05 |
| A                | 111      | 07 |

Megjegyzés: A 0-ás bit a byte legkisebb helyiértékű bitje.

M ciklusok: 2    T állapotok: 8/4,4/    4 MHz V.i.: 2.00

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetében; egyébként törlődik
- N: törlődik

# RLC r

---

C: az r által specifikált regiszter 7-es bitjének tartalmát veszi fel

Példa:

Ha az r regiszter tartalma

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

akkor az

RLC r

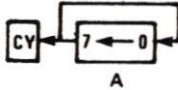
utasítás végrehajtásának eredményeként az r regiszter, és az átvitel bit tartalma a következő lesz:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| C | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

# RLCA

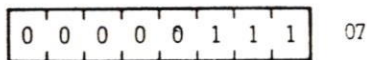
18040 RLC

Művelet:



Formátum: OP kód

RLCA



Leírás:

Az akkumulátor /A regiszter/ tartalma a CY átvitel bit megkezdésével egy helyiértékkel balra tolódik. A 0-ás bit tartalma az 1-es bitbe kerül, az 1-es bit előző tartalma a 2-es bitbe, stb. A 7-es bit tartalma az átvitel bitbe, és a 0-ás bitbe másolódik. /A 0-ás bit a byte legkisebb helyiértékű bitje/.

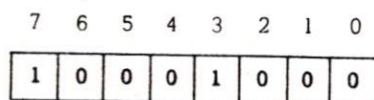
M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

A jelzőbit regiszter érintett bitjei:

- S: változatlan marad
- Z: változatlan marad
- H: törlődik
- P/V: változatlan marad
- N: törlődik
- C: az akkumulátor 7-es bitjének tartalmát veszi fel

Példa:

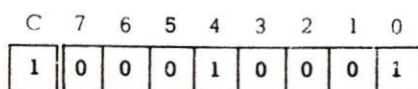
Ha az akkumulátor tartalma



akkor az

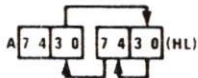
RLCA

utasítás végrehajtásának eredményeként az akkumulátor, és az átvitel bit tartalma a következő lesz:



# RLD

Művelet:



Formátum: OP kód

RLD

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|

 ED

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 6F

Leírás:

A HL tartalma által specifikált memóriarekesz négy alacsonyabb helyiértékű bitjének /3, 2, 1 és 0 bit/ tartalma ugyanezen byte négy magasabb helyiértékű /7, 6, 5 és 4/ bitjébe töltődik. A magasabb helyiértékű négy bit eredeti tartalma az akkumulátor /A regiszter/ alacsonyabb helyiértékű négy bitjébe töltődik. Az akkumulátor alacsonyabb helyiértékű négy bitjének eredeti tartalma viszont a HL tartalma által specifikált memóriarekesz alacsonyabb helyiértékű négy bitjébe kerül. Az akkumulátor magasabb helyiértékű négy bitjének tartalma változatlan marad.

M ciklusok: 5    T állapotok: 18/4,4,3,4,3/    4 MHz V.i.: 4.50

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha a művelet után az akkumulátor tartalma negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha a művelet után az akkumulátor tartalma nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül, ha az akkumulátor tartalma páros paritású a művelet után; egyébként törlődik
- N: törlődik
- C: változatlan marad

# RLD

Példa:

Ha a HL regiszterpár tartalma 5000H, az akkumulátor, és az 5000H című memóriarekesz tartalma pedig

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |

Akkumulátor

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |

(5000H)

akkor az

RLD

utasítás végrehajtásának eredményeként az akkumulátor, és az 5000H című memóriarekesz tartalma a következő lesz:

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |

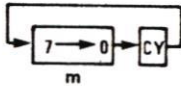
Akkumulátor

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |

(5000H)

# RR m

Művelet:



Formátum:

OP kód

Operandusok

RR

m

Az m operandus r, (HL), (IX+d) vagy (IY+d) bármelyike lehet, úgymint az analóg RLC utasításoknál. A különböző lehetséges OP kód-operandus kombinációk tárgykódban a következők lehetnek:

|           |                 |    |
|-----------|-----------------|----|
| RR r      | 1 1 0 0 1 0 1 1 | CB |
|           | 0 0 0 1 1 ← r → |    |
| RR (HL)   | 1 1 0 0 1 0 1 1 | CB |
|           | 0 0 0 1 1 1 1 0 | 1E |
| RR (IX+d) | 1 1 0 1 1 1 0 1 | DD |
|           | 1 1 0 0 1 0 1 1 | CB |
|           | ← d →           |    |
|           | 0 0 0 1 1 1 1 0 | 1E |
| RR (IY+d) | 0 0 0 1 1 1 1 0 | 1E |
|           | 1 1 0 0 1 0 1 1 | CB |
|           | ← d →           |    |
|           | 0 0 0 1 1 1 1 0 | 1E |

Az r a B, C, D, E, H, L vagy A regisztereket azonosítja a következő módon:

