UNIVERSITI SAINS MALAYSIA

Peperiksaan Semester Kedua
Sidang Akademik 1992/93

April 1993

ZSE 416/4 - Pengantar Mikropemproses/Mikrokomputer

Masa : (3 jam)

Sila pastikan bahawa kertas peperiksaan ini mengandungi LIMA muka surat yang bercetak sebelum anda memulakan peperiksaan ini.

Jawab kesemua LIMA soalan.
Kesemuanya wajib dijawab di dalam Bahasa Malaysia.

1. (a) (i) Tukar nombor-nombor desimal -127 dan -128 kepada nombor-nombor pelengkap duaan.

(ii) Tukar nombor-nombor pelengkap duaan 11111011 dan 01110111 kepada nombor-nombor desimal yang setara.

(20/100)

(b) Dengan menggunakan pelengkap duaan, cari jawapan dalam 8 bit bagi nombor-nombor desimal berikut:

(i) (+20) + (-60)
(ii) (-3) + (-4)
(iii) (+3) - (+8)
(iv) (+89) + (-46)

(40/100)

(c) Cari nombor oktal dan heksadesimal setara bagi 10101010.01010101.

(10/100)

(d) Jelaskan mengapa komputasi atau pengiraan dalam mikropemproses tidak menggunakan sistem desimal?

(30/100)

2. (a) Pernyataan Boolean, F bagi menyatakan secara ringkas tentang output daripada get dan litar logik adalah

$$F = A.\overline{B}.\overline{C}.D + A.B.C.D$$

(i) Lakarkan litar logik untuk F.

(ii) Dapatkan jadual kebenaran bagi litar tersebut.

(25/100)

(b) Huraikan 3 (tiga) jenis bas yang terlibat dalam semua pemindahan data bagi suatu mikropemproses.

(35/100)

(c) Terangkan apakah yang dimaksudkan dengan teknik input/output. Huraikan 3 (tiga) teknik asas bagi input/output.

(40/100)

3. (a) Papan mata Intel SDK-85 ditunjukkan dalam gambarajah di bawah:

RESET	VECT INTR	C	D	E	F
SINGLE STEP	GO	8 H	9 L	A	B
SUBST MEM	EXAM REG	4 SPH	5 SPL	6 PCH	7 PCL
NEXT ,	EXEC .	0	1	2	3 I

Nyatakan apakah fungsi kunci-kunci berikut:

(i) RESET
(ii) SUBST MEM
(iii) EXAM REG
(iv) SINGLE STEP

(30/100)

(b) Dengan menunjukkan langkah-langkah dalam memperolehi jawapan anda, apakah kandungan akumulator selepas mengeksekusikan arahan-arahan berikut:

(i) ADD E
DAA
} Jika nilai awal: [A] = 46_{16} dan [E] = 14_{16}

(ii) ANI B) Jika nilai awal: [A] = 57_{16}

(iii) CMP E) Jika nilai awal: [A] = 42_{16} dan [E] = 27^{16}

(30/100)

(c) (i) Program di bawah menunjukkan suatu program dalam bahasa mesin. Dengan menggunakan Kad Rujukan Bahasa Asembli Intel 8085/8080, tukarkan bahasa mesin ini ke bahasa asembli.

Alamat (Heks)	Kandungan (Heks)
2000	01
2001	00
2002	20
2003	2A
2004	00
2005	20
2006	09
2007	CF

(ii) Cari kandungan pasangan daftar H, L selepas mengeksekusikan program dalam c(i).

(40/100)

4. (a) Tunjukkan lokasi semua flag di dalam daftar status dan terangkan bagaimana flag-flag ini bertindak.

(20/100)

(b) Nyatakan setiap flag di dalam daftar status samada terpengaruh atau tidak dengan arahan-arahan 8085 berikut:

(i) STC
(ii) LHLD
(iii) ORA
(iv) OUT

(20/100)

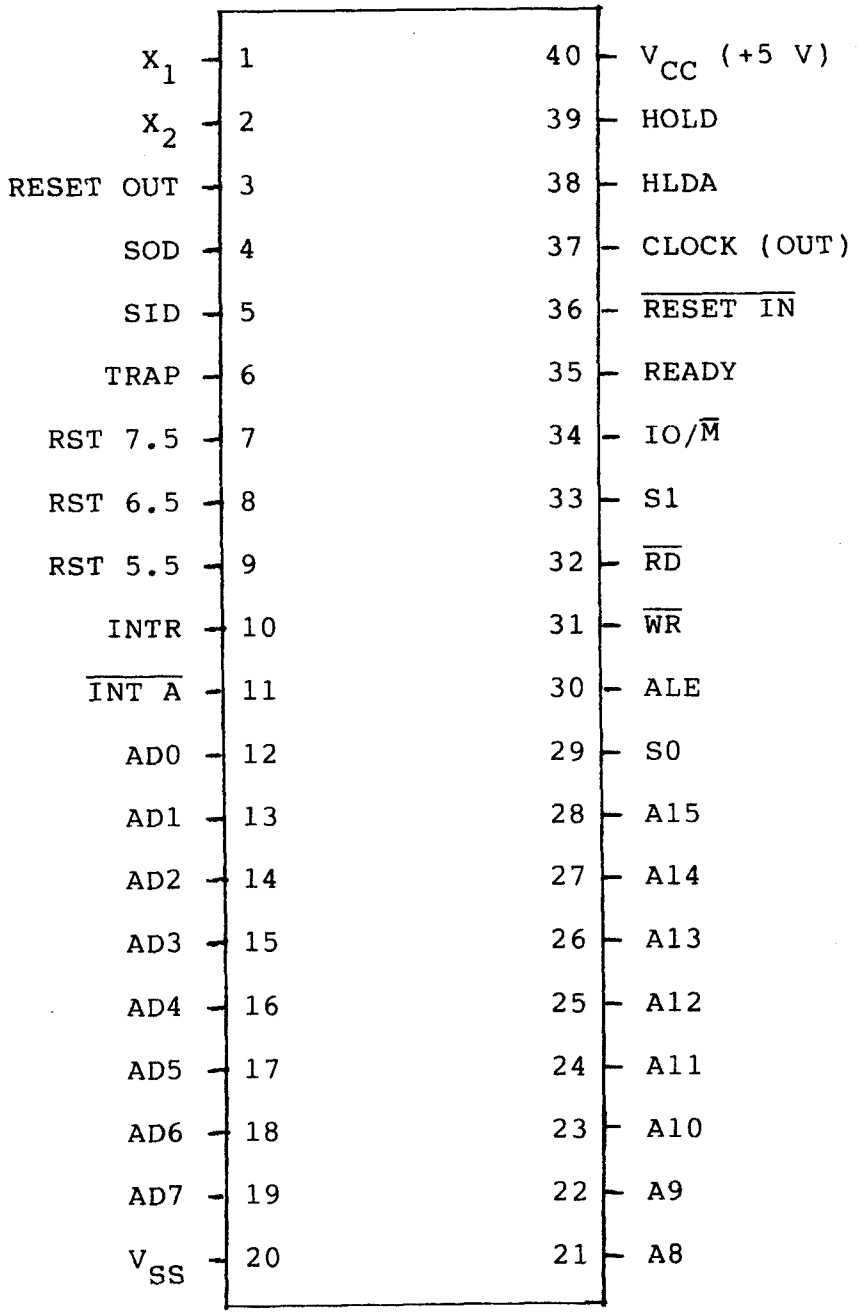
(c) (i) Lakarkan satu carta aliran terperinci bagi membanding dua nombor heksadesimal dan menstorkan nombor terkecil di dalam lokasi ingatan 2040H.

(ii) Berdasarkