

### 8-bit-Ladebefehle

	A	B	C	D	E	H	L	(HL)	(BC)	(DE)	(nn)	n
LD A, .	7F	78	79	7A	7B	7C	7D	7E	0A	1A	3AXXXX	3EXX
LD B, .	47	40	41	42	43	44	45	46				06XX
LD C, .	4F	48	49	4A	4B	4C	4D	4E				0EXX
LD D, .	57	50	51	52	53	54	55	56				16XX
LD E, .	5F	58	59	5A	5B	5C	5D	5E				1EXX
LD H, .	67	60	61	62	63	64	65	66				26XX
LD L, .	6F	68	69	6A	6B	6C	6D	6E				2EXX
LD (HL), .	77	70	71	72	73	74	75					36XX
LD (BC), .	02											
LD (DE), .	12											
LD (nn), .	32xxxx											

	A	B	C	D	E	H	L
LD ., (IX+d)	DD7EXX	DD46XX	DD4EXX	DD56XX	DD5EXX	DD66XX	DD6EXX
LD ., (IY+d)	FD7EXX	FD46XX	FD4EXX	FD56XX	FD5EXX	FD66XX	FD6EXX
LD (IX+d), .	DD77XX	DD70XX	DD71XX	DD72XX	DD73XX	DD74XX	DD75XX
LD (IY+d), .	FD77XX	FD70XX	FD71XX	FD72XX	FD73XX	FD74XX	FD75XX
LD (IX+d), n		DD36XXXX		LD (IY+d), n	FD36XXXX		

	S	Z	H	P/V	N	C
LD A, I	ED57	*	*	0	*	0 -
LD A, R	ED5F	*	*	0	*	0 -
LD I, A	ED47	-	-	-	-	-
LD R, A	ED4F	-	-	-	-	-

### 16-bit-Ladebefehle

	BC	DE	HL	SP	IX	IY
LD ., nn	01XXXX	11XXXX	21XXXX	31XXXX	DD21XXXX	FD21XXXX
LD ., (nn)	ED4Bxxxx	ED5Bxxxx	2AXXXX	ED7Bxxxx	DD2AXXXX	FD2AXXXX
LD (nn), .	ED43xxxx	ED53xxxx	22XXXX	ED73xxxx	DD22XXXX	FD22XXXX
LD SP			F9		DDF9	FDF9

	BC	DE	HL	AF	IX	IY
PUSH ..	C5	D5	E5	F5	DDE5	FDE5
POP ..	C1	D1	E1	F1	DDE1	FDE1

EX (SP), HL	E3		EX DE, HL	EB
EX (SP), IX	DDE3		EX AF, AF'	08
EX (SP), IY	FDE3		EXX	D9 BC<->BC', DE<->DE', HL<->HL'

### Blocktransfer- und Suchbefehle

	S	Z	H	P/V	N	C
LDI	EDA0	-	-	0	*	0 - LD (DE), (HL); INC HL; INC DE; DEC BC
LDIR	EDB0	-	-	0	0	0 - wie LDI, wiederholen bis BC=0
LDD	EDA8	-	-	0	*	0 - LD (DE), (HL); DEC HL; DEC DE; DEC BC
LDDR	EDB8	-	-	0	0	0 - wie LDD, wiederholen bis BC=0
CPI	EDA1	*	*	*	*	1 - CP A, (HL); INC HL; DEC BC
CPIR	EDB1	*	*	*	*	1 - wie CPI, wiederh. bis BC=0 oder A=(HL)
CPD	EDA9	*	*	*	*	1 - CP A, (HL); DEC HL; DEC BC
CPDR	EDB9	*	*	*	*	1 - wie CPD, wiederh. bis BC=0 oder A=(HL)