# RR m

| Regiszter | r   |    |
|-----------|-----|----|
| B         | 000 | 18 |
| C         | 001 | 19 |
| D         | 010 | 1A |
| E         | 011 | 1B |
| H         | 100 | 1C |
| L         | 101 | 1D |
| A         | 111 | 1F |

## Leírás:

Az m operandus tartalma a CY átvitel biten keresztül egy helyiértékkel jobbra tolódik. A 7-es bit tartalma a 6-os bitbe kerül, a 6-os bit előző tartalma pedig az 5-ös bitbe, stb. A 0-ás bit tartalma az átvitel bitbe, az átvitel bit előző tartalma pedig a 7-es bitbe kerül. A 0-ás bit a byte legkisebb helyiértékű bitje.

| UTASÍTÁS  | M CIKLUSOK | T ÁLLAPOTOK     | 4 MHz V.i. |
|-----------|------------|-----------------|------------|
| RR r      | 2          | 8/4,4/          | 2.00       |
| RR (HL)   | 4          | 15/4,4,4,3/     | 3.75       |
| RR (IX+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |
| RR (IY+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetében; egyébként törlődik
- N: törlődik
- C: az m operandus által meghatározott byte 0-ás bitjének tartalmát veszi fel

## Példa:

Ha a HL regiszterpár tartalma 4343H, és a 4343H című memóriarekesz, valamint az átvitel bit tartalma

7 6 5 4 3 2 1 0 C

akkor az

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|

RR (HL)

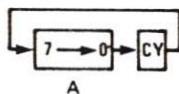
utasítás végrehajtásának eredményeként a 4343H című memóriarekesz, és az átvitel bit tartalma a következő lesz:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | C |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |

# RRA

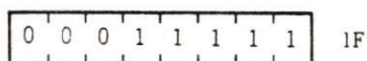
18080 RAR

Művelet:



Formátum: OP kód

RRA



Leírás:

Az akkumulátor /A regiszter/ tartalma a CY átvitel biten keresztül egy helyiértékkel jobbra tolódik. A 7-es bit tartalma a 6-os bitbe kerül, a 6-os bit előző tartalma az 5-ös bitbe, stb. A 0-ás bit tartalma az átvitel bitbe, az átvitel bit előző tartalma pedig a 7-es bitbe kerül. A 0-ás bit a byte legkisebb helyiértékű bitje.

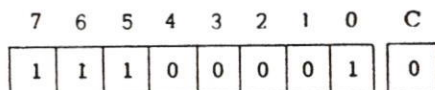
M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

A jelzőbit regiszter érintett bitjei:

S: változatlan marad  
Z: változatlan marad  
H: törlődik  
P/V: változatlan marad  
N: törlődik  
C: az akkumulátor 0-ás bitjének tartalmát veszi fel

Példa:

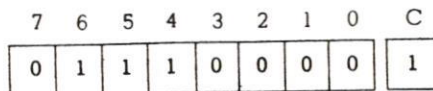
Ha az akkumulátor, és az átvitel bit tartalma



akkor az

RAA

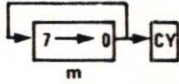
utasítás végrehajtásának eredményeként az akkumulátor, és az átvitel bit tartalma a következő lesz:





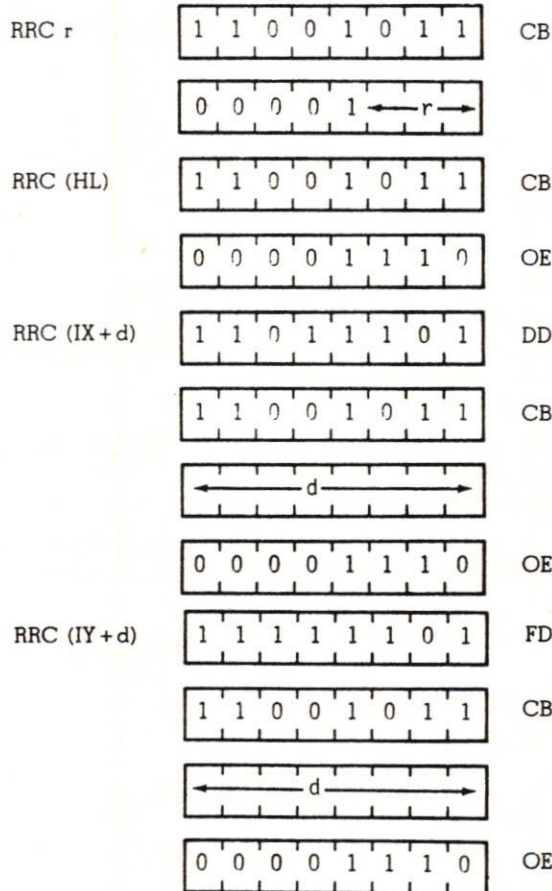
# RRC m

Művelet:



Formátum:      OP kód              Operandusok  
                          RRC                              m

Az m operandus r, (HL), (IX+d) vagy (IY+d) bármelyike lehet, úgy, mint az RLC utasításoknál. A különböző OP kód-operandus kombinációk a következők lehetnek:



Az r a B, C, D, E, H, L vagy A regisztereket azonosítja az alábbi módon:

| Regiszter | r   |    |
|-----------|-----|----|
| B         | 000 | 08 |
| C         | 001 | 09 |

# RRC m

| <u>Regiszter</u> | <u>r</u>      |
|------------------|---------------|
| D                | 010 <i>QA</i> |
| E                | 011 <i>QB</i> |
| H                | 100 <i>QC</i> |
| L                | 101 <i>QD</i> |
| A                | 111 <i>QF</i> |

## Leírás:

Az m operandus tartalma a CY átvitel bit megkerülésével egy helyiértékkel jobbra tolódik. A 7-es bit tartalma a 6-os bitbe kerül, a 6-os bit előző tartalma pedig az 5-ös bitbe, stb. A 0-ás bit tartalma az átvitel bitbe, és a 7-es bitbe kerül. A 0-ás bit a byte legkisebb helyiértékű bitje.

| UTASÍTÁS   | M CIKLUSOK | T ÁLLAPOTOK     | 4 MHz V.i. |
|------------|------------|-----------------|------------|
| RRC r      | 2          | 8/4,4/          | 2.00       |
| RRC (HL)   | 4          | 15/4,4,4,3/     | 3.75       |
| RRC (IX+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |
| RRC (IY+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: az m operandus által meghatározott byte 0-ás bitjének tartalmát veszi fel

## Példa:

Ha az A regiszter tartalma

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |

akkor az

RRC A

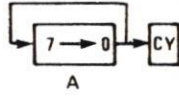
utasítás végrehajtásának eredményeként az A regiszter, és az átvitel bit tartalma a következő lesz:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | C |
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |

# RRCA

18080 RRC

Művelet:



Formátum: OP kód

RRCA

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 OP

Leírás:

Az akkumulátor /A regiszter/ tartalma a CY átvitel bit megke-  
rülésével egy helyiértékkel jobbra tolódik. A 7-es bit tartal-  
ma a 6-os bitbe kerül, a 6-os bit előző tartalma pedig az 5-ös  
bitbe, stb. A 0-ás bit tartalma az átvitel bitbe, és a 7-es  
bitbe kerül. A 0-ás bit a byte legkisebb helyiértékű bitje.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

A jelzőbit regiszter érintett bitjei:

S: változatlan marad  
Z: változatlan marad  
H: törlődik  
P/V: változatlan marad  
N: törlődik  
C: az akkumulátor 0-ás bitjének tartalmát veszi  
fel

Példa:

Ha az akkumulátor tartalma

7 6 5 4 3 2 1 0

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

akkor az

RRCA

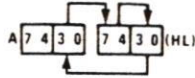
utasítás végrehajtásának eredményeként az akkumulátor, és az  
átvitel bit tartalma a következő lesz:

7 6 5 4 3 2 1 0 C

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|---|

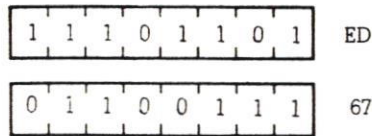
# RRD

Művelet:



Formátum: OP kód

RRD



Leírás:

A HL tartalma által specifikált memóriarekesz négy alacsonyabb helyiértékű bitjének /3, 2, 1 és 0 bit/ tartalma az akkumulátor /A regiszter/ alacsonyabb helyiértékű négy bitjébe kerül. Az akkumulátor regiszter alacsonyabb helyiértékű négy bitjének eredeti tartalma a HL tartalma által specifikált memóriarekesz négy magasabb helyiértékű bitjébe kerül át, míg ugyanezen négy bit eredeti tartalma ennek a byte-nak a négy alacsonyabb helyiértékű bitjébe töltődik át. Az akkumulátor magasabb helyiértékű négy bitjének tartalma érintetlen marad.

M ciklusok: 5    T állapotok: 18/4,4,3,4,3/    4 MHz V.i.: 4.50

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha a művelet után az akkumulátor tartalma negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha a művelet után az akkumulátor tartalma nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül, ha a művelet után az akkumulátor tartalma páros paritású; egyébként törlődik
- N: törlődik
- C: változatlan marad

Példa:

Ha a HL regiszterpár tartalma 5000H, és az akkumulátor, valamint az 5000H című memóriarekesz tartalma

# RRD

---

7 6 5 4 3 2 1 0  
1 0 0 0 0 1 0 0      Accumulator

7 6 5 4 3 2 1 0  
0 0 1 0 0 0 0 0      (5000H)

akkor az

RRD

utasítás végrehajtása után az akkumulátor, és az 5000H című memóriarekesz tartalma a következő lesz:

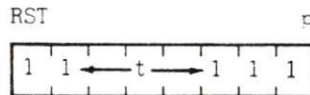
7 6 5 4 3 2 1 0  
1 0 0 0 0 0 0 0      Accumulator

7 6 5 4 3 2 1 0  
0 1 0 0 0 0 1 0      (5000H)

# RST p

Művelet:  $(SP-1) \leftarrow PC_H$ ,  $(SP-2) \leftarrow PC_L$ ,  $PC_H \leftarrow 0$ ,  $PC_L \leftarrow P$

Formátum: OP kód                      Operandusok



## Leírás:

A programszámláló /PC/ aktuális tartalma az operatív memóriában elhelyezkedő veremtárba kerül, majd a PC-be a p operandus által meghatározott, az operatív memória 0-ás lapján /legalsó 256 byte-ján/ lévő 8 speciális rendeltetésű memóriarekesz címének egyike töltődik.

A program végrehajtása ezután a PC megváltoztatott tartalma által specifikált OP kód kiemelésével folytatódik. A művelet végrehajtása során először az SP tartalma dekrementálódik, és a PC magasabb helyiértékű byte-ja az így nyert cím által kijelölt memóriarekeszbe töltődik. Ezt az SP ismételt dekrementálása, majd a PC alacsonyabb helyiértékű byte-jának az új SP tartalommal megcímezett memóriarekeszbe másolása követi. A „ReStart” utasítás az alábbi táblázatban lévő 8 cím egyikére történő ugrást biztosítja. A p operandus a táblázat megfelelő t kódjával együtt alakul tárgykóddá. Megjegyzés: miután az érintett címek mindegyike a memória 0-ás lapján helyezkedik el, az utasítás végrehajtása során a PC magasabb helyiértékű byte-ja értelemszerűen OOH értéket vesz fel, míg a PC alacsonyabb helyiértékű 8 bitjébe az alábbi táblázat p oszlopának megfelelő eleme másolódik:

| p   | t  |               |
|-----|----|---------------|
| 00H | C7 | 000      RST0 |
| 08H | CF | 001      RST1 |
| 10H | D7 | 010      RST2 |
| 18H | DF | 011      RST3 |
| 20H | E7 | 100      RST4 |
| 28H | EF | 101      RST5 |
| 30H | F7 | 110      RST6 |
| 38H | FF | 111      RST7 |

M ciklusok: 3      T állapotok: 11/5,3,3/      4 MHz V.i.: 2.75

## Példa:

Ha a programszámláló tartalma 15B3H, akkor az

RST 18H /1101111 tárgykód/

utasítás végrehajtásának eredményeként a PC a következő OP kódra mutató 0018H címet fogja tartalmazni.

# SBC A,s

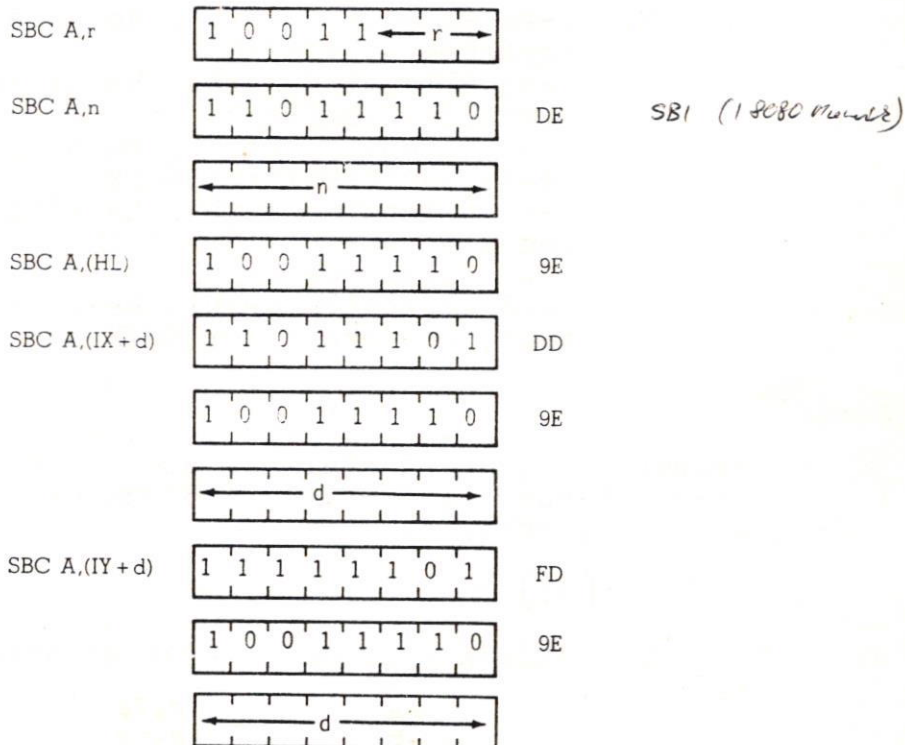
18080 SBB<sub>2</sub>

Művelet:  $A \leftarrow A - s - CY$

Formátum: OP kód            Operandusok

SBC                            A,s

Az s operandus r, n, (HL), (IX+d) vagy (IY+d) bármelyike lehet, úgy, mint az ADD utasításoknál. A különböző OP kód-operandus kombinációk a következők lehetnek:



Az r a B, C, D, E, H, L vagy A regisztereket azonosítja az előbbi módon:

| <u>Regiszter</u> | <u>r</u> |    |
|------------------|----------|----|
| B                | 000      | 98 |
| C                | 001      | 99 |
| D                | 010      | 9A |
| E                | 011      | 9B |
| H                | 100      | 9C |
| L                | 101      | 9D |
| A                | 111      | 9F |

## Leírás:

Az s operandus az átvitel bittel együtt kivonódik az akkumulátor tartalmából, és az eredmény az akkumulátorba kerül.

## SBC A,s

---

| UTASÍTÁS     | M CIKLUSOK | T ÁLLAPOTOK   | 4 MHz V.i. |
|--------------|------------|---------------|------------|
| SBC A,r      | 1          | 4             | 1.00       |
| SBC A,n      | 2          | 7/4,3/        | 1.75       |
| SBC A,(HL)   | 2          | 7/4,3/        | 1.75       |
| SBC A,(IX+d) | 5          | 19/4,4,3,5,3/ | 4.75       |
| SBC A,(IY+d) | 5          | 19/4,4,3,5,3/ | 4.75       |

### A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsordulás áll fenn; egyébként törlődik
- N: 1-es állapotba kerül
- C: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik

### Példa:

Ha az akkumulátor 16H-t tartalmaz, az átvitel bit tartalma 1, a HL regiszterpár 3433H-t, és a 3433H című memóriarekesz 05H-t tartalmaz, akkor az

SBC A,(HL)

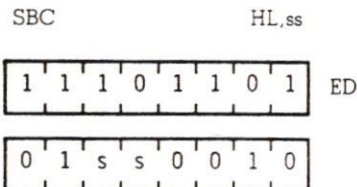
utasítás végrehajtásának eredményeként az akkumulátor 10H-t fog tartalmazni.



# SBC HL,ss

Művelet: HL ← HL-ss-CY

Formátum: OP kód Operandumok



Leírás:

Az ss regiszterpár tartalma /a BC, DE, HL vagy SP regiszterpárok bármelyike/ az átvitel bittel együtt kivonódik a HL regiszterpár tartalmából, és az eredmény a HL-ben tárolódik. Az ss operandus tárgykódban a következő lehet:

| <u>Regiszterpár</u> | <u>ss</u> |    |
|---------------------|-----------|----|
| BC                  | 00        | 42 |
| DE                  | 01        | 52 |
| HL                  | 10        | 62 |
| SP                  | 11        | 72 |

M ciklusok: 4 T állapotok: 15/4,4,4,3/ 4 MHz V.i.: 3.75

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsordulás áll fenn; egyébként törlődik
- N: 1-es állapotba kerül
- C: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik

Példa:

Ha a HL regiszterpár tartalma 9999H, a DE regiszterpáré 1111H, és az átvitel bit tartalma 1, akkor az

SBC HL,DE

utasítás végrehajtásának eredményeként a HL tartalma 8887H lesz.

# SCF

18080 STC

---

Művelet: CY ← 1

Formátum: OP kód  
SCF

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|

 37

Leírás:

A jelzőbit regiszter C bitje 1-es állapotba kerül.

M ciklusok: 1      T állapotok: 4      4 MHz V.i.: 1.00

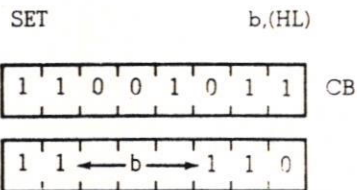
A jelzőbit regiszter érintett bitjei:

S: változatlan marad  
Z: változatlan marad  
H: törlődik  
P/V: változatlan marad  
N: törlődik  
C: 1-es állapotba kerül

# SET b,(HL)

Művelet:  $(HL)_b \leftarrow 1$

Formátum: OP kód            Operandumok



## Leírás:

A HL regiszterpár tartalma által megcímzett memóriarekesz b operandus által kijelölt bitje /0-tól 7-es bitig bármelyik/ 1-be lesz beállítva. A b operandus tárgykódban a következő lehet:

| <u>1-be állított bit</u> | <u>b</u> |
|--------------------------|----------|
| 0                        | 000      |
| 1                        | 001      |
| 2                        | 010      |
| 3                        | 011      |
| 4                        | 100      |
| 5                        | 101      |
| 6                        | 110      |
| 7                        | 111      |

M ciklusok: 4    T állapotok: 15/4,4,4,3/    4 MHz V.i.: 3.75

## Példa:

Ha a HL regiszterpár tartalma 3000H, akkor a

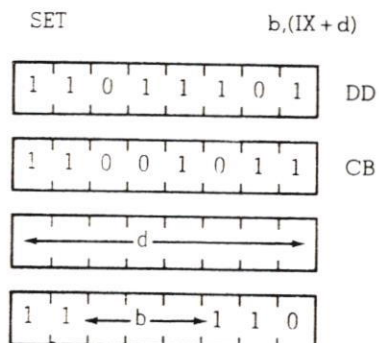
SET 4,(HL)

utasítás végrehajtása után a 3000H című memóriarekeszben lévő 4-es bit 1-es tartalmat vesz fel. /A 3000H című memóriarekeszben a 0-ás bit a legkisebb helyiértékű bit/.

