

1.11. Мнемоника команд микропроцессоров Intel 8080 и Z80



Мнемоника		Действие	CZPSNH	Дл.	Вр.	Комментарии	Примечания
INTEL 80	Z80						
ADC r	ADC A,r	A:=A+r+CY	**V*0*	1	4	r ≡ A,B,C,D,E,H,L	Add with Carry to Acc
ADC M	ADC A,(HL)	A:=A+(HL)+CY		1	7		
ACI,n	ADC A,n	A:=A+n+CY		2	7	n - byte (0..FF)	
—	ADC A,(ii+n)	A:=A+(ii+n)+CY		3	19	ii=IX,IY	Add register + Carry to[HL]
—	ADC HL,rr	HL:=HL+rr+CY	**V*0x	2	15	rr ≡ BC,DE,HL,SP	
ADD r	ADD A,r	A:=A+r	**V*0*	1	4		Add
ADD M	ADD A,(HL)	A:=A+(HL)		1	7		
ADI n	ADD A,n	A:=A+n		2	7		
—	ADD A,(ii+n)	A:=A+(ii+n)		3	19		
DAD rr	ADD HL,rr	HL:=HL+rr	*...0x	1	11		
—	ADD IX,ry	IX:=IX+px		2	15	ry ≡ BC,DE,SP,IY	IY, IX — 16 bits
—	ADD IY,rx	IY:=IY+py		2	15	rx ≡ BC,DE,SP,IX	
ANA r	AND r	A:=A and r	0*P*01	1	4		Logical AND
ANA m	AND (HL)	A:=A and (HL)		1	7		
ANI n	AND n	A:=A and n		2	7		
—	AND (ii+n)	A:=A and (ii+n)		3	19		
—	BIT b,r	Z:=not rб	.*xx01	2	8	b — номер бита (0÷7)	Test BIT
—	BIT b,(HL)	Z:=not (HL)б		2	12	xб — bit b of	
—	BIT b,(ii+n)	Z:=not (ii+n)б		4	20	location x	
CALL nn	CALL nn	PUSH PC;PC:=nn		3	17	nn - word (0..FFFF)	Call subroutin
—	CALL cc,nn	Если cc, то CALL nn иначе продолжить		3	17	cc ≡ C,NC,Z,NZ,M,P,PE,PO	Call conditional
CC nn	CALL C,nn				10		
CM nn	CALL M,nn						
CNC nn	CALL NC,nn						
CNZ nn	CALL NZ,nn						
CP nn	CALL P,nn						
CPE nn	CALL PE,nn						
CPO nn	CALL PO,nn						
CZ nn	CALL Z,nn						
CMC	CCF	CY:=not CY	*...0x	1	4	CoMplement Carry Flag	
CPM r	CP r	A-r	**V*1*	1	4		Compare
CPM m	CP (HL)	A-(HL)		1	7		
CPI n	CP n	A-n		2	7		
—	CP (ii+n)	A-(ii+n)		3	19		

Мнемоника		Действие	CZPSNH	Дл.	Вр.	Комментарии	Примечания
INTEL 80	Z80						
—	CPD	A-(HL);dec HL;dec BC	.***1*	2	16	PV=0 если BC=0 иначе PV=1	Compare, decrem
—	CPDR	Повторять CPD, пока Z=1 or BC=0	.***1*	2	21 16		Compare, decrement, repeat
—	CPI	A-(HL);inc HL;dec BC	.***1*	2	16	PV=0 если BC=0 иначе PV=1	Compare, decrem
—	CPIR	Повторять CPI, пока Z=1 or BC=0	.***1*	2	21 16		Compare, decrement, repeat
CMA	CPL	A:=A xor 255	11	1	4		Complement Acc
DAA	DAA		**P*,*	1	4		Dec. Adj. Acc.
DCR r	DEC r	r:=r-1	.V*1*	1	4		Decrement
DCM M	DEC (HL)	(HL):=(HL)-1		1	11		
—	DEC (ii+n)	(ii+n):=(ii+n)-1		3	23		
DCX rr	DEC rr	rr:=rr-1		1	6		
—	DEC ii	ii:=ii-1		2	10		
DI	DI	IFF:=0		1	4	Disable interrupts	
—	DJNZ e	dec B; если B≠0 JR e если B=0 продолжать		2	13 8	e — относительный адрес	Decrement & jump if Not 0
EI	EI	IFF:=1		1	4	Enable interrupts	

Оригинал

 **Fix Me!**  **Delete!**

INTEL 80	Мнемоника Z80	Действие	CZPSNH	Дл.	Вр.	Комментарии	Примечания
ADC r	ADC A, r	A:=A+r+CY	**V*0*	1	4	r ≡ A,B,C,D,E,H,L	Add with Carry
ADC M	ADC A, (HL)	A:=A+(HL)+CY		1	7		to Acc
ACI, n	ADC A, n	A:=A+n+CY		2	7	n - byte (0..FF)	
-	ADC A, (ii+n)	A:=A+(ii+n)+CY		3	19	ii=IX,IY	Add register +
-	ADC HL, rr	HL:=HL+rr+CY	**V*0x	2	15	rr ≡ BC,DE,HL,SP	Carry to[HL]
ADD r	ADD A, r	A:=A+r	**V*0*	1	4		
ADD M	ADD A, (HL)	A:=A+(HL)		1	7		Add

	ADI n	ADD A,n	A:=A+n		2	7		
	-	ADD A,(ii+n)	A:=A+(ii+n)		3	19		
	DAD rr	ADD HL,rr	HL:=HL+rr	*...0x	1	11		
	-	ADD IX,ry	IX:=IX+px		2	15	ry ≡ BC,DE,SP,IY	IY \ 16
	-	ADD IY,rx	IY:=IY+py		2	15	rx ≡ BC,DE,SP,IX	IX /
bits								
AND	ANA r	AND r	A:=A and r	0*P*01	1	4		
	ANA m	AND (HL)	A:=A and (HL)		1	7		Logical
	ANI n	AND n	A:=A and n		2	7		
	-	AND (ii+n)	A:=A and (ii+n)		3	19		
BIT	-	BIT b,r	Z:=not r6	.*xx01	2	8	b - номер бита (0÷7)	
	-	BIT b,(HL)	Z:=not (HL)6		2	12	x6 - bit b of	Test
	-	BIT b,(ii+n)	Z:=not (ii+n)6		4	20	location x	
CALL subroutin	CALL nn	CALL nn	PUSH PC;PC:=nn		3	17	nn - word (0..FFFF)	Call
	-	CALL cc,nn	Если cc, то CALL nn		3	17	cc ≡ C,NC,Z,NZ,M,P,PE,	Call
conditional	CC nn	CALL C,nn	иначе продолжить			10		P0
	CM nn	CALL M,nn						
	CNC nn	CALL NC,nn						
	CNZ nn	CALL NZ,nn						
	CP nn	CALL P,nn						
	CPE nn	CALL PE,nn						
	CPO nn	CALL P0,nn						
	CZ nn	CALL Z,nn						
	CMC	CCF	CY:=not CY	*...0x	1	4		CoMplement Carry Flag
Compare	CPM r	CP r	A - r	**V*1*	1	4		
	CPM m	CP (HL)	A - (HL)		1	7		
	CPI n	CP n	A - n		2	7		
	-	CP (ii+n)	A - (ii+n)		3	19		

		CPD	A - (HL); dec HL; dec BC	***1*	2	16	PV=0 если BC=0 иначе PV=1	Compare, decrem
		CPDR	Повторять CPD ,	***1*	2	21		Compare, decre-
		пока Z=1 or BC=0		16			ment, repeat	
		CPI	A - (HL); inc HL; dec BC	***1*	2	16	PV=0 если BC=0 иначе PV=1	Compare, decrem
		CPIR	Повторять CPI ,	***1*	2	21		Compare, decre-
		пока Z=1 or BC=0		16			ment, repeat	
	CMA	CPL	A := A xor 255	11	1	4		Complement Acc
DAA	DAA		**P*.*	1	4		Dec. Adj.	Acc.
DCR r	DEC r	r := r - 1	. *V*1*	1	4			
DCM M	DEC (HL)	(HL) := (HL) - 1		1	11			Decrement
-	DEC (ii+n)	(ii+n) := (ii+n) - 1		3	23			
DCX rr	DEC rr	rr := rr - 1		1	6			
-	DEC ii	ii := ii - 1		2	10			
DI	DI	IFF:=0		1	4		Disable interrupts	
-	DJNZ e	dec B; если B≠0 JR e		2	13	e - относительный	Decrement	&
	если B=0 продолжать		8	адрес		jump if Not 0		
EI	EI	IFF:=1		1	4		Enable interrupts	

Intel 80	Мнемоника Z80	Действие	CZPSNH	Дл.	Вр.	Комментарии	Примечания
-	EX AF, AF'	AF $\leftrightarrow$ AF'			1	4	
XCHG	EX DE, HL	DE $\leftrightarrow$ HL			1	4	
EXchange	XTHL	EX (SP), HL	(SP) $\leftrightarrow$ HL		1	4	

-	EX (SP),ii	(SP) $\leftrightarrow$ ii		2	23		
-	EXX	BC $\leftrightarrow$ BC';DE $\leftrightarrow$ DE'; HL $\leftrightarrow$ HL'		1	4	Exchange Register pair & Alternatives	
HLT	HALT	Останов центр.проц.		1	4		
-	IM 1	Режим прерываний 1		2	8		
-	IM 2	Режим прерываний 2					
-	IM 3	Режим прерываний 3					
IN n	IN A,(n)	A:=port(n)		2	11		
-	IN r,(C)	r:=port(C)	.*P*0*	2	12		
-	IN ?, (C)	Только устанавливает флаги как IN r,(C)	.*P*0*	2	12	Не является командой	INput
			(код ED 70)				
INR r	INC r	r:=r+1	.*V*0*	1	4		
INR M	INC (HL)	(HL):=(HL)+1		1	11		
-	INC (ii+n)	(ii+n):=(ii+n)+1		3	23		
INCrement	INX rr	rr:=rr+1		1	6		
-	INC ii	ii:=ii+1		2	10		
-	IND	(HL):=port(C); dec HL;dec B	x*xx1x	2	16	Z=1 если B=0, иначе Z=0	INput,Decre- ment
-	INDR	Повторять IND , пока B=0	x1xx1x	2	21	ment,Repeat	INput,Decre- ment
-	INI	(HL):=port(C); inc HL;dec B	x*xx1x	2	16	Z=1 if B=0,else Z=0	INput,Incre- ment
-	INIR	Повторять INI , пока B=0	x1xx1x	2	21	ment,Repeat	INput,Incre- ment
JNP nn	JP nn	PC:=nn		3	10		

p	JC nn	JP C,nn							
	JM nn	JP M,nn							
	JNC nn	JP NC,nn							
	JP nn	JP P,nn							
	JNZ nn	JP NZ,nn							J u m
	JPE nn	JP PE,nn							
	JPO nn	JP PO,nn							
	JZ nn	JP Z,nn							
	-	JP cc,nn	Если cc, то JP nn		3	10			
	PCHL	JP (HL)	PC:=HL		1	4			
jump	-	JP (ii)	PC:=ii		2	8			
	-	JR e	PC:=PC+e		2	12			Relative
	-	JR cond,e	Если cond, то JR e		2	12	cond ≡ C,NC,Z,NZ		
		иначе NOP			7				
d	MOV r,r	LD r,r	r:=r		1	4			
	MOV r,M	LD r,(HL)	r:=(HL)		1	7			
	MVI r,n	LD r,n	r:=n		2	7			
	-	LD r,(ii+n)	r:=(ii+n)		3	19			
	MOV M,r	LD (HL),r	(HL):=r		1	7			
	-	LD (ii+n),r	(ii+n):=r		3	19			
	MVI M,n	LD (HL),n	(HL):=n		2	10			
	-	LD (ii+n),n	(ii+n):=n		4	19			
	LDAX B	LD A,(BC)	A:=(BC)		1	7			L o a
	LDAX D	LD A,(DE)	A:=(DE)		1	7			
	LDA nn	LD A,(nn)	A:=(nn)		3	13			
	STAX B	LD (BC),A	(BC):=A		1	7			
	STAX D	LD (DE),A	(DE):=A		1	7			
	STA nn	LD (nn),A	(nn):=A		3	13			
	-	LD A,I	A:=I	..***00	2	9	PV=IFF		
	-	LD A,R	A:=R		2	9	PV=IFF		
	-	LD I,A	I:=A		2	9			

-	LD R, A	R:=A		2	9		
INTEL 80	Мнемоника Z80	Действие	CZPSNH	Дл.	Вр.	Комментарии	Примечания
LXIrr, nn	LD rr, nn	rr:=nn			3	10	
---	LD ii, nn	ii:=nn			4	14	
LHLD nn	LD HL, (nn)	HL:=(nn)			3	16	
---	LD rr, (nn)	rr:=(nn)			4	20	
---	LD ii, (nn)	ii:=(nn)			4	20	
SHLD nn	LD (nn), HL	(nn):=HL			3	16	
---	LD (nn), rr	(nn):=rr			4	20	
---	LD (nn), ii	(nn):=ii			4	20	
SPHL	LD SP, HL	SP:=HL			1	6	
---	LD SP, ii	SP:=ii			1	10	
---	LDD	(DE):=(HL);			2	16	PV=0 if BC=0, else
PV=1	load, decrement	dec DE, HL, BC					
---	LDDR	Повторять LDD			2	21	
load, decrement	до Z=1 or BC=0			16		repead	
---	LDI	(DE):=(HL);			2	16	PV=0 if BC=0, else
PV=1	load, increment	inc DE, HL; dec BC					
---	LDIR	Повторять LDI			2	21	
load, increment	до Z=1 or BC=0			16		repead	
---	NEG	A:=0-A	**V*1*	2	8		Negate Acc
NOP	NOP	Отсутствие операции		1	4		
ORA r	OR r	A:=A or r	0*P*00	1	4		Logical
OR							

ORA M	OR (HL)	A:=A or (HL)		1	7	
ORI n	OR n	A:=A or n		2	7	
---	OR (ii+n)	A:=A or (ii+n)		3	19	
Output, decre-	OTDR	Повторять OUTD	x1xx1x	2	21	
	до B=0			16		ment, repeat
Output, incre-	OTIR	Повторять OUTI	x1xx1x	2	21	
	до B=0			16		ment, repeat
OUT n	OUT (n), A	port(n):=A		2	11	Output
---	OUT (C), r	port(C):=r		2	12	
Output, decre-	OUTD	port(C):=(HL);	x*xx1x	2	16	Z=1 if B=0, else Z=0
		dec HL; dec B				ment
out, increment	OUTI	port(C):=(HL);	x*xx1x	2	16	Z=1 if B=0, else Z=0
		inc HL; dec B				
POP gg stack	POP qq	qq:=(SP); SP:=SP+2		1	10	qq ≡ AF, BC, DE, HL
	POP ii	ii:=(SP); SP:=SP+2		2	14	Pop from
PUSH gg stack	PUSH qq	SP:=SP-2; (SP):=qq		1	11	Push to
	PUSH ii	SP:=SP-2; (SP):=ii		2	15	
Bit	RES b, r	r6:=0		2	8	
	RES b, (HL)	(HL)6:=0		2	15	Reset
	RES b, (ii+n)	(ii+n)6:=0		4	23	
RET from subroutine	RET	POP PC		1	10	Return
	RET cc	Если cc то RET				
	RET c	иначе NOP		1	11	
RC	RET c				5	
RM	RET M					Return

conditional	RNC	RET NC							
	RNZ	RET NZ							
	RP	RET P							
	RPE	RET PE							
	RPO	RET PO							
	RZ	RET Z							
---	RETI	Возврат из прерыван.		2	14	Return from interrupt			
INTEL 80	Мнемоника Z80	Действие	CZPSNH	Дл.	Вр.	Комментарии	Примечания		
---	RETN	Возвр.из немаск.прер.		2	14	Return from Non-mascable			
left	RL r		**P*00	2	8	Rotate			
Carry	RL (HL)			2	15	through			
---	RL (ii+n)			4	23				
RAL	RLA		*...00	1	4	Rotate Acc. left throgh			
Carry									
left	RLC r		**P*00	2	8	Rotate			
---	RLC (HL)			2	15	circular			
---	RLC (ii+n)			4	23				
RLC	RLCA		*...00	1	4	Rotate Acc. left circular			
Acc &	RLD	A	.P*00	2	18	Rotate			
left						Memore			
						decimal			

right	RR r		**P*00	2	8	Rotate
	RR (HL)			2	15	through
	RR (ii+n)			4	23	Carry
RAR	RRA		*...00	1	4	Rotate Acc. right through
Carry						
	RRC r		**P*00	2	8	Rotate
	RRC (HL)			2	15	right
	RRC (ii+n)			4	23	circular
RRC	RRCA		*...00	1	4	Rotate Acc & Memory right
Decimal						
Acc &	RRD		.P*00	2	18	Rotate
						memory
						Right
						Decimal
RST adr	RST adr	CALL adr		1	11	Restart
SBB r	SBC A,r	A:=A-r-CY	**V*1*	1	4	Subtract
SBB M	SBC A,(HL)	A:=A-(HL)-CY		1	7	with
SBI n	SBC A,n	A:=A-n-CY		2	7	Carry
---	SBC A,(ii+n)	A:=A-(ii+n)-CY		3	19	(borrow)
---	SBC HL,rr	HL:=HL-rr-CY	**V*1x	2	15	
STC	SCF	CY:=1	1...00	1	4	Set Carry
Flag						
---	SET b,r	r6:=1		2	8	
---	SET b,(HL)	(HL)6:=1		2	15	Set bit
---	SET b,(ii+n)	(ii+n)6:=1		4	23	
---	SLA r		**P*00	2	8	

---	SLA (HL)	CY ← 7 ← 0 < 0	2 15	Shift left Arithmetic
---	SLA (ii+n)	L _ _ _ _ _ L _ _ _ _ _	4 23	
---	SRA r	r _ _ _ _ _ r _ _ _ _ _	**P*00 2 8	
---	SRA (HL)	7 → 0 CY	2 15	Shift right Arithmetic
---	SRA (ii+n)	L _ _ _ _ _ L _ _ _ _ _	4 23	
right	SRL r	r _ _ _ _ _ r _ _ _ _ _	**P*00 2 8	Shift
---	SRL (HL)	0 7 → 0 CY	2 15	Logical
---	SRL (ii+n)	L _ _ _ _ _ L _ _ _ _ _	4 23	
Subtract	SUB r	SUB r	A:=A-r	**V*0* 1 4
	SUB M	SUB (HL)	A:=A-(HL)	1 7
	SVI n	SUB n	A:=A-n	2 7
	---	SUB (ii+n)	A:=A-(ii+n)	3 19
XOR	XRA r	XOR r	A:=A xor r	0*P*00 1 4
	XRA M	XOR (HL)	A:=A xor (HL)	1 7
	XRI n	XOR n	A:=A xor n	2 7
	---	XOR (ii+n)	A:=A xor (ii+n)	3 19

😊 Мнемоника команд микропроцессоров Intel 8080 и Z80 приведены в книге [Центральный процессор Z80CPU](#), в разделе [Приложение Б. Соответствие мнемоник ассемблера Z80 и I8080](#)

https://sysadminmosaic.ru/msx/basic_dialogue_programming_language/111

2023-02-22 21:06