### Sprungbefehle

	Z	NZ	C	NC	PE	PO	M	P
JP	CAXXXX	C2XXXX	DAXXXX	D2XXXX	EAXXXX	E2XXXX	FAXXXX	F2XXXX
CALL	CCXXXX	C4XXXX	DCXXXX	D4XXXX	ECXXXX	E4XXXX	FCXXXX	F4XXXX
RET	C8	C0	D8	D0	E8	E0	F8	F0
JR	28XX	20XX	38XX	30XX				

	unbedingt	(HL)	(IX)	(IY)	
JP	C3XXXX	E9	DDE9	FDE9	
CALL	CDXXXX				
RET	C9				
JR	18XX				
					00 08 10 18 20 28 30 38
					RST C7 CF D7 DF E7 EF F7 FF
DJNZ	10XX	DEC B; JR NZ, d			
RETI	ED4D				
RETN	ED45				

### CPU-Steuerbefehle

	S	Z	H	P/V	N	C
NOP	00	-	-	-	-	-
HALT	76	-	-	-	-	-
CCF	3F	-	-	*	-	0 *
SCF	37	-	-	0	-	0 1
EI	FB	-	-	-	-	-
DI	F3	-	-	-	-	-
IM 0	ED46	-	-	-	-	-
IM 1	ED56	-	-	-	-	-
IM 2	ED5E	-	-	-	-	-

Komplementiere Carry-Flag  
Setze Carry-Flag  
Interrupts freigeben  
Interrupts sperren  
Interrupt-Modus 0  
Interrupt-Modus 1  
Interrupt-Modus 2

### Ein-/Ausgabebefehle

	A	B	C	D	E	H	L	S	Z	H	P/V	N	C
IN ., (C)	ED78	ED40	ED48	ED50	ED58	ED60	ED68	*	*	0	*	0	-
OUT (C), .	ED79	ED41	ED49	ED51	ED59	ED61	ED69	-	-	-	-	-	-

	S	Z	H	P/V	N	C
IN A, n	DBXX	-	-	-	-	-
OUT n, A	D3XX	-	-	-	-	-
INI	EDA2	?	*	?	?	1 - IN (HL), (C); INC HL; DEC B
INIR	EDB2	?	1	?	?	1 - wie INI, wiederholen bis B=0
IND	EDA4	?	*	?	?	1 - IN (HL), (C); DEC HL; DEC B
INDR	EDB4	?	1	?	?	1 - wie IND, wiederholen bis B=0
OUTI	EDA3	?	*	?	?	1 - OUT (C), (HL); INC HL; DEC B
OTIR	EDB3	?	1	?	?	1 - wie OUTI, wiederholen bis B=0
OUTD	EDAB	?	*	?	?	1 - OUT (C), (HL); DEC HL; DEC B
OTDR	EDBB	?	1	?	?	1 - wie OUTD, wiederholen bis B=0

### 8-bit Arithmetische und Logische Befehle

	B	D	D	E	H	L (HL)	A	n	(IX+d)	(IY+d)	S	Z	H	P/V	N	C	
ADD .	80	81	82	83	84	85	86	87	C6XX	DD86XX	FD86XX	*	*	*	*	0	*
ADC .	88	89	8A	8B	8C	8D	8E	8F	CEXX	DD8EXX	FD8EXX	*	*	*	*	0	*
SUB .	90	91	92	93	94	95	96	97	D6XX	DD96XX	FD96XX	*	*	*	*	1	*
SBC .	98	99	9A	9B	9C	9D	9E	9F	DEXX	DD9EXX	FD9EXX	*	*	*	*	1	*
AND .	A0	A1	A2	A3	A4	A5	A6	A7	E6XX	DDA6XX	FDA6XX	*	*	1	*	0	0
XOR .	A8	A9	AA	AB	AC	AD	AE	AF	EEXX	DDAEXX	FDAEXX	*	*	1	*	0	0
OR .	B0	B1	B2	B3	B4	B5	B6	B7	F6XX	DDB6XX	FDB6XX	*	*	1	*	0	0
CP .	B8	B9	BA	BB	BC	BD	BE	BF	FEXX	DDB6XX	FDB6XX	*	*	*	*	1	*
INC .	04	0C	14	1C	24	2C	34	3C		DD34XX	FD34XX	*	*	*	*	0	-
DEC .	05	0D	15	1D	25	2D	35	3D		DD35XX	FD35XX	*	*	*	*	1	-

	S	Z	H	P/V	N	C		
DAA	27	*	*	*	*	1	*	BCD-Korrektur im Akku
CPL	2F	*	*	*	*	1	*	Komplementiere Akku (2er-Komplement)
NEG	ED44	*	*	*	*	1	*	Komplementiere Akku (2er-Komplement)

### 16-bit Arithmetische und Logische Befehle

	BC	DE	HL	SP	IX	IY	S	Z	H	P/V	N	C
INC .	03	13	23	33	DD23	FD23	-	-	-	-	-	-
DEC .	0B	1B	2B	3B	DD2B	FD2B	-	-	-	-	-	-
ADD HL, .	09	19	29	39			-	-	*	-	0	*
ADC HL, .	ED4A	ED5A	ED6A	ED7A			*	*	*	*	0	*
SDC HL, .	ED42	ED52	ED62	ED72			*	*	*	*	1	*
ADD IX, .	DD09	DD19		DD39	DD29		-	-	*	-	0	*
ADD IX, .	FD09	FD19		FD39	FD29		-	-	*	-	0	*

### Rotations- und Schiebepfehle

	B	D	D	E	H	L	(HL)	A	(IX+d)	(IY+d)
RR	CB18	CB19	CB1A	CB1B	CB1C	CB1D	CB1E	CB1F	DDCBXX1E	FDDBXX1E
RL	CB10	CB11	CB12	CB13	CB14	CB15	CB16	CB17	DDCBXX16	FDDBXX16
RRC	CB08	CB09	CB0A	CB0B	CB0C	CB0D	CB0E	CB0F	DDCBXX0E	FDDBXX0E
RLC	CB00	CB01	CB02	CB03	CB04	CB05	CB06	CB07	DDCBXX06	FDDBXX06
SRA	CB28	CB29	CB2A	CB2B	CB2C	CB2D	CB2E	CB2F	DDCBXX2E	FDDBXX2E
SLA	CB20	CB21	CB22	CB23	CB24	CB25	CB26	CB27	DDCBXX26	FDDBXX26
SRL	CB38	CB39	CB3A	CB3B	CB3C	CB3D	CB3E	CB3F	DDCBXX3E	FDDBXX3E

	S	Z	H	P/V	N	C	
RR/RL	*	*	0	*	0	*	Rotiere Reg rechts/links durch Carry
RRC/RLC	*	*	0	*	0	*	Rotiere Reg rechts/links
SRA/SLA	*	*	0	*	0	*	Shift Reg rechts/links arithmetisch
SRL	*	*	0	*	0	*	Shift Reg rechts logisch

	S	Z	H	P/V	N	C		
RRCA	0F	-	-	0	-	0	*	Rotiere Akku rechts
RLCA	07	-	-	0	-	0	*	Rotiere Akku links
RRA	1F	-	-	0	-	0	*	Rotiere Akku rechts durch Carry
RLA	17	-	-	0	-	0	*	Rotiere Akku linksh Carry
RLD(HL)	ED6F	*	*	0	*	0	-	Rotiere Rotiere Digit zw. Akku und (HL)
RRD(HL)	ED6F	*	*	0	*	0	-	Rotiere Rotiere Digit zw. Akku und (HL)

### Einzelbitbefehle

	B	D	D	E	H	L	(HL)	A	(IX+d)	(IY+d)
BIT 0	CB40	CB41	CB42	CB43	CB44	CB45	CB46	CB47	DDCBXX46	FDDBXX46
BIT 1	CB48	CB49	CB4A	CB4B	CB4C	CB4D	CB4E	CB4F	DDCBXX4E	FDDBXX4E
BIT 2	CB50	CB51	CB52	CB53	CB54	CB55	CB56	CB57	DDCBXX56	FDDBXX56
BIT 3	CB58	CB59	CB5A	CB5B	CB5C	CB5D	CB5E	CB5F	DDCBXX5E	FDDBXX5E
BIT 4	CB60	CB61	CB62	CB63	CB64	CB65	CB66	CB67	DDCBXX66	FDDBXX66
BIT 5	CB68	CB69	CB6A	CB6B	CB6C	CB6D	CB6E	CB6F	DDCBXX6E	FDDBXX6E
BIT 6	CB70	CB71	CB72	CB73	CB74	CB75	CB76	CB77	DDCBXX76	FDDBXX76
BIT 7	CB78	CB79	CB7A	CB7B	CB7C	CB7D	CB7E	CB7F	DDCBXX7E	FDDBXX7E
RES 0	CB80	CB81	CB82	CB83	CB84	CB85	CB86	CB87	DDCBXX86	FDDBXX86
RES 1	CB88	CB89	CB8A	CB8B	CB8C	CB8D	CB8E	CB8F	DDCBXX8E	FDDBXX8E
RES 2	CB90	CB91	CB92	CB93	CB94	CB95	CB96	CB97	DDCBXX96	FDDBXX96
RES 3	CB98	CB99	CB9A	CB9B	CB9C	CB9D	CB9E	CB9F	DDCBXX9E	FDDBXX9E
RES 4	CBA0	CBA1	CBA2	CBA3	CBA4	CBA5	CBA6	CBA7	DDCBXXA6	FDDBXXA6
RES 5	CBA8	CBA9	CBAA	CBAB	CBAC	CBAD	CBAE	CBAF	DDCBXXAE	FDDBXXAE
RES 6	CBB0	CBB1	CBB2	CBB3	CBB4	CBB5	CBB6	CBB7	DDCBXXB6	FDDBXXB6
RES 7	CBB8	CBB9	CBBA	CBBB	CBBC	CBBD	CBBE	CBBF	DDCBXXBE	FDDBXXBE
SET 0	CBC0	CBC1	CBC2	CBC3	CBC4	CBC5	CBC6	CBC7	DDCBXXC6	FDDBXXC6
SET 1	CBC8	CBC9	CBCA	CBCB	CBCC	CBCE	CBCF		DDCBXXCE	FDDBXXCE
SET 2	CBD0	CBD1	CBD2	CBD3	CBD4	CBD5	CBD6	CBD7	DDCBXXD6	FDDBXXD6
SET 3	CBD8	CBD9	CBDA	CBDB	CBDC	CBDD	CBDE	CBDF	DDCBXXDE	FDDBXXDE
SET 4	CBE0	CBE1	CBE2	CBE3	CBE4	CBE5	CBE6	CBE7	DDCBXXE6	FDDBXXE6
SET 5	CBE8	CBE9	CBEA	CBEB	CBEC	CBED	CBEE	CBEF	DDCBXXEE	FDDBXXEE
SET 6	CBF0	CBF1	CBF2	CBF3	CBF4	CBF5	CBF6	CBF7	DDCBXXF6	FDDBXXF6
SET 7	CBF8	CBF9	CBFA	CBFB	CBFC	CBFD	CBFE	CBFF	DDCBXXFE	FDDBXXFE

Flagbeeinflussung:	S	Z	H	P/V	N	C
BIT	?	*	1	?	0	-
SET	-	-	-	-	-	-
RES	-	-	-	-	-	-

### Flag-Register:

Bit	7	6	5	4	3	2	1	0
	S	Z	X	H	X	P/V	N	C

	gesetzt	nicht gesetzt	wird bei
C Carry-Flag	C	NC	Übertrag von Bit 7
N Add-/Subtract-Flag			Subtraktionen
P/V Parity-/Overflow-Flag	PE	PO	gerader Parität
H Half-Carry-Flag			Übertrag von Bit 3
Z Zero-Flag	Z	NZ	Ergebnis 0
S Sign-Flag	M	P	neg. Ergebnis
X nicht verwendet			

Beeinflussung:	1 gesetzt	0 zurückgesetzt
	*	abhängig vom Ergebnis einer Operation
	-	nicht beeinflusst
	?	unbestimmt