# SET b,(IX+d)

Művelet:  $(IX+d)_b \leftarrow 1$

Formátum: OP kód            Operandumok



## Leírás:

Az IX regiszterpár /IX index regiszter/ tartalmának a d 2-es komplementű kódban adott egész számmal alkotott összege által kijelölt memóriarekesznek a b operandus által specifikált bitje 1-es állapotba kerül.

| <u>1-be állított bit</u> | <u>b</u> |
|--------------------------|----------|
| 0                        | 000      |
| 1                        | 001      |
| 2                        | 010      |
| 3                        | 011      |
| 4                        | 100      |
| 5                        | 101      |
| 6                        | 110      |
| 7                        | 111      |

M ciklusok: 6    T állapotok: 23/4,4,3,5,4,3/    4 MHz V.i.: 5.75

## Példa:

Ha az index regiszter tartalma 2000H, a

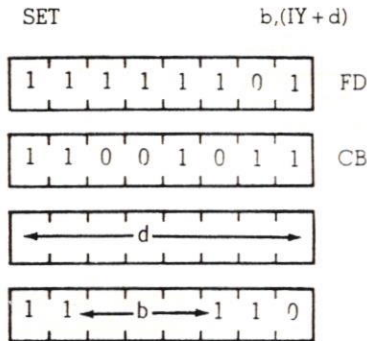
SET 0,(IX+3H)

utasítás végrehajtásának eredményeként a 2003H című memóriarekesz 0-ás bitje 1-es tartalmat vesz fel. /A 2003H című memóriarekeszben a 0-ás bit a legkisebb helyiértékű bit/.

# SET b.(IY+d)

Művelet:  $(IY+d)_b \leftarrow 1$

Formátum: OP kód            Operandusok



## Leírás:

Az IY regiszterpár /IY index regiszter/ tartalmának a d 2-es komplementű kódban adott egész számmal alkotott összege által kijelölt memóriarekesznek a b operandus által specifikált bitje 1-es állapotba kerül. A b operandus jelentése a következő:

| <u>1-be állított bit</u> | <u>b</u> |
|--------------------------|----------|
| 0                        | 000      |
| 1                        | 001      |
| 2                        | 010      |
| 3                        | 011      |
| 4                        | 100      |
| 5                        | 101      |
| 6                        | 110      |
| 7                        | 111      |

M ciklusok: 6    T állapotok: 23/4,4,3,5,4,3/    4 MHz V.i.: 5.75

## Példa:

Ha az IY index regiszter tartalma 2000H, a

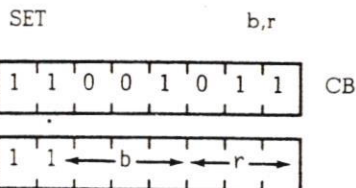
SET 0,(IY+3H)

utasítás végrehajtásának eredményeként a 2003H című memóriarekesz 0-ás bitje 1-es tartalommal lesz feltöltve. /A 2003H című memóriarekeszben a 0-ás bit a legkisebb helyiértékű bit/.

# SET b,r

Művelet:  $r_b \leftarrow 1$

Formátum:      OP kód      Operandusok



## Leírás:

Az r regiszter /a B, C, D, E, H, L vagy A regiszterek bármelyike/ b bitje /a 7-estől a 0-ásig bármelyik bit/ 1-es állapotba kerül. A b és r operandusok jelentése:

| <u>Bit</u> | <u>r</u> | <u>Regiszter</u> | <u>r</u> |
|------------|----------|------------------|----------|
| 0          | 000      | B                | 000      |
| 1          | 001      | C                | 001      |
| 2          | 010      | D                | 010      |
| 3          | 011      | E                | 011      |
| 4          | 100      | H                | 100      |
| 5          | 101      | L                | 101      |
| 6          | 110      | A                | 111      |
| 7          | 111      |                  |          |

M ciklusok: 2    T állapotok: 8/4,4/    4 MHz V.i.: 2.00

## Példa:

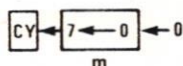
A

SET 4,A

utasítás végrehajtása után az A regiszter 4-es bitje 1-es tartalmat vesz fel. /A 0-ás bit a legkisebb helyiértékű bit/.

# SLA m

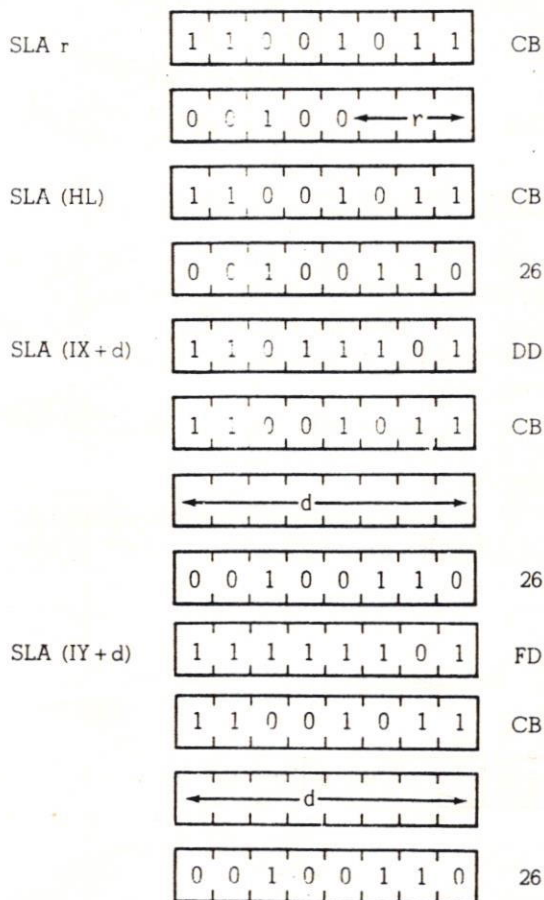
Művelet:



Formátum:            OP kód            Operandusok

SLA                            m

Az m operandus r, (HL), (IX+d) vagy (IY+d) bármelyike lehet, úgy, mint az RLC utasításoknál. A különböző lehetséges OP kód-operandus kombinációk az alábbiak:



Az r a B, C, D, E, H, L vagy A regisztereket azonosítja az alábbiak szerint:

| <u>Regiszter</u> | <u>r</u> |    |
|------------------|----------|----|
| B                | 000      | 20 |
| C                | 001      | 21 |
| D                | 010      | 22 |
| E                | 011      | 23 |

# SLA m

| <u>Regiszter</u> | <u>r</u> |
|------------------|----------|
| H                | 100 24   |
| L                | 101 25   |
| A                | 111 27   |

## Leírás:

Az utasítás az m operandus tartalmán egy helyiértékkel balra történő aritmetikai léptetést végez. A 0-ás bitbe 0 kerül, míg a 0-ás bit eredeti tartalma az 1-es bitbe kerül, az 1-es bit eredeti tartalma a 2-es bitbe töltődik, stb. Végül a 7-es bit az átvitel bitbe kerül. A 0-ás bit a legkisebb helyiértékű.

| UTASÍTÁS   | M CIKLUSOK | T ÁLLAPOTOK     | 4 MHz V.i. |
|------------|------------|-----------------|------------|
| SLA r      | 2          | 8/4,4/          | 2.00       |
| SLA (HL)   | 4          | 15/4,4,4,3/     | 3.75       |
| SLA (IX+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |
| SLA (IY+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: az m operandus által meghatározott byte 7-es bitjének tartalmát veszi fel

## Példa:

Ha az L regiszter tartalma

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
|   | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |

akkor az

SLA

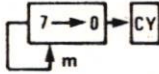
utasítás végrehajtásának eredményeként az L regiszter, és az átvitel bit tartalma a következő lesz:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| C | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 |



# SRA m

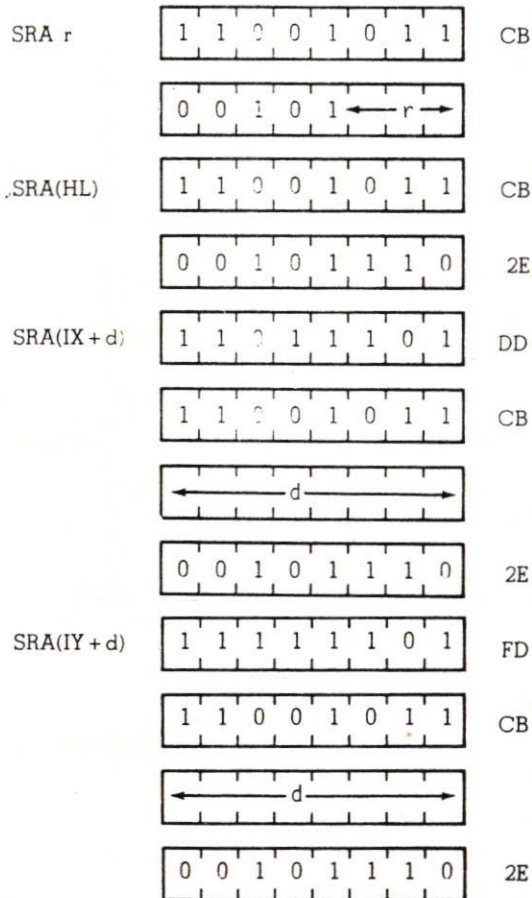
Művelet:



Formátum:      OP kód              Operandusok

                         SRA                              m

Az m operandus r, (HL), (IX+d) vagy (IY+d) bármelyike lehet, úgy, mint az RLC utasításoknál. A különböző OP kód-operandus kombinációk az alábbiak:



Az r a B, C, D, E, H, L vagy A regisztereket jelenti az alábbiak szerint:

| Regiszter | r   |    |
|-----------|-----|----|
| B         | 000 | 28 |
| C         | 001 | 29 |
| D         | 010 | 2A |
| E         | 011 | 2B |

# SRA m

| Regiszter | r            |
|-----------|--------------|
| H         | 100 $\tau c$ |
| L         | 101 $2D$     |
| A         | 111 $2F$     |

## Leírás:

Az utasítás az m operandus tartalmán egy helyiértékkel jobbra történő aritmetikai léptetést végez. A 7-es bit tartalma a 6-os bitbe töltődik, a 6-os bit eredeti tartalma az 5-ös bitbe kerül, stb. A 0-ás bit tartalma az átvitel bitbe kerül, és a 7-es bit eredeti tartalma változatlan marad. A 0-ás bit a legkisebb helyiértékű bit.

| UTASÍTÁS   | M CIKLUSOK | T ÁLLAPOTOK     | 4 MHz V.i. |
|------------|------------|-----------------|------------|
| SRA r      | 2          | 8/4,4/          | 2.00       |
| SRA (HL)   | 4          | 15/4,4,4,3/     | 3.75       |
| SRA (IX+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |
| SRA (IY+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: az m operandus által meghatározott byte 0-ás bitjének tartalmát veszi fel

## Példa:

Ha az IX index regiszter tartalma 1000H, és az 1003H című memóriarekesz tartalma

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |

akkor az

SRA (IX+3H)

utasítás végrehajtása után az 1003H című memóriarekesz, és az átvitel bit tartalma a következő lesz:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | C |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |



# SRL m

| Regiszter | r      |
|-----------|--------|
| E         | 011 3B |
| H         | 100 3C |
| L         | 101 3D |
| A         | 111 3F |

## Leírás:

Az utasítás az m operandus tartalmán egy helyiértékkel jobbra történő logikai léptetést végez. A 7-es bit tartalma a 6-os bitbe kerül, a 6-os bit tartalma az 5-ös bitbe kerül, stb. A 0-ás bit tartalma az átvitel bitbe kerül, és a 7-es bit törlődik. A 0-ás bit a legkisebb helyiértékű.

| UTASÍTÁS   | M CIKLUSOK | T ÁLLAPOTOK     | 4 MHz V.i. |
|------------|------------|-----------------|------------|
| SRL r      | 2          | 8/4,4/          | 2.00       |
| SRL (HL)   | 4          | 15/4,4,4,3/     | 3.75       |
| SRL (IX+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |
| SRL (IY+d) | 6          | 23/4,4,3,5,4,3/ | 5.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: törlődik
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: az m operandus által meghatározott byte 0-ás bitjének tartalmát veszi fel

## Példa:

Ha a B regiszter tartalma

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

akkor az

SRL B

utasítás végrehajtásának eredményeként a B regiszter, és az átvitel bit tartalma a következő lesz:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | C |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |



# SUB s

---

| UTASÍTÁS   | M CIKLUSOK | T ÁLLAPOTOK   | 4 MHz V.i. |
|------------|------------|---------------|------------|
| SUB r      | 1          | 4             | 1.00       |
| SUB n      | 2          | 7/4,3/        | 1.75       |
| SUB (HL)   | 2          | 7/4,3/        | 1.75       |
| SUB (IX+d) | 5          | 19/4,4,3,5,3/ | 4.75       |
| SUB (IY+d) | 5          | 19/4,4,3,5,3/ | 4.75       |

A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik
- P/V: 1-es állapotba kerül, ha túlcsordulás áll fenn; egyébként törlődik
- N: 1-es állapotba kerül
- C: 1-es állapotba kerül, ha alulcsordulás történt; egyébként törlődik

Példa:

Ha az akkumulátor 29H-t, és a D regiszter 11H-t tartalmaz, akkor a

SUB D

utasítás végrehajtásának eredményeként az akkumulátor 18H-t fog tartalmazni.



# XOR s

---

| UTASÍTÁS   | M CIKLUSOK | T ÁLLAPOTOK   | 4 MHz V.i. |
|------------|------------|---------------|------------|
| XOR r      | 1          | 4             | 1.00       |
| XOR n      | 2          | 7/4,3/        | 1.75       |
| XOR (HL)   | 2          | 7/4,3/        | 1.75       |
| XOR (IX+d) | 5          | 19/4,4,3,5,3/ | 4.75       |
| XOR (IY+d) | 5          | 19/4,4,3,5,3/ | 4.75       |

## A jelzőbit regiszter érintett bitjei:

- S: 1-es állapotba kerül, ha az eredmény negatív; egyébként törlődik
- Z: 1-es állapotba kerül, ha az eredmény nulla; egyébként törlődik
- H: 1-es állapotba kerül
- P/V: 1-es állapotba kerül páros paritás esetén; egyébként törlődik
- N: törlődik
- C: törlődik

## Példa:

Ha az akkumulátor tartalma 96H /10010110/, akkor az

XOR 5DH /Megj.: 5DH = 01011101/

utasítás végrehajtásának eredményeként az akkumulátor tartalma CBH /11001011/ lesz.