

Op Code Table

0	1	2	3	4	5	6	7
00 NOP	** LXI B ++	STAX B	INX B	* INR B	* DCR B	MVI B +	** RLC
01 —	DAD B	LDAX B	DCX B	* INR C	* DCR C	MVI C +	** RRC
02 —	** LXI D ++	STAX D	INX D	* INR D	* DCR D	MVI D +	** RAL
03 —	DAD D	LDAX D	DCX D	* INR E	* DCR E	MVI E +	** RAR
04 —	** LXI H ++	SHLD ++	INX H	* INR H	* DCR H	MVI H +	DAA
05 —	DAD H	LHLD ++	DCX H	* INR L	* DCR L	MVI L +	CMA
06 —	** LXI SP ++	STA ++	INX SP	* INR M	* DCR M	MVI M +	** STC
07 —	DAD SP	LDA ++	DCX SP	* INR A	* DCR A	MVI A +	** CMC
0	1	2	3	4	5	6	7
10 MOV B,B	MOV B,C	MOV B,D	MOV B,E	MOV B,H	MOV B,L	MOV B,M	MOV B,A
11 MOV C,B	MOV C,C	MOV C,D	MOV C,E	MOV C,H	MOV C,L	MOV C,M	MOV C,A
12 MOV D,B	MOV D,C	MOV D,D	MOV D,E	MOV D,H	MOV D,L	MOV D,M	MOV D,A
13 MOV E,B	MOV E,C	MOV E,D	MOV E,E	MOV E,H	MOV E,L	MOV E,M	MOV E,A
14 MOV H,B	MOV H,C	MOV H,D	MOV H,E	MOV H,H	MOV H,L	MOV H,M	MOV H,A
15 MOV L,B	MOV L,C	MOV L,D	MOV L,E	MOV L,H	MOV L,L	MOV L,M	MOV L,A
16 MOV M,B	MOV M,C	MOV M,D	MOV M,E	MOV M,H	MOV M,L	HLT	MOV M,A
17 MOV A,B	MOV A,C	MOV A,D	MOV A,E	MOV A,H	MOV A,L	MOV A,M	MOV A,A
0	1	2	3	4	5	6	7
20 ADD B	ADD C	ADD D	ADD E	ADD H	ADD L	ADD M	ADD A
21 ADC B	ADC C	ADC D	ADC E	ADC H	ADC L	ADC M	ADC A
22 SUB B	SUB C	SUB D	SUB E	SUB H	SUB L	SUB M	SUB A
23 SBB B	SBB C	SBB D	SBB E	SBB H	SBB L	SBB M	SBB A
24 ANA B	ANA C	ANA D	ANA E	ANA H	ANA L	ANA M	ANA A
25 XRA B	XRA C	XRA D	XRA E	XRA H	XRA L	XRA M	XRA A
26 ORA B	ORA C	ORA D	ORA E	ORA H	ORA L	ORA M	ORA A
27 CMP B	CMP C	CMP D	CMP E	CMP H	CMP L	CMP M	CMP A
0	1	2	3	4	5	6	7
30 RNZ	POP B	JNZ ++	JMP ++	CNZ ++	PUSH B	ADI +	RST 0
31 RZ	RET	JZ ++	—	CZ ++	CALL ++	ACI +	RST 1
32 RNC	POP D	JNC ++	OUT +	CNC ++	PUSH D	SUI +	RST 2
33 RC	—	JC ++	IN +	CC ++	—	SBI +	RST 3
34 RPO	POP H	JPO ++	XTHL	CPO ++	PUSH H	ANI +	RST 4
35 RPE	PCHL	JPE ++	XCHG	CPE ++	—	XRI +	RST 5
36 RP	POP PSW	JP ++	DI	CP ++	PUSH PSW	ORI +	RST 6
37 RM	SPHL	JM ++	EI	CM ++	—	CPI +	RST 7

Red Op Code indicates all flags affected

Black Op Code Indicates No Flags Affected

(each + indicates an extra instruction byte)

*All Flags Except Carry Affected

**Only Carry Flags Affected

MEMORY BLOCKS

DEC	OCTAL	DEC	OCTAL
8K	040	40K	240
16K	100	48K	300
24K	140	56K	340
32K	200		

FLAG REGISTER

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
S	Z	0	AC	0	P	1	C
0 = \bar{S} , \bar{Z}		0 = \bar{AC}			2 = \bar{P} , \bar{C}		
1 = Z		2 = AC			3 = C		
2 = S					6 = P		
3 = S, Z					7 = P, C		

AC = AUX CARRY

S = SIGN

Z = ZERO

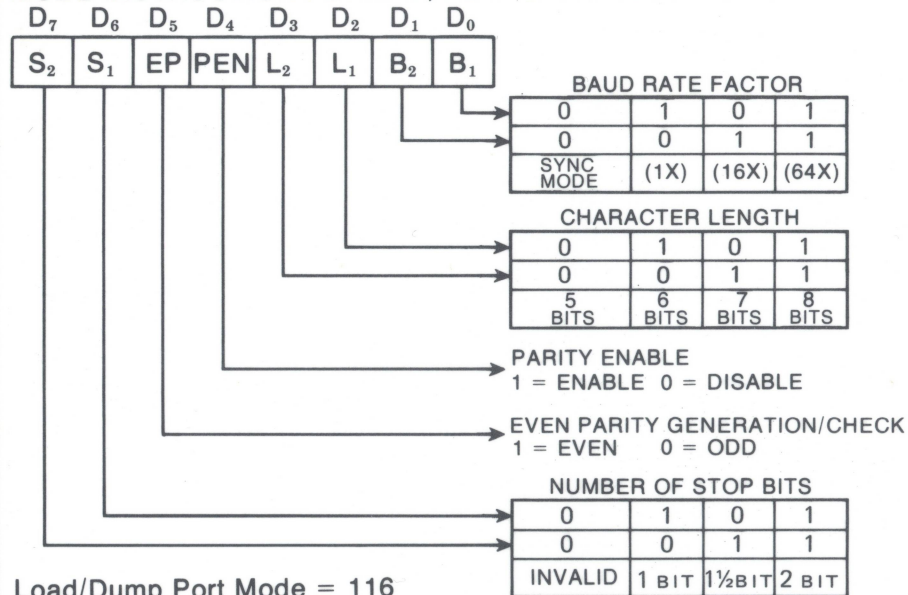
C = CARRY

P = PARITY

INSTRUCTION SET

Mnemonic	Description	Mnemonic	Description
ACI	Add immediate to A with carry	MVI M	Move immediate memory
ADC M	Add memory to A with carry	MVI r	Move immediate register
ADC r	Add register to A with carry	MOV M, r	Move register to memory
ADD M	Add memory to A	MOV r, M	Move memory to register
ADD r	Add register to A	MOV r1, r2	Move register to register
ADI	Add immediate to A	NOP	No-operation
ANA M	And memory with A	ORA M	Or memory with A
ANA r	And register with A	ORA r	Or register with A
ANI	And immediate with A	ORI	Or immediate with A
CALL	Call unconditional	OUT	Output
CC	Call on carry	PCHL	H & L to program counter
CM	Call on minus	POP B	Pop register pair B & C off stack
CMA	Complement A	POP D	Pop register pair D & E off stack
CMC	Complement carry	POP H	Pop register pair H & L off stack
CMP M	Compare memory with A	POP PSW	Pop A and Flags off stack
CMP r	Compare register with A	PUSH B	Push register Pair B & C on stack
CNC	Call on no carry	PUSH D	Push register Pair D & E on stack
CNZ	Call on no zero	PUSH H	Push register Pair H & L on stack
CP	Call on positive	PUSH PSW	Push A and Flags on stack
CPE	Call on parity even	RAL	Rotate A left through carry
CPI	Compare immediate with A	RAR	Rotate A right through carry
CPO	Call on parity odd	RC	Return on carry
CZ	Call on zero	RET	Return
DAA	Decimal adjust A	RLC	Rotate A left
DAD B	Add B & C to H & L	RM	Return on minus
DAD D	Add D & E to H & L	RNC	Return on no carry
DAD H	Add H & L to H & L	RNZ	Return on no zero
DAD SP	Add stack pointer to H & L	RP	Return on positive
DCR M	Decrement memory	RPE	Return on parity even
DCR r	Decrement register	RPO	Return on parity odd
DCX B	Decrement B & C	RRC	Rotate A right
DCX D	Decrement D & E	RST	Restart
DCX H	Decrement H & L	RZ	Return on zero
DCX SP	Decrement stack pointer	SBB M	Subtract memory from A with borrow
DI	Disable Interrupt	SBB r	Subtract register from A with borrow
EI	Enable Interrupts	SBI	Subtract immediate from A with borrow
HLT	Halt	SHLD	Store H & L direct
IN	Input	SPHL	H & L to stack pointer
INR M	Increment memory	STA	Store A direct
INR r	Increment register	STAX B	Store A indirect
INX B	Increment B & C registers	STAX D	Store A indirect
INX D	Increment D & E registers	STC	Set carry
INX H	Increment H & L registers	SUB M	Subtract memory from A
INX SP	Increment stack pointer	SUB r	Subtract register from A
JC	Jump on carry	SUI	Subtract immediate from A
JM	Jump on minus	XCHG	Exchange D & E, H & L Registers
JMP	Jump unconditional	XRA M	Exclusive Or memory with A
JNC	Jump on no carry	XRA r	Exclusive Or register with A
JNZ	Jump on no zero	XRI	Exclusive Or immediate with A
JP	Jump on positive	XTHL	Exchange top of stack, H & L
JPE	Jump on parity even		
JPO	Jump on parity odd		
JZ	Jump on zero		
LDA	Load A direct		
LDAX B	Load A indirect		
LDAX D	Load A indirect		
LHLD	Load H & L direct		
LXI B	Load immediate register Pair B & C		
LXI D	Load immediate register Pair D & E		
LXI H	Load immediate register Pair H & L		
LXI SP	Load immediate stack pointer		

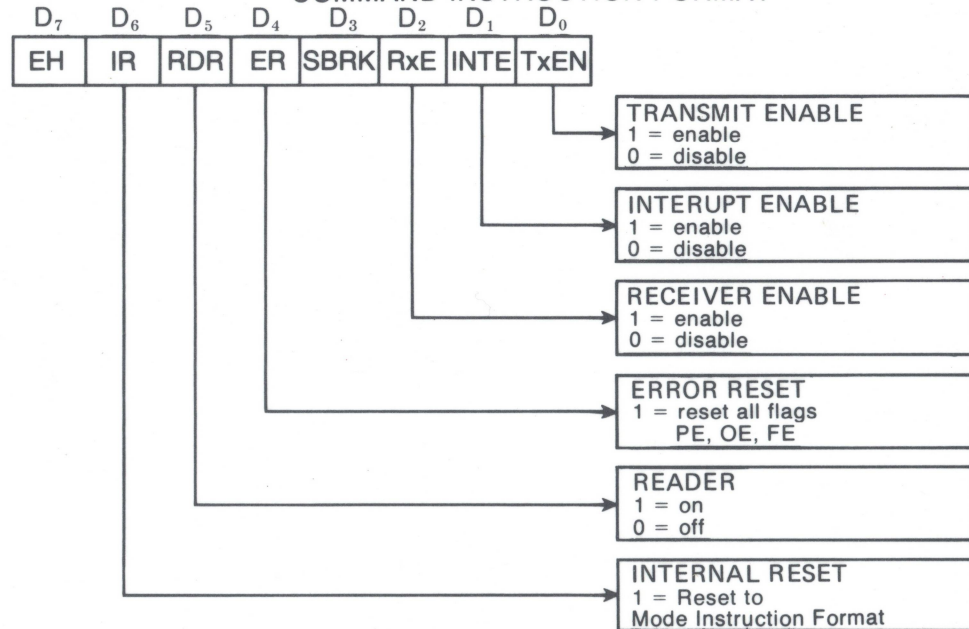
MODE INSTRUCTION FORMAT, ASYNCHRONOUS MODE



Load/Dump Port Mode = 116

IN/OUT INSTRUCTIONS

COMMAND INSTRUCTION FORMAT



Status Read Format

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
DSR	SYNDET	FE	OE	PE	TxE	RxRDY	TxRDY

I/O Port Assignment

370 = Data } Load/
371 = Mode/Command } Dump

372 = Data } System
373 = Mode/Command } Console

OCTAL ASCII TABLE

000 NUL	020 DLE	040 SP	060 0	100 @	120 P	140 `	160 p
001 SOH	021 DC1 (X-ON)	041 !	061 1	101 A	121 Q	141 a	161 q
002 STX	022 DC2 (TAPE)	042 "	062 2	102 B	122 R	142 b	162 r
003 ETX	023 DC3 (X-OFF)	043 #	063 3	103 C	123 S	143 c	163 s
004 EOT	024 DC4 (TAPE)	044 \$	064 4	104 D	124 T	144 d	164 t
005 ENQ	025 NAK	045 %	065 5	105 E	125 U	145 e	165 u
006 ACK	026 SYN	046 &	066 6	106 F	126 V	146 f	166 v
007 BEL	027 ETB	047 '	067 7	107 G	127 W	147 g	167 w
010 BS	030 CAN	050 (070 8	110 H	130 X	150 h	170 x
011 HT	031 EM	051)	071 9	111 I	131 Y	151 i	171 y
012 LF	032 SUB	052 *	072 :	112 J	132 Z	152 j	172 z
013 VT	033 ESC	053 +	073 ;	113 K	133 [153 k	173 {
014 FF	034 FS	054 ,	074 <	114 L	134 \	154 l	174
015 CR	035 GS	055 -	075 =	115 M	135]	155 m	175 }
016 SO	036 RS	056 .	076 >	116 N	136 ^ (↑)	156 n	176 ~
017 SI	037 US	057 /	077 ?	117 O	137 _ (←)	157 o	177 DEL

IN/OUT

I/O Data

I/O Port Number

MEM # RTM/Ø, Enter Data, Port #

PC 6 OUT

MEM # RTM/Ø, Enter 000, Port #

HL 5 IN

DUMP

Program Entry Point

REG PC ALTER / RST/Ø

MEM # RTM/Ø

Start of Dump

040 000

040 001

ALTER / RST/Ø

Low Byte

Hi Byte

End of Dump

MEM # RTM/Ø

HI BYTE

LOW BYTE

9 DUMP