



IPARI INFORMATIKAI KÖZPONT

Z 80-as sorozat
VIII. rész:
CPU utasításkészlet

Tervezési segédlet

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

OMFB T árcaprogram

"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 Istvánná

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 "JuP-k alkalmazása" c. témafigyelésünk 1981/4. és 1982/1., 2-3. sz. Jrdőivein történő felmérés alapján - katalógus-hiánnyal küzdő kollegák részére készült.

A felhasználók további kérésére a sorozat egy részletes CPU utasítás-készlettel 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: SIÓ
- 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./ minden egyes utasítás esetén 4 MHz-es CPU órajel frekvenciát feltételezve μ 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 μ 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 A,s

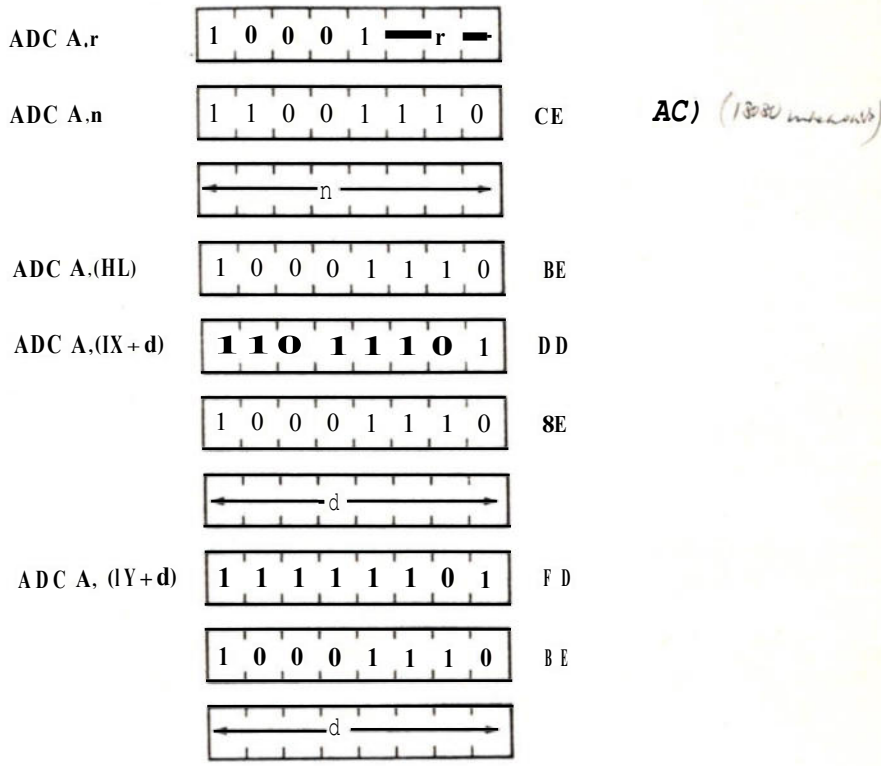
1.Sggg Abc o

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 / regisztereket a fenti tárgykódban a következőképpen azonosítja:

<u>Regiszter</u>	<u>r</u>	
B	000	&
C	001	M
D	010	ih
E	011	ib
H	100	it
L	101	8D
A	111	SF

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.

UT/SÍTÁS	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 jelzobit regiszter érintett bitiei:

- 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:

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)

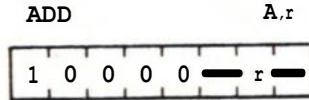
végrehajtása után az akkumulátor tartalma 27H lesz.

ADD A,r

! ?CVCJ /10D*^r

Művelet: A ← A + r

Formátum: OP kód Operandusok



Leírás:

Az r regiszter tartalma hozzáadódik az akkumulátor tartalmához, és az eredmény az akkumulátorban tárolódik. Az r szimbólumtárgykódban 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	7
E	011	12
H	100	V
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 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 44H és a C regiszteré 11H, az

ADD A,C

végrehajtásának eredményeként az akkumulátor tartalma 55H lesz.

ADD A.(IY+d)

művelet: $A \leftarrow A + (iY+d)$

Formátum: OP kód Operandusok

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,(1Y+5H)

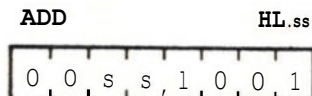
végrehajtásának eredményeként az akkumulátor tartalma 33H lesz.

ADD HL,ss

ISIAÉ P/ft is

Művelet: HL ← HL + ss

Formátum: OP kód Operandusok



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	9*
DE	01	19
HL	10	??
SP	11	&

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

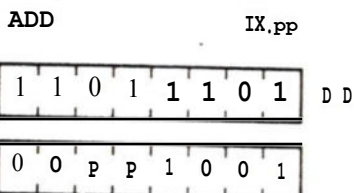
ADD HL,DE

végrehajtása után a HL regiszterpár tartalma 5353H lesz.

ADD DC,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 "1
IX	10 U
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

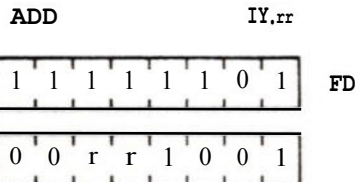
ADD IX,BC

végrehajtásának eredményeként az IX tartalma 8888H lesz.

ADD IY.rr

Művelet: $IY \leftarrow IY + rr$

Formátum: OP kód Operandumok



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	01
DE	01	19
IY	10	29
SP	11	V

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

ADD IY,BC

végrehajtása után az IY tartalma 8888H lesz.

AND s

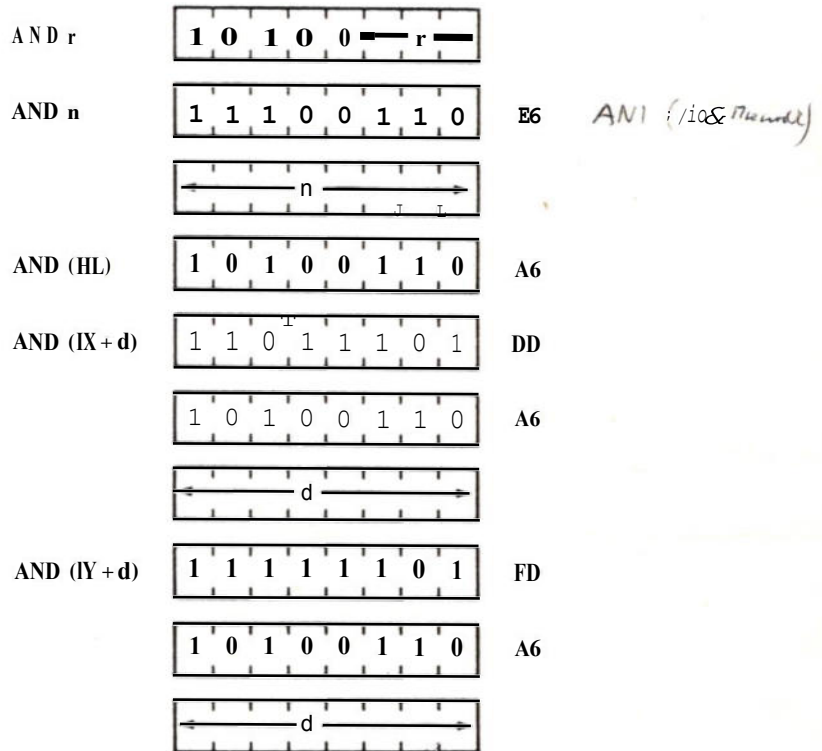
ISVW APJA 0

Művelet: $A \leftarrow A \text{ AND } s$

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 az alábbiak szerint:

Regiszter	£	M
B	000	M
O	001	fi*
D	010	/17
E	011	A3
H	100	^
L	101	Ajr
A	111	Af

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	N 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

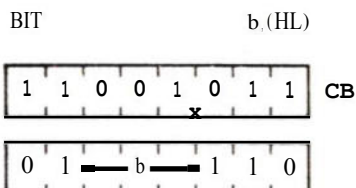
AND B

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

BIT b.(HL)

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

Formátum: OP kód Operandusok



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	ki
1	001	sf
2	010	37
3	011	&
4	100	U
5	101	a
6	110	^
7	111	7f

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

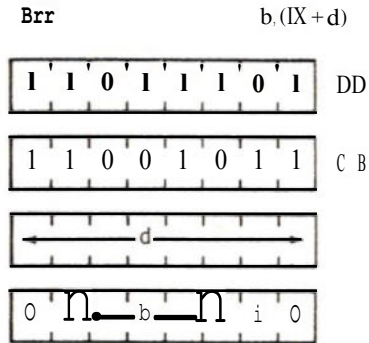
BIT 4,(HL)

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: $Z \leftarrow \overline{P \wedge}_b$

Formátum: OP kód Operandusok



Leírás:

Ennek ez 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:

Vonatköző bit	b	
0	000	*C
1	001	*E
2	010	SE
3	011	SE
4	100	bC
5	101	^
6	110	74
7	in	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

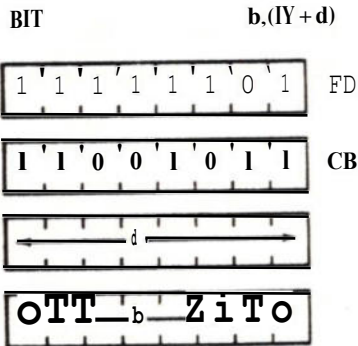
BIT 6 (IX+4H)

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

formátum: OP kód Operandusok



Leírás:

Ennek az utasításnak a végrehajtása után a Z bit tartalmazni fogja az IY regiszterpár /IY index regiszter/ és 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 <i>96</i>
1	001 <i>4E</i>
2	010 <i>M</i>
3	011 <i>if</i>
4	100 <i>it</i>
5	101 <i>cf</i>
6	110 <i>lí</i>
7	111 <i>^</i>

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

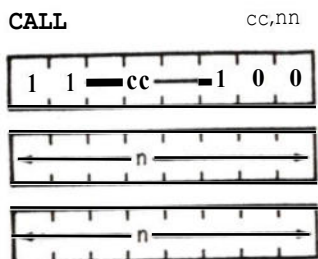
EIT 6, (1Y+4H)

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

CALL cc,nn

Művelet: Ha cc igaz: $(SP-1) \leftarrow PCH$
 $(SP-2) \leftarrow PO_{\underline{1}}, PC \leftarrow nn$

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 veremtar 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 veremtar 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 veremtar 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 veremtar 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 <i>nn</i>
Cc	001	Z zérus	Z	eZ <i>ia nn</i>
p4	010	NC nincs átvitel	C	C UC <i>nn</i>
DC	011	C átvitel	C	CC <i>nn</i>
E4	100	PO páratlan paritás	P/V	c PC <i>nn</i>
ZC	101	PE páros paritás	P/V	c PC <i>nn</i>
F4	110	P pozitív előjel	S	cf <i>κ ←</i>
fi	H1	M negatív előjel	S	crt <i>nn</i>

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:

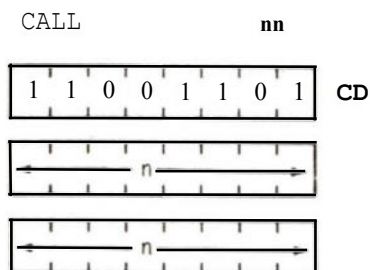
CALL NC,2135H

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: $(SP-1) \leftarrow PCH,$ $(SP-2) \leftarrow PCL,$ $PC \leftarrow nn$

Formátum: OP kód Operandusok



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 orra 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 RÉT 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 Vihelyezése történik meg. Ezután az EP 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

iStfio *CNC*

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

regiszter megfelelő bitjének beállítása jelzi.

UTASÍTÁS	M CIKLUSOK	T ÁLLAPOTOK	4 MHz V.i.
CP r	•	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



Leírás:

A HL regiszterpár által megcímzett 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.

¥ 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-1/0; 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 ← HL-1, BC ← BC-1

Formátum: OP kód

CPDR

1	1	1	0	1	1	0	1
---	---	---	---	---	---	---	---

 ED

1	0	1	1	1	0	0	1
---	---	---	---	---	---	---	---

 B9

Leírás:

A HL regiszterpár által megcímezett memóriarekesz tartalma az akkumulátor regiszter tartalmával kerül összehasonlításra. Az összehasonlítás eredményeként egy feltétel hit fog 1-es állapotba kerülni. A HL és a BC /Eyte 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 $A/(HI)$, a programszámláló 2-vel dekrementálódik, és ez 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=0 és A=(HL) esetén:

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

BC=Q 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-1/0; 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

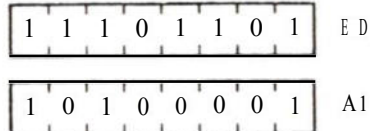
utasítás végrehajtása után a HL regiszterpár tartalma 1115H, a byte számláló tartalma 0004H lesz, a P/V bit 1-es állapotba kerül, és a Z bit is 1-es állapotba kerül.

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 Y.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 $BC-1/0$; egyébként törlődik
- N: 1-es állsnotba kerül
- C: változatlan marad

Példa:

Ha a HL regiszternár 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

1	1	1	0	1	1	0	1
---	---	---	---	---	---	---	---

 ED

1	0	1	1	0	0	0	1
---	---	---	---	---	---	---	---

 31

Leírás:

A HL regiszterpár által megcímezett memórierekész tartalma az akkumulátor regiszter tartalmával kerül összehasonlításra. Az összehasonlítás eredményeként egy feltétel hit 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/(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=0 és A/(HL) esetén:

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

BC=Q 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-1/0; 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:

```
C1111H) : 52H
C1112H) : 00H
C1113H) : F3H
```

ekkor 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

ISOIO SHA

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 összeadásakor 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 tartalom DAA előtt	Magasabb helyiértékű SZÁM /7-4 bit/ HEXA értéke	H tartalom DAA előtt	Alacsonyabb helyiértékű SZÁM /3-0 bit/ HEXA értéke	BYTE- hoz- adott szám	C tartalom DAA 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, akkor

$$\begin{array}{r} 0001 \quad 0101 \\ + 0010 \quad 0111 \\ \hline 0011 \quad 1100 \quad 30 \end{array}$$

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

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

DEC m

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

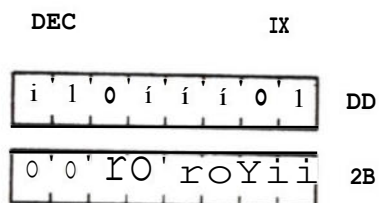
DEC D

végrehajtásának eredményeként a D regiszter tartalma 29H lesz.

DEC IX

Művelet: IX ← IX-1

Formátum: OP kód Operánd úsok



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

EEC IX

végrehajtásának eredményeként az IX index regiszter tartalma 2005K 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

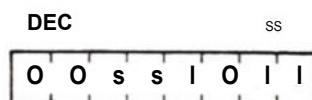
végrehajtásának eredményeként az IY index regiszter tartalma 7648H lesz.

DEC ss

IW PCX ^{ss}

Művelet: ss ← ss-1

Formátum: OP kód Operandumok



Leírás:

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

<u>Regiszterpár</u>	<u>ss</u>	
BC	00	^
DE	01	1b
HL	10	2»
SP	11	36

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

Példa:

Ha a HL regiszterpár tartalma 1001H, a

DEO HL

végrehajtásának eredményeként a HL tartalma 1000H lesz.

D1

Művelet: IPP ← 0

Formátum: OP kód

D1

1	1	1	1	0	0	1	1
---	---	---	---	---	---	---	---

 F3

Leírás:

A megszakítás enged élyező flip-flopok /IFF1 és IFF2/ törlésével a D1 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

D1

utasítást hajtja végre, a maszkolható megszakítás mindaddig letiltásra kerül, amíg azt egy később végrehajtott E1 utasítás újra nem enged élyezi. 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,INBUE      ; a pointerek előkészítése
LD      DE,OUTBUF     ;

LOOP:   LD      A,(HL)  ; a következő byte megszerzése
        ; az input pufferből
LD      (DE),A        ; tárolás az output pufferben
GP      ODH           ; CR volt ?
JR      Z,KÉSZ        ; Igen, vége
ING     HL            ; nem, pointer inkrementálás
IRC     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 ilip-flopokat /IPP1 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 kérélem /Interrupt Request - INT/ elfogadására.

EX (SP),HL

j³⁰ XTHL

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

Formátum: OP kód Operancusok

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 8356H-t tartalmaz, a 8856H című memóriarekeszben lévő byte 11H és a 8857H című memóriarekeszben lévő byte 22H, akkor az

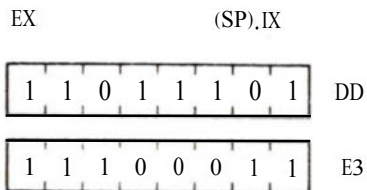
EX (SP), HL

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 3356H-t fog tartalmazni.

EX (SP),IX

Művelet: $IX_H \leftrightarrow (SP+1), \quad [IX] \leftrightarrow (SP)$

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 állaoatok: 23/4,4,3,4,3,5/ 4 MHz V.i.: 5.75

Példa:

Ha az IX index regiszter 3938H-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: $IY_{H} \leftarrow (SP+1)$, $IY_{L} \sim (SP)$

Formátum: OP kód Operandusok

EX (SP),IY

1	1	1	1	1	1	0	1
x							

 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 Operandumok

EX AF,AF'

0	0	0	0	1	0	0	0
---	---	---	---	---	---	---	---

 08

Lairás:

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

186£0 XC HG,

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 regiszterpárok tartalma felcserélődik.

in 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: $(BC) \leftrightarrow (BC')$, $(DE) \leftrightarrow (DE')$, $(HL) \leftrightarrow (HL')$

Formátum: OP kód

EXX

1	1	0	1	1	0	0	1
---	---	---	---	---	---	---	---

 D9

Leírás:

A BC, DE és HL rémiszterpárookban lévő minőén egyes 2 byte-os érték rendre felcserélődik a BC', DE' és HL' regiszterpárookban lévő 2 byte-os értékkel.

M ciklusok: 1 T állanotok: 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: 9100F; 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 s memória frissítő logika működtetéséhez LOP utasításokat hajt végre.

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

IM 0

Művelet: Interrupt Mode 0 beállítása

Főnntám: OP kód Operanősok

IM	0	
1 1 1 0 1 1 0 1		ED
0 1 0 0 0 1 1 0		46

Leírás:

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

M ciklusok: 2 T állanotok: 3/4,4/ 4 MHz V.i.: 2.00

IM 1

Művelet: Interrupt Mode 1 beállítása

Formátum: OP kód Ooerandusok

IM	1	
1 1 1 0 1 1 0 1	ED	
0 1 0 1 0 1 1 0	5 6	

Leírás:

Az IM utasítás az 1-es megszakítási üzemmódot /mode 1/ állítja be. Ebben az üzemmódban a processzor a megszakítási kérélemre egy 003BH-&s rekeszre irányuló „Rest&rt” végrehajtásával válaszol.

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

IM 2

Művelet: Interrupt Moce 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:

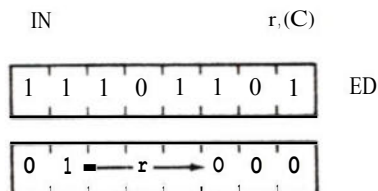
Az IM 2 utasítás a 2-es megszakítási üzemmódot /mode 2/ állítja be. Ez az üzemmód egy tetszőleges memóriarekeszre irányuló „Indirekt Call” végrehajtását teszi lehetővé. Ebben az üzemmódban a CPU egy 16 bites memóriacímet állít össze. A cím felső 8 bitjét az I „Interrupt Vector Register” tartalmazza,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

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

IN r,(C)

Művelet: r ← (0)

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 E 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:

Regiszter	Z	Un
B	000	4BH
O	001	5BH
D	010	5BH
E	011	5BH
H	100	C _p H
L	101	6BH
A	111	7BH

30H OP A. Lévegye van. Kétféle van!

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 m n

Művelet: (HL) ← (HL) + I

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 \leftarrow IX+1

Formátum: OP kód Operandusok

INC

IX

1	1	0	1	1	1	0	1
---	---	---	---	---	---	---	---

 DD

C	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

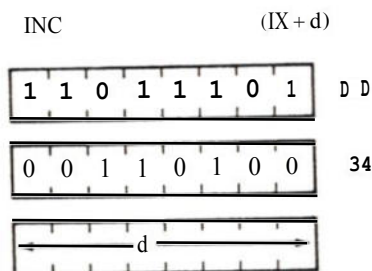
INC IX

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

INC (IX+d)

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

Formátum: OP kód Operancusok



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/7: 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 34^u , 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	F D
---	---	---	---	---	---	---	---	-----

0	1	1	0	0	0	1	1	2 3
---	---	---	---	---	---	---	---	-----

Leírás:

Az IY index regiszter tartalma inkrementálódik.

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

Példa:

Ha az index regiszter tartalma 2977H, akkor az

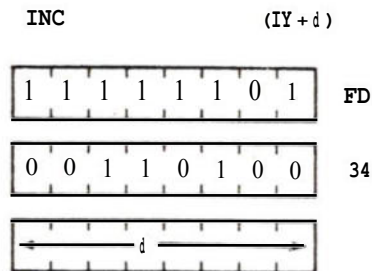
INC IY

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 jelzobit 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)

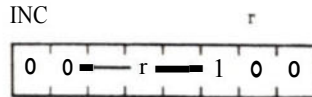
utasítás végrehajtásának eredményeként a 2030H című memóriarekesz tartalma 35H lesz.

INC r

í 130 iHZ 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:

<u>Regiszter</u>	<u>r</u>	
A	111	5A
B	000	6A
C	001	6K
D	010	7A
E	011	SIC
H	100	7A
L	101	ZC

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

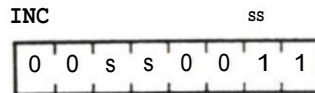
utasítás végrehajtásának eredményeként a D regiszter tartalma 29H lesz.

INC ss

IA80 iFOX

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>Reg. Lszterpár</u>	<u>ss</u>	
BC	00	01
DE	01	13
HL	10	i5
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

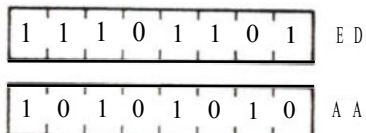
utasítás végrehajtásának eredményeként a HL 1001H-t fog tartalmazni.

IND

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

Formátum: OP kód

IND



Leírás:

A periféria regisztert közvetve, regiszter indirekt címzésig móddal címző I/O utasítás, amelynél a periféria regiszter címét a G 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

Pálda:

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 OFPFH-t, és a B regiszter OFH-t fog tartalmazni.

INDR

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

Formátum: OP kód

INDR

1	1	1	0	1	1	0	1
---	---	---	---	---	---	---	---

 ED

1	0	1	1	1	0	1	0
---	---	---	---	---	---	---	---

 3A

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 S 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 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

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. Ugyanakkor 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

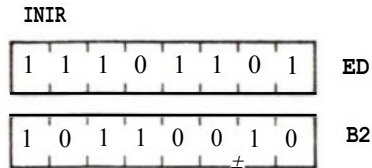
INI

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

INIR

Művelet: $(HL) \leftarrow (C)$, $B \leftarrow B-1$, $HL \leftarrow HL+1$

Formátum: OP kód



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. 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 orogramszámláló tartalma 2-vel dekrementálódik, és az utasítás végrehajtása ismétlődik. Megjegyeztük, 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

ekkor az

INIR

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

<u>Rekesz</u>	<u>Tartaloin</u>
1000H	51H
1001H	A9H
1002H	03H

JP (HL)

IIW PCffL

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.

IP (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

IP	(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

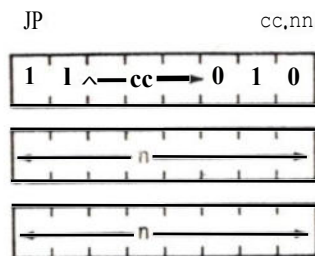
JP (IY)

utasítás végrehajtásának eredményeként a programszámláló tartalma 4800H lesz.

IP cc,nn

Művelet: Ha cc igaz, PC ← nn

Formátum: OP kód Operandusok



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 kezdő 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	
ez	000	NZ nem zérus	Z	JUZ ^
a	001	Z zérus	Z	JZ <i>na</i>
DZ	010	FTC nincs átvitel	C	JAJc <i>na</i>
DA	011	C átvitel	C	X ^
F2	100	PO páratlan paritás	P/V	JPO <i>na</i>
sfi	101	PE páros paritás	P/V	JPE <i>na</i>
FZ	110	P pozitív előjel	S	JN' ^
rA	H1	M negatív előjel	S	Jg IKA.

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

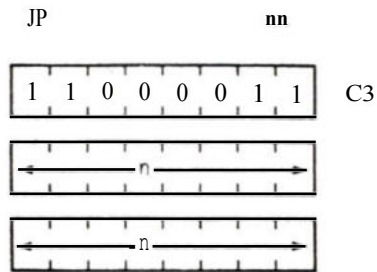
végrehajtásának eredményeként a programszámláló 1520H-t fog tartalmazni, és a következő gépi ciklusban a CPU az 1520H-s címen elhelyezett 03H-s byte-ot fogja kiemelni.

JP nn

'&iü JW wv

Művelet: PC ← nn

Formátum: OP kód Operandusok



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

Leírás:

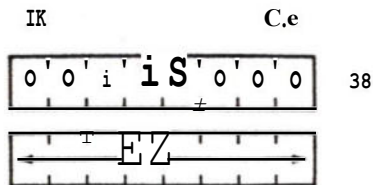
Feltétel nélküli vezérlés átadó utasítás. Végrehajtása során az nn operandus a programszámláló regiszterbe /PC/ töltődik, és így kijelöli azt a memóriarekeszt, ahonnan a CPU a következő utasítást kiemeli.

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 vezérlésáttást 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, CÍMKE

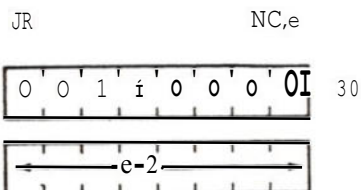
Az eredményként kapott tárgykód, és a PC végső értéke az alábbi táblázaton látható:

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

JR NC.e

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

Formátum: OP kód Operandusok



Leírás:

Az utasítás az átvitel bit vizsgálatának eredményétől függően egy más programszegmensre történő relatív címzéssel feltételes vezérlésáttárást 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 ismételní. Az assembler nyelvű utasítás:

CÍMKE: JR NC,CÍMKE

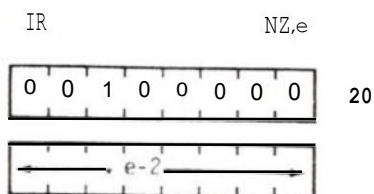
Az eredményként kapott tárgykód, és a PC végső értéke az alábbi táblázaton látható:

Rekesz		OP kód
480	CÍMKE: JR NC,CÍMKE	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 vezérlés:adást 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 címből fogja a CPU kiemelni. Az ugrás hossza az utasítás 0? 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 rekesznyi kell visszaugrani. Az assembler nyelvű utasítás:

JR NZ,CÍMKE

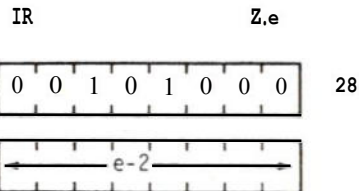
Az eredményként kapott tárgykód, és a PC végső értéke az alábbi táblázaton látható:

<u>Rekesz</u>		<u>OP kód</u>
47C	CÍMKE: -	PC az ugrás után
47D	-	
47E	-	
47F	-	
480	JR NZ,CÍMKE	20
481		FA /-6, 2-es komplement kódban/

JR Z.e

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

Formátum: OP kód Operandusok



Leírás:

Az utasítás a Z zérus bit vizsgálatának eredményétől függően egy más jDrogramszegmensre 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 0? 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 memóriacím-től 5 rekesznyi-t kell előre ugrani. Az assembler nyelvű utasítás a következő:

JR Z,CÍMKE

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

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

LD A.(BC)

I\$020 LDA X &L

Művelet: A * (BC)

Jormátum: O? kód Operandusok

LD A.(BC)

0	0	0	0	1	0	1	0
---	---	---	---	---	---	---	---

 OA

Leírás:

A 3C regiszterpár tartalma által specifikált memóriarekesz terstslma 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)

WogO LÖAX bE

Művelet: A ← (DE)

Formátum: OP kód Operandumok
 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: 1 T állapotok: 7/4,3/ 4 MHz V.i.: 1.75

Példa:

Ha a DE regiszterpár tartalma számértékben 30A2H, és a 30A2K című memóriarekeszben lévő byte 22H, akkor az

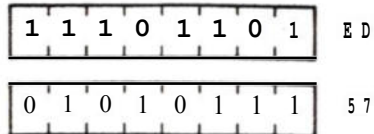
LD A,(DE)

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

LD A,I

Művelet: $A \leftarrow I$

Formátum: OP kód Operandumok
 LD A,I



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 érinteti: 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.

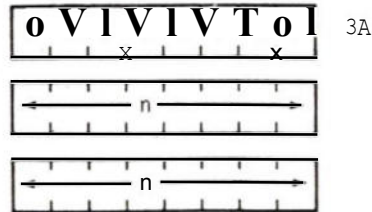
LD A,(nn)

13030 LOA Kz

Művelet: A ← (nn)

Formátum: OP kód Operandusok

10 A.(nn)



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 04K byte az akkumulátorba töltődik.

LD A,R

Művelet: A \ll -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: 9/4,0/ 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
- 17: 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 SVA y Se

Művelet: (BC) ← A

Formátum: OP kód Operandumok

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

12020 \leq TA X DE

Művelex: (DE) ← A

Formátum: OP kód Operandueok
 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, ér az akkumulátorban lévő byte AOH, akkor az

LD (DE),A

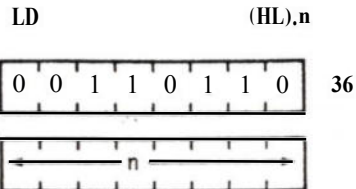
utasítás végrehajtásának eredményeként AOH a 1128H című memóriarekeszbe töltődik.

LD (HL),n

10010 MV\ M, n

Művelet: (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

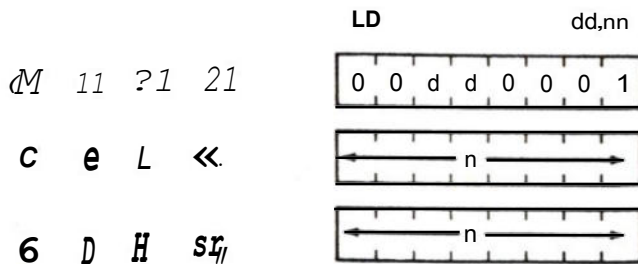
utasítás végrehajtásának eredményeként a 4444H című memóriarekeszbe 28H töltődik.

LD dd,nn

ílcS₂ LX I árt/**

Művelet: dd * << nn

Formátum: OP kód Operandusok



Leírás:

Az rm 2 byte-os egész szám a dd regiszterpárba töltődik, ahol a dd a 3C, DE, HL vagy SP regiszterpárok egyikét jelöli, ezek tárgykódban a következők lehetnek:

<u>Regiszterpár</u>	<u>dd</u>
3C	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

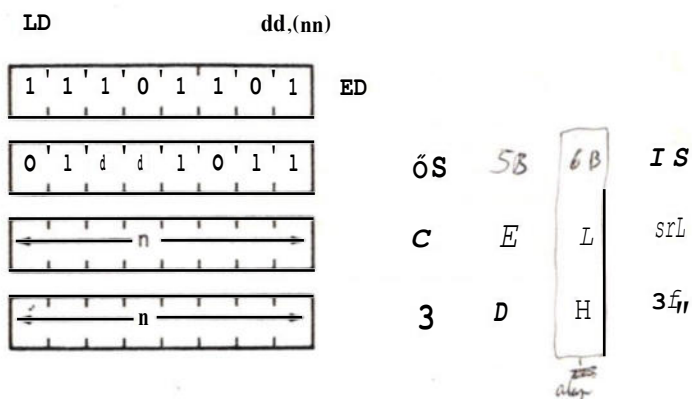
LD HL,5000H

utasítás végrehajtása után a HL regiszterpár tartalma 5000H lesz.

LD dd,(nn)

Művelet: $dd \leftarrow (nn+1), \quad adj \leftarrow (nn)$

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>
1C	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.

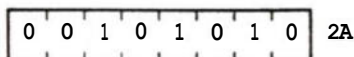
LD HL.(nn)

iSiSo LMO AK

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

Formátum OP kód Operandumok

LD HL.(nn)



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 11H-t tartalmaz, akkor az

LD HL,(4545H)

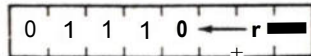
utasítás végrehajtásának eredményeként a HL regiszterpár A137H-t fog tartalmazni.

LD (HL),r

Művelet: (HL) ← r

Formátum: OP kód Operandumok

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>£</u>	
A	111	77
B	000	70
C	001	71
D	010	T
E	011	91
H	100	n
L	101	Fi

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 LA

Művelet: $I \leftarrow A$

Formátum: OP kód Operánd úsok

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

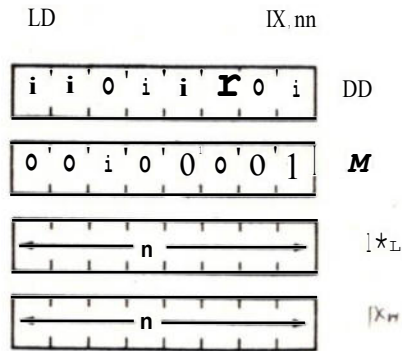
LD I,A

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

LD IX.nn

Művelet: $IX \leftarrow nn$

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

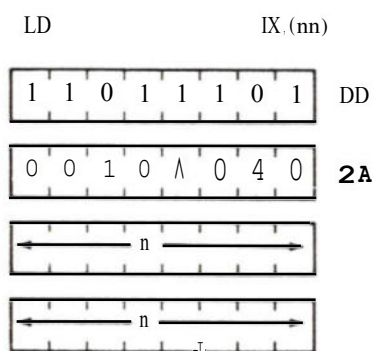
LD IX,45A2H

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

LD IX,(nn)

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

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

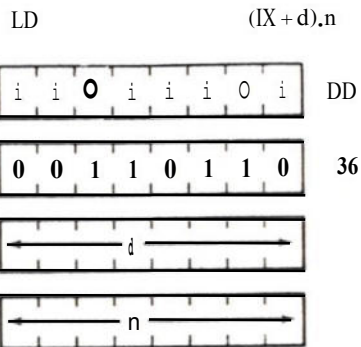
LD IX,(6666H)

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

LD (IX+d).n

Művelet: $(IX+d) \leftarrow n$

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.

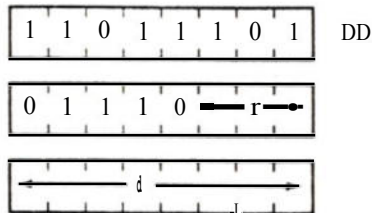
LD (IX+d),r

Művelet: $(iX+d) \leftarrow r$

Formátum: OP kód Operandusok

LD

(IX+d),r



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 regisztert azonosítja, amelyek tárgykódban a következők:

<u>Regiszter</u>	<u>f</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

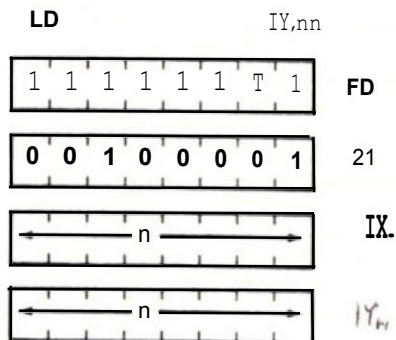
LD (1X+6H),C

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

LD IY,nn

t: IY ← nn

ura: OP kód Operandusok



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

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

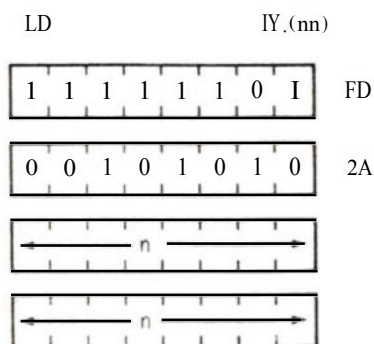
LD IY,7733H

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

LD IY,(nn)

Művelet: $IY_{\wedge} \leftarrow (nn+1)$, $IY_{\lrcorner} \leftarrow (nn)$

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)

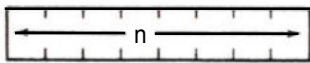
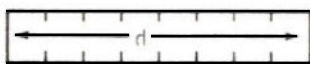
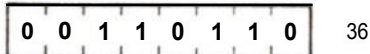
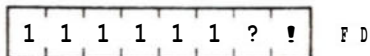
utasítás végrehajtásának eredményeként az IY index regiszter tartalma DA92H lesz.

LD (IY+d),n

Ut: (IY+d) ← n

Form: OP kód Operandusok

LD (IY+d),n



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

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

IY index regiszter tartalma számértékben A940H, az

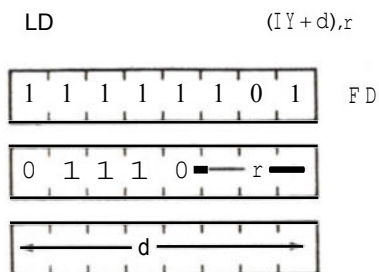
LD (1Y+10H),97H

ás végrehajtásának eredményeként a 97H byte az A950H amóriarekeszbe 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 komplex egész számként értelmezett címkiegészítés összege határozza meg. Az r szimbólum az alábbi táblázat alapján specifikus:

<u>Regiszter</u>	<u>Z</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.: t

Példa:

Ha a C regiszterben lévő byte 48H, és az IY index regiszter 2A11K-t tartalmaz, akkor az

LD (IY+4H),C

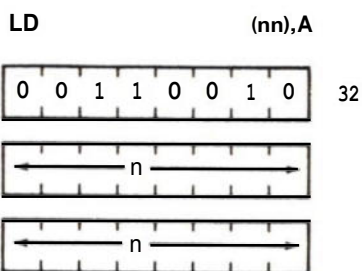
utasítás elvégzi a 2A11H+4H összegzést, és 48H-t a 2A15H című memóriarekeszbe tölti.

LD (nn),A

14080 STA nn

t: (nn) ← A

um: OP kód Operandumok



omulátor tartalma az nn operandusok által specifikált arekeszbe töltődik. A fenti tárgykódban elsőként szeri opersndus nn alacsonyabb helyiértékű byte-ja.

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

akkumulátor tartalma D7H, akkor az

H

LD (3141H),A

is végrehajtása után D7H a 3141H című memóriarekeszbe Lk.

LD (nn),HL

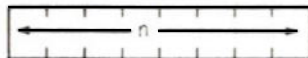
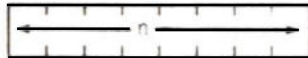
1880 ZHLD ^

élet: $(nn+1) \leftarrow H, (nn) \leftarrow A$

mátum: OP kód Operandusok

LD (nn),HL

0 0 1 0 0 0 ! ? 22



rás:

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

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

la:

i HL regiszterpár tartalma 483AH, akkor az

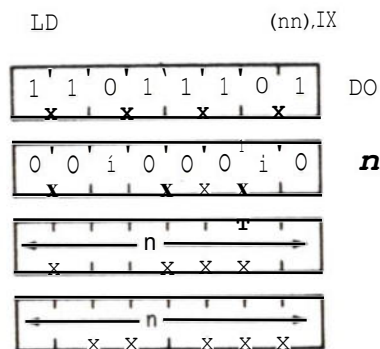
LD (B229H),HL

sítás végrehajtásának eredményeként a B229H című memóriasz tartalma 3AH, a B22AH című memóriarekeszé pedig 48H z.

LD (nn).IX

Művelet: $(nn+1) \leftarrow IX_{\wedge}, (nn) \leftarrow IX_{\wedge}$

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

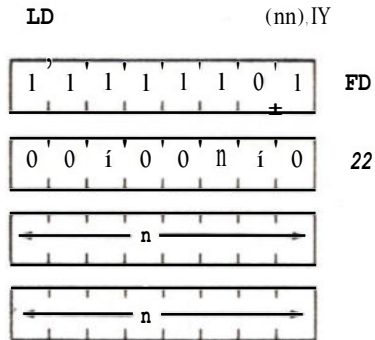
LD (4392H),IX

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

LD (nn),IY

Operáció: $\wedge nn+1 \wedge \leftarrow IYH, (nn) \leftarrow IYL$

Ná t um: OP kód Operandusok



rás:

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 ábrában elsőként szereplő n operandus nn alacsonyabb helyiértékű byte-ja.

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

la:

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

LD (8838H),IY

Operáció 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

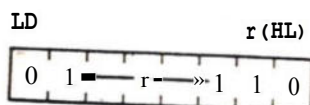
LD R,A

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

LD r,(HL)

efekt: $r \leftarrow (HL)$

átviteli mód: OP kód Operandusok



leírás:

L tartalma által kijelölt memóriarekeszben lévő byte az HL regiszterbe töltődik, ahol r tárgykódban az A, B, C, D, E, H vagy L regisztereket azonosítja:

<u>Regiszter</u>	<u>Z</u>
A	011
B	000
C	001
D	010
E	011
H	100
L	101

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

Példa:

Ha HL regiszterpár tartalma számértékben 73A1H, és a 75A1H című memóriarekeszben lévő byte 58H, akkor az

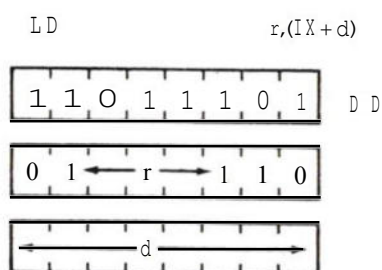
LD C,(HL)

utasítás végrehajtásának eredményeként 58H a C regiszterbe töltődik.

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 memó kész 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)

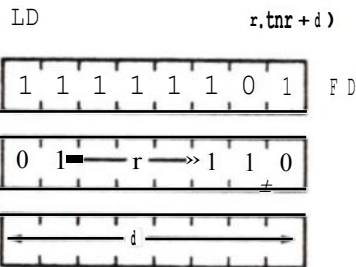
utasítás a 25AFH+19H összeadást eredményezi, ami a 25C mű memóriarekeszre mutat. Ha ez a memóriarekesz 39H-t tartalmaz, az utasítás végrehajtásának eredményeként a B regiszter szintén 39H-t fog tartalmazni.

U
 r
 5M
 *H
 10
 f
 >K
 Hh
 O
 1
 a
 D
 ff
 a
 H

LD r,(IY+d)

: r ← {1Y+d'}

m: OP kód Operandusok



d operandus /az IY index regiszter tartalma, és a d számmal adott címkiegészítés által kijelölt memóriare-
 irtalmának összege/ az r regiszterbe töltődik, ahol r
 i, C, D, E, H vagy L regisztereket azonosítja, amelyek
 kódban a következők:

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

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

CY index regiszter tartalma számértékben 25AFH, az

LD B, (IY+19H)

ás a 25AFH+19H összeadást eredményezi, amely a 25C8H
 emóriarekeszre mutat. Ha ez a memóriarekesz 39H-t tar-
 , az utasítás végrehajtásának eredményeként a B regisz-
 tén 39H-t fog tartalmazni.

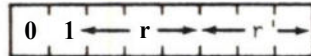
LD r,r'

ISC hcv y,y'

Művelet: $r \leftarrow r'$

Formátum: OP kód Operandusok

LD ..'



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

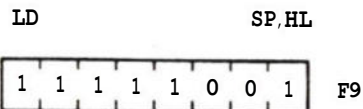
utasítás végrehajtása után mindkét regiszter 10H-t fog tartalmazni.

LD SP,HL

rti 3t) SFHL

Művelet: SP ← HL

Formátum: OP kód Operandusok



Leírás:

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

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

Példa:

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

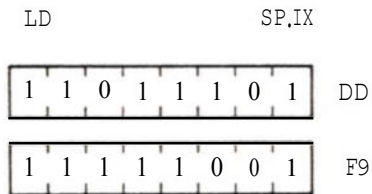
LD SP,HL

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

LD SP,IX

Művelet: SP ← IX

Formátum: OP kód Operandusok



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 93DAH, akkor az

LD SP,IX

utasítás végrehajtása után az SP regiszter tartalma szintén 93DAH lesz.

LD SP,IY

Művelet; SP ← IY

Formátum; OP kód Operandusok

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;

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

LD SP,IY

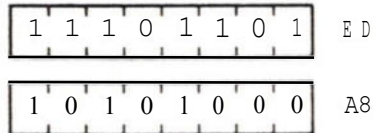
utasítás végrehajtásának eredményeként az SP regiszter is A227H-t fog tartalmazni.

LDD

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

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 83H, 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:

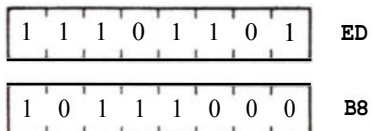
HL	:	1110H
(1111H)	:	88H
DE	:	2221H
(2222H)	:	88H
BC	:	6H

LDDR

Művelet: (DE) ← (HL), DE ← DS-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 regiszteroár, 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 rnegszakítási kérelmek is érvényre juthatnak.

He 3C/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
I/V: törlődik
N: törlődik
Q: 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
C1112H3	:	8SH	C2223H)	:	66H

akkor az

LDDR

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

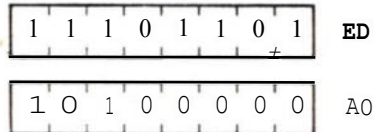
HL	:	1111H			
DS	:	2222H			
BC	:	0000H			
C1114H)	:	A5H	C2225H)	:	A5H
C1113H3	:	36H	(2224H)	:	36H
(1112H)	:	8BH	C2223H)	:	88H

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 83H, 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:

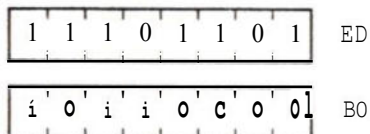
HL	:	1112H
(1111H)	:	88H
DE	:	2223H
(2222H)	:	88H
BC	:	6H

LDIR

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

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 megszakítás! kérelmek 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	C2223H) :	59H
U113H) :	A5H	C2224H) :	C5H

akkor az

LDIR

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

HL :	1114H		
DE :	2225H		
BC :	0000H		
(1111H) :	88H	C2222H) :	88H
C1112H) :	36H	(2223H) :	36H
(1113H) :	A5H	(2224H) :	A5H

NEG

Művelet: A O-A
Formátún: 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 30H 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

i	0	0	i	1	C	0	0
---	---	---	---	---	---	---	---

a HEG

utasítás végrehajtásának eredményeként az akkumulátor tartalma

0	i	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.

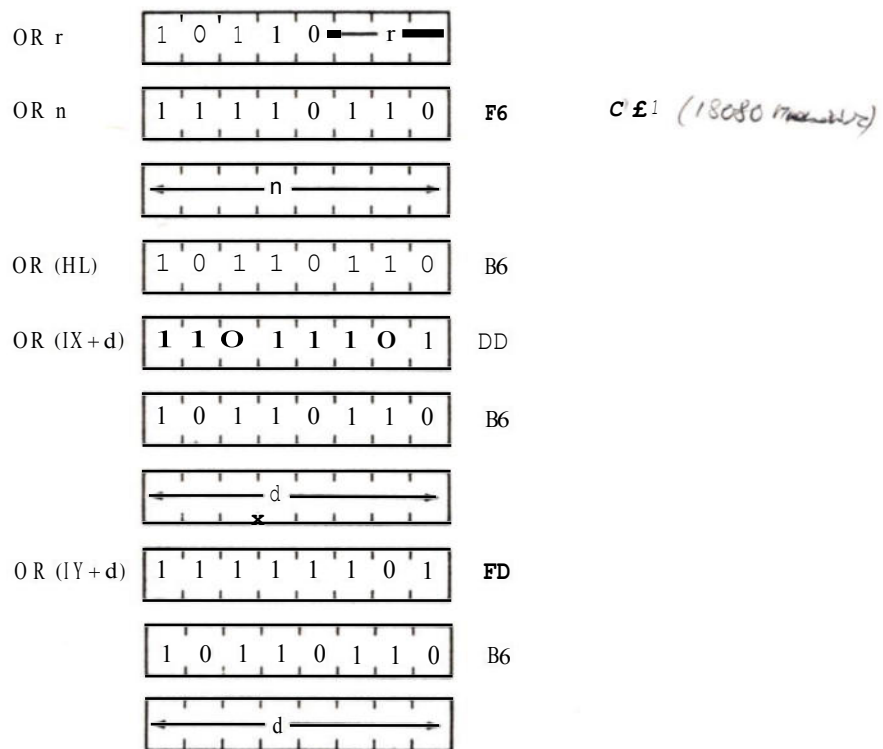
M ciklusok: 1 T állapotok: 4 4 MHz V.i.: 1.00

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>L</u>	
S	000	£0
C	001	D'
D	010	*L
E	011	^
H	100	W
L	101	5§
A	111	*3

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 43H 01001000 , és az akkumulátoré 12H (00010010) , akkor az

OR H

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

OTDR

Művelet: $\wedge C) \leftarrow (HL)$, $B \leftarrow B-1$, $HL \leftarrow HL-1$

fformátum: OP kód

OTDR

1	1	1	0	1	1	0	1	ED
1	0	1	1	1	0	1	1	BB

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/O:

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

Példa:

Ka a C regiszter tartalma 07H, a B regiszteré 03H, a HL regiszterpár 1000K-t tartalmaz, és a memóriaregiszterek tartalma a következő:

<u>Rekesz</u>	<u>Tartalom</u>
OPPEK	51H
OPPPH	A9H
1000K	03H

akkor az

OTDR

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

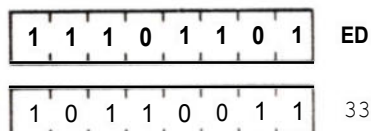
03H
A9H
51H

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, amelynek 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 tartalma nem zérus, akkor a programszámláló tartalma 2-vel dekrementálódik, és az utasítás megismétlődik. Ha viszont L 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/O:

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 jolzőbit regiszter érintett bitjei:

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

Példa:

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

<u>Rskesz</u>	<u>Tartalom</u>
1000H	51H
1001H	A9H
1002H	03H

akkor az

OTIR

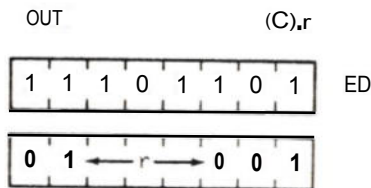
utasítás végrehajtása után a HL regiszterpár tartalma 1003H lesz, a B regiszter 0-t fog tartalmazni, és az eredetileg a memóriában lévő byte-sorozat a következő sorrendben a 07H című periféria regiszterbe töltődik:

51H
A9H
03H

OUT (C),r

Művelet: (C) ← r

formátum: OP kód Opersndusok



Leírás:

A periféria regisztert közvetve, regiszter indirekt címzési móddal címző 1/0 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	L	
B	000	<1ii
C	001	49ii
D	010	f1H
E	011	yjn
H	100	(in
L	101	iUH
A	111	79ii

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

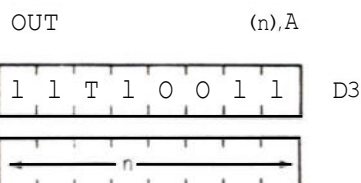
OUT (C),D

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

OUT (n),A

Üvelet: (n),← A

Formátum: OP kód Operandusok



Leírás:

A végrehajtás során az utasításnak a címbusz alacsony helyiértékű bitjeir[®] /A0-tól A7-ig/ helyezett n operandusa a 256 lehetséges periféria regiszter egyikét választja ki. Ugyanakkor 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

ffgvelet; (C)>>- (HL), B ← B-1, HL ← HL-1

Formátum: OP kód

OUTD

1	1	1	0	1	1	0	1
---	---	---	---	---	---	---	---

 ED

1	0	1	0	1	0	1	1
---	---	---	---	---	---	----------	---

 AB

Leírás:

A periféria regisztert közvetve, regiszter indirekt címzésig móddal címző **1/0** 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 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 megcímzett 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 MKz V.x.: 4.00

A jelzőbit regiszter érintett bitjei:

S: ismeretlen
Z: 1-es állapotba kéri, 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 D 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 1/0 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 FL tartalma által megcímezett memóriarekeszből kiolvasott, majd átmenetileg a CPU-ban tárolt byte a C regiszter tartalma által kijelölt periféria regiszterbe töltődik. Végül a FL 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/Y: ismeretlen
- N: 1-es állapotba kerül
- C: változatlan marad

Példa:

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

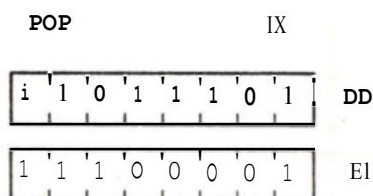
OUTI

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 Operandusok



Leírás:

Az utasítás az operatív memóriában elhelyezkedő LIPO 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 aktu á-
listetejére 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 55^a-t, és az 1001H című memóriarekesz 33H-t tartalmaz,
a

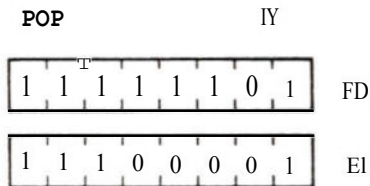
POP IX

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

POP IY

Művelet: $IY_{\wedge} \leftarrow (SP+1), IY_{\llcorner} \leftarrow (SP)$

Formátura: OP kód Operandusok



Leírás:

Az utagítás 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-
tejéremutató 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.

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

Példa:

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

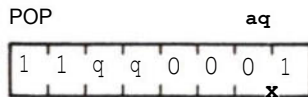
POP IY

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

POP qq

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

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	t*	
DE	01	D ^u	
HL	10	ff	
AF	11	M	I Sose <i>invol.</i> ftP PS _v

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

Példa:

Ja 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 3305H, 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 tetejéremutató 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

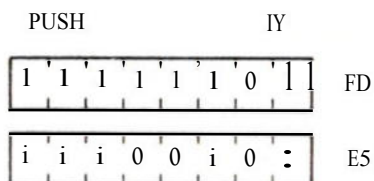
PUSH IX

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: $(SP-2) \leftarrow IYL, (SP-1) \leftarrow IYH$

Formátum: OP kód Operandusok



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.

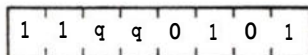
PUSH qq

Művelet: $(SP-2) \leftarrow qqL, (SP-1) \leftarrow qqH$

Formátum: OP kód Operandusok

PUSH

qq



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 <i>CS</i>
DE	01 <i>D r</i>
HL	10 <i>fS</i>
AP	11 <i>F\$ 1008 mem. VIM BW</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.

RÉS 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.
RÉS r	4	8/4,4/	2.00
RÉS (HL)	4	15/4,4,4,3/	3.75
RÉS (IX+d)	6	23/4,4,3,5,4,3/	5.75
RÉS CLY+d)	6	23/4,4,3,5,4,3/	5.75

Példa:

A

RÉS 6,D

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

RÉT

Művelet: $PCL \leftarrow (SP), PCH \leftarrow (SP+1)$

Formátum: OP kód

RÉT

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

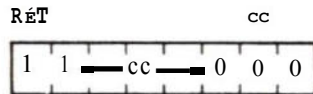
RÉT

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

RÉT cc

levellet: Pa cc ivaz: $PCL \leftarrow (SP)$, $PC_E \leftarrow (SP+1)$

Formátum: OP kód Operandumok



Leírás:

Ha a cc feltétel teljesül, az utasítás a vezérlést az eredeti Droaramnak adja vissza. Végrehajtása során az előzőleg CALL utasítással a veremtár tetejére mentett eredeti programszám-láló /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 helyi-értékű 8 bitjébe. /Az SP másodszor inkrementálódik/. A CPU következő gépi ciklusa a következő OP kódot már az ilyen mó-don a PC-ben jelenlévő cím által meghatározott memóriarekesz-ből kapja. Ha a cc feltétel nem teljesül, a PC a szokott mó-don 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>Actuális bit</u>	
CA	000	liZ nem zérus	Z	KAZ
C 2	001	Z zérus	Z	HZ
Ds	010	Nc nincs átvitel	C	R ^{Kc}
03	011	C átvitel	C	Re
El	100	PO páratlan paritás	P/V	V?0
Ei	101	PE páros paritás	P/V	R?t
F0	110	P pozitív előjel	S	Rf
FS	111	M negatív előjel	S	Rf

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

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

RÉT M

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

RÉTI

Fűvelet: Visszatérés programmegszakításból

Formátum: OP kód

RÉTI

1	1	1	0	1	1	0	1
---	---	---	---	---	---	---	---

 ED

0	1	0	0	1	1	0	1
---	---	---	---	---	---	---	---

 4D

Leírás:

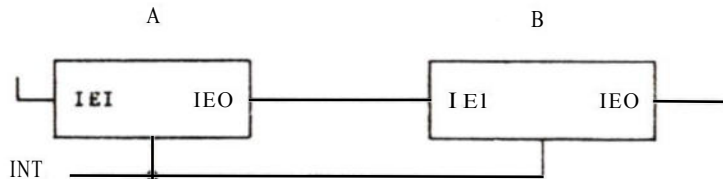
Az utasítást megszakítást kiszolgáló rutin végén használjuk azért, hogy

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

M ciklusok: 4 T állapotok: 14/4,4,3,3/ 4 FHz 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 enged élyező kimenete törlődik, és ezáltal mindaddig, amíg B kiszolgálása tart, az összes alacsonyabb nrrioritású megszakításkérés blokkolódik/. Ezután A kér programmegszakítást, ezáltal felfüggesztve B kiszolgálását. /A-nak az IEO programmegszakítást enged élyező kimenete törlődik, jelezve, hogy egy magasabb prioritású eszköz kiszolgálása van folyamatban/. Az A rutin befejezésekor végrehajtott RÉTI utasítás ismét 1-es állapotba helyezi A-nak a programmegszakítást enged élyező IEO kimenetét, és így lehetővé teszi, hogy a B-t kiszolgáló rutin folytatódjék. A B rutin befejezésekor kiadott újabb RÉTI utasítás a B eszköz megszakítást enged élyező 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.

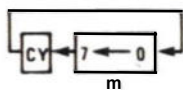
M ciklusok: 4 T állapotok: 14/4,4,3,3/ 4 MHz V.i.: 3.50

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

Típus:



Formátum:

OP kód

Operandusok

RL

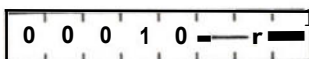
m

Az m operandus $r, (pL), (iX+dj)$ vagy $(iY+dj)$ bármelyike lehet, hasonlóan, mint az RLC utasításoknál. A lehetséges OP kód-operandus kombinációk tárgykódban a következők:

RL r

i	i	0	0	i	0	i	i
---	---	---	---	---	---	---	---

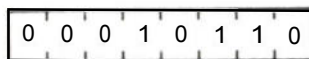
 CB



RL (HL)

1	1	0	0	1	0	1	1
---	---	---	---	---	---	---	---

 CB

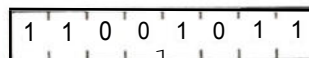


16

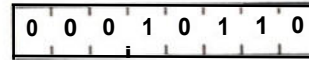
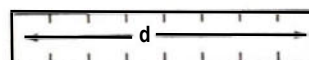
RL (IX - d)

1	1	0	1	1	1	0	1
---	---	---	---	---	---	---	---

 DD



CB

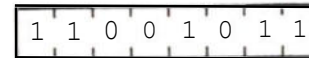


16

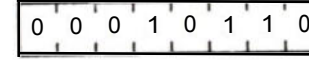
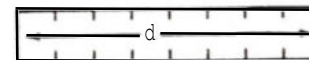
RL (iY + d)

1	1	1	1	1	1	0	1
---	---	---	---	---	---	---	---

 FD



CB



16

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

Regiszter:

<u>Regiszter:</u>	<u>r</u>	
P	000	U
C	001	14
D	010	11
E	011	rb
H	100	rh
L	101	15
A	111	Vt

RL m

Leírás:

Az m operandus tartalma e CY átvitel biten /a jelzőbit regiszter G 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
- K: 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

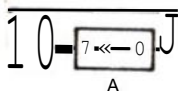
utasítás végrehajtásának eredményeként a D regiszter, és az átvitel bit tartalma a következő lesz:

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

RLA

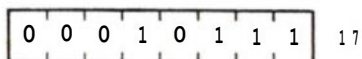
1!JC RAL

Művelet:



Formátum: OP kód

RLA



Leírás:

Az akkumulátor /A regiszter/ tartalma a CY átvitel biten /C bit/ keresztül egy helyiértékkel balra tolódik. A 0-ás bit tartalma az 1-es bitbe kerül, az 1-es bit előző tartalma pedig a P-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.

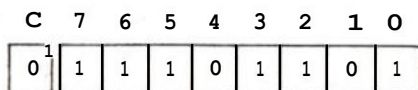
M ciklusok: 1 T állapdotok: 4 4 MHz V.i.: 1.00

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:

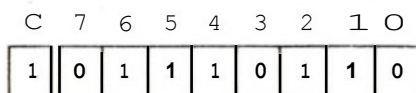
Ha az akkumulátor, és az átvitel bit tartalma



akkor az

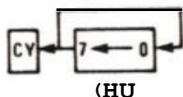
RLA

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



RLC (HL)

Művelet:



Formátum: OP kód Operandusok

RLC

(HL)

1	1	0	0	1	0	1	1
---	---	---	---	---	---	---	---

 CB

0	0	0	0	0	0	1	1	0	1
---	---	---	---	---	---	---	---	---	---

 06

Leírás:

A HL regiszterpár tartalma által specifikált memóriarekesz tartalma a CY átvitel bit /0 bit/ megkerülésével 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 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

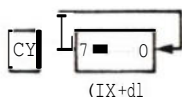
RLC (HL)

utaeítáp végrehajtásának eredményeként a 2S28H című memória-
rekeez, ép az átvitel bit tartalma a következő lesz:

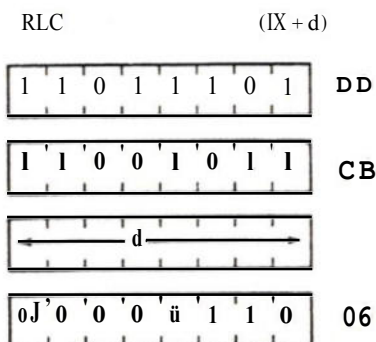
C	7	6	5	4	3	2	1	0
1	0	0	0	1	0	0	0	1

RLC (IX+d)

Művelet:



Formátum: OP kód Operandusok



Leíró?:

Az IX index regiszter tartalma, és a 2-es kompiernensű kódban értelmezett d egész szám összege által specifikált memóriarekesz tartalma a CY átvitel bit /0 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.: 5.73

A jelzóbit 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:

Ka 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

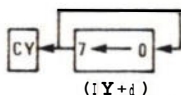
RLC (IX+2H)

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

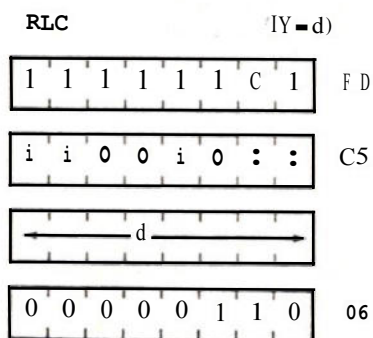
C	7	6	5	4	3	2	1	0
1	0	0	0	1	0	0	0	1

RLC (IY+d)

Művelet:



Formátum: OP kód Operandumok



Leírás:

Az IY inax 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.

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

A jelzobit 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

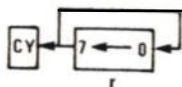
RLC (IY+2H)

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

C	7	6	5	4	3	2	1	0
<u>rn</u>	0	1	0	1	0	0	0	1

RLC r

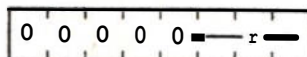
Művelet:



Formátum: OP kód Operandusok

RLC

r



Leírás

Az r operandus által meghatározott regiszter tartalma a CY átvitel hit 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>	L	
B	000	04
C	001	17
D	010	42
E	011	*5
H	100	^
L	101	05
A	111	0?

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

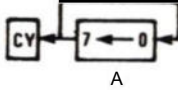
utasítás végrehajtásának eredményeként az r regiszter, és az átvitel bit tartalma a következő lesz:

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

RLCA

▷▷ PLC

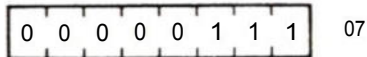
Művelet:



Formátum:

OP kód

RLCA



Leírás:

Az akkumulátor /A regiszter/ tartalma a CY átvitel bit megke-
rülésével egy helyiértékkel balra tolódik. A 0-ás bit tartal-
ma az 1-es bitbe kerül, az 1-es bit előző tartalma 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/.

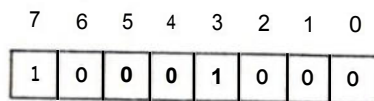
M ciklusok: 1 T állapotok: 4 4 MHz V.i.: 1.00

A jelzőbit regiszter érintett bitjei:

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

Példa:

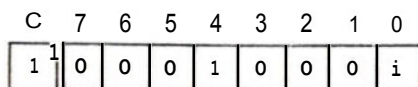
Ha az akkumulátor tartalma



akkor az

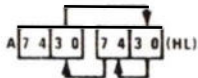
RLCA

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
							x

 ED

0	1	1	0	1	1	1	1
							x

 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 soecifikált 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

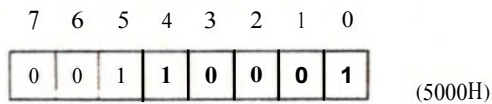
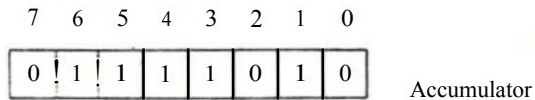
A jelzőbit regiszter érintett bitjei:

- 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:

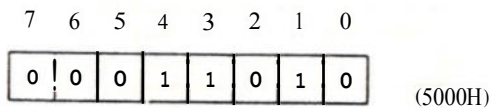
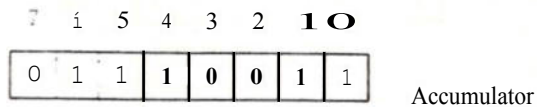
Ha a HL regiszterpár tartalma 5000H, az akkumulátor, és az 5000H című memóriarekesz tartalma pedig



akkor az

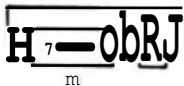
RLD

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



RR m

Művelet:



Formátum:

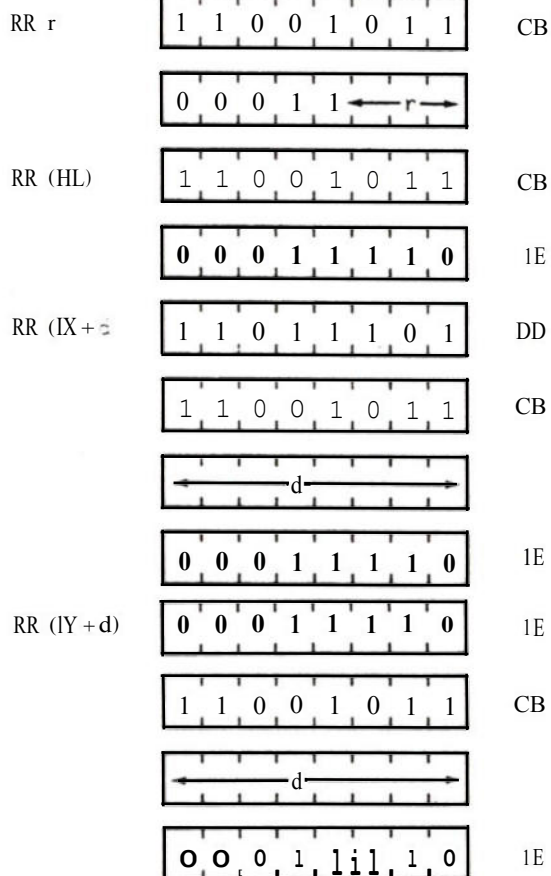
OP kód

Operandusok

RR

m

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



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

RR m

Regiszter	r	
E	000	4i
C	001	^
D	010	W
E	011	M
H	100	IC
L	101	W
A	111	Af

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	IKLUSOK	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: 7-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)

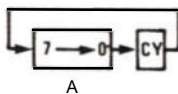
utasítás végrehajtásának eredményeként a 4343H című memóriarekesz, és az átvitel bit tartalma a következő lesz:

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

RRA

IWI KAR

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.

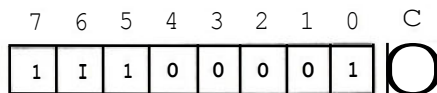
M ciklusok: 1 T állapotok: 4 4 T_{iz} V.i_„: 1.00

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:

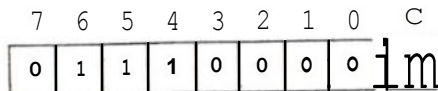
Ha az akkumulátor, és az átvitel bit tartalma



akkor az

RAA

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



RRC m

Regiszter	r	
L	010	W
E	011	W
H	100	t
L	101	#D
A	111	^

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	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+dj)	6	23/4,4,3,5,4,?/	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

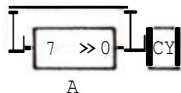
utasítás végrehajtásának eredményeként az A regiszter, és az átvitel bit tartalma a következő lesz:

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

RRCA

146ée P?EC

Művelet:



Formátum: OP kód

RRCA

0	0	0	0	1	1	1	1
---	---	---	---	---	---	---	---

 OF

Leírás:

Az akkumulátor /A regiszter/ tartalma a CY átvitel bit megke-
rülésével egy helyiértékkel jobbra tolódik. A 7-es bit tartal-
má 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.

✓ ciklusok: 1 T állapotok: 4 4 KHz 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

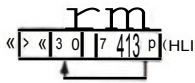
utasítás végrehajtásának eredményeként az akkumulátor, és az
átvitel bit tartalma a következő lesz:

7 6 5 4 3 2 1 0 C

1	C	0	0	1	0	0	0	1	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 ez 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.

K 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:

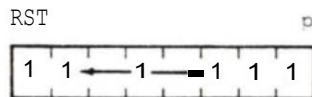
7 6 5 4 3 2 1 0
i 0 0 0 0 0 0 0 Accumulator

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

RST p

Művelet: (SP-1) ← PCH, (SP-2) ← PCL, PCH ← 0, PCL ← P

Formátum: OP kód Ooerandusok



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 00H értéket vesz fel, míg a PC alacsonyabb helyiértékű 8 bitjébe az alábbi táblázat p oszlopának megfelelő eleme másolódik:

P		t	
00H	<i>C1</i>	000	<i>RST0</i>
08H	<i>CF</i>	001	<i>gspü</i>
10H	3?	010	<i>R Syt</i>
18H	<i>Df</i>	011	<i>RST3</i>
20H	<i>E4</i>	100	<i>is7%</i>
28H	EP	101	<i>*SR*</i>
30H	<i>P*</i>	110	<i>£PRb</i>
38H	^	111	<i>∞*T></i>

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

UTASÍTÁS	M CIKLUSOK	T ÁLLAPOTOK	4 MHz V.io
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

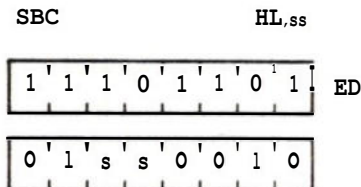
SBC A,(HL)

utasítás végrehajtásának eredményeként az akkumulátor 10H-t fog tartalmazni.

SBC HL,ss

Művelet: PL *- HL-ss-CY

Formátum: OP kód Operandusok



Leírás:

Az ss regiszterpár tartalma /a EC, 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 tárgyódban a következő lehet:

<u>Regiszterpár</u>	<u>ss</u>	
BC	00	eit
DE	01	sz
HL	10	fe
SP	11	^

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

A jelzobit 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 e HL regiszterpár tartalma 9999H, a DE regiszterpáré H11H, és ez 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

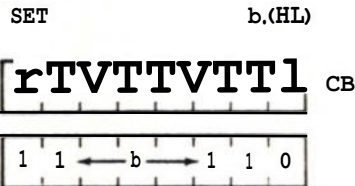
A jelzőbit regiszter érintett bitjei:

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

SET b.(HL)

Fűvelet: (HL)_Λ ← 1

Formátum: OP kód Operandusok



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

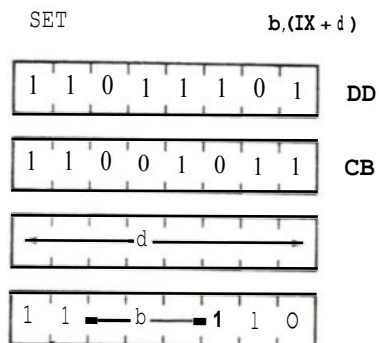
SET 4,(HL)

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 Operandusok



Leírás:

Az IX reáiszterpár /IX index regiszter/ tartalmának a d 2-es komplementű kódban adott emész számmal alkotott összege által kiemellett memóriarekesznek a b operandus által specifikált bit-
le 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:

W_a az index regiszter tartalma 2000H, a

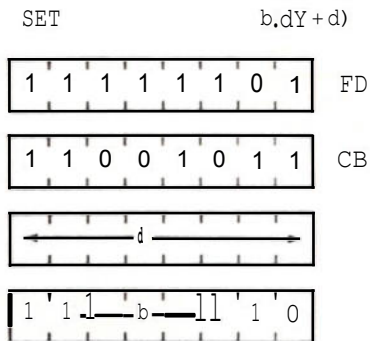
SET 0,(1X+3H)

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: $(1Y+d)_b \leftarrow 1$

Formátum: OP kód Operandusok



Leírás:

Az IY regiszterelőár /IY index regiszter/ tartalmának a d 2-es komplementű kódban adott érték számmal alkotott összege által kijelölt memóriarekesznek a b operansdus 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.: S075

Példr:

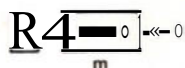
Ha az IY index regiszter tartalma 2000H, a

SET 0,(1Y+3H)

utasítás végrehajtásának eredményeként a 2003^A című memóriarekesz 0-ás bitje 1-es tartalmat vesz fel. /A 2003H című memóriarekeszben a 0-ás bit a legkisebb helyiértékű bit/.

SLA m

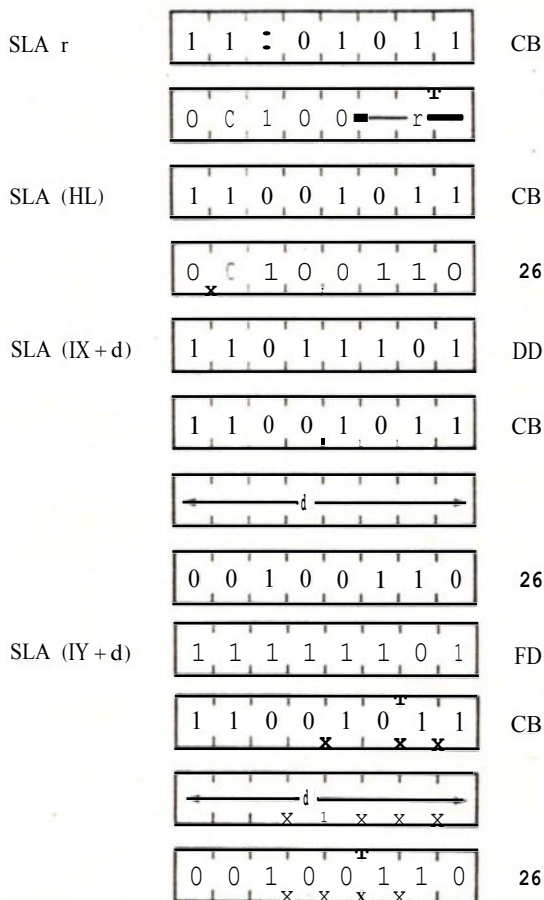
Művelet:



Formátum: OP kód Operandusok

SLA r

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>	X	
E	000	e✓
C	001	z✓
D	010	2 2
E	011	?3

SLA m

Regiszter	L	ZL
H	100	25
L	101	27
A	111	

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

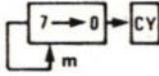
SLA

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

	C	7	6	5	4	3	2	1	0
1	H	1	1	0	0	0	0	1	0

SRA m

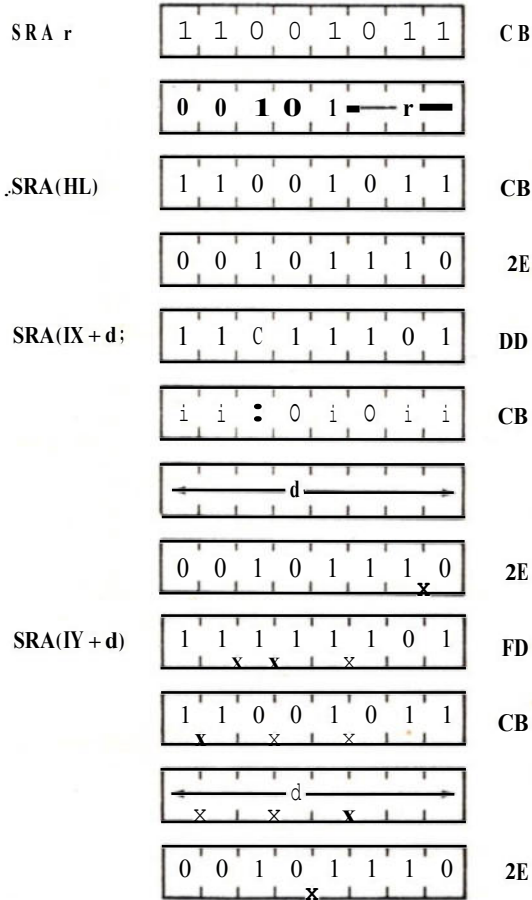
Művelet:



Formátum: OP kód Operandusok

SRA m

Az m operandus r, (üL), (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	L	
3	000	^
C	001	^
D	010-	M
E	011	M

SRA m

Regiszter	r
ii	100 re
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 ClX+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

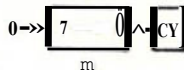
SRA (1X+3H)

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

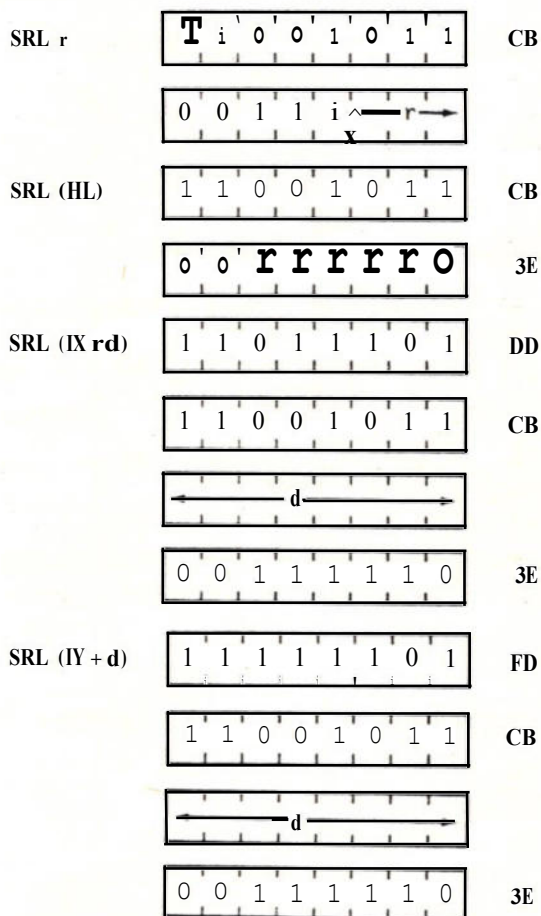
Művelet:



Formátum: OP kód Operandusok

 SRL r.

Az m operandus r, (HL), (iX+d) vagy (iY+d) bármelyike lehet, úgy, mint az RLC utasításoknál. A különböző lehetséges OP kód-operendus kombinációk az alábbiak:



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

<u>Regiszter</u>	<u>L</u>	
P	000	5f
C	001	◀◀
D	010	^

SRL m

Regiszter	r
E	011 3»
H	100 iC
L	101 90
A	111 iF

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	Ti	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 CLX+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
- Y: 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 E 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 HH-t tartalmaz, akkor a

SUB D

utasítás végrehajtásának eredményeként az akkumulátor 18H-t fog tartalmazni.

XOR s

1 ü64o nu *cod* XAA

Művelet: $A \star - A \odot s$

Formátum: OP kód Operandusok

XOR 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 az alábbiak:

XOR s		
XOR n		EE Xfci (iWC IWwJ)
XOR (HL)		AE
XOR (iX+c)		DD
		AE
XOR (iY+d)		FD
		AE

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

<u>Regiszter</u>	<u>Z</u>	
B	000	4i
C	001	A*
D	010	AA
E	011	M
H	100	Ac
L	101	Ae
A	111	AF

Leírás:

Az utasítás az s operandus által specifikált byte tartalma, és az akkumulátor regiszter tartalma között létesít bitenkénti KÍZÁRO VAGY kapcsolatot. Az eredmény az akkumulátorban tárolódik.

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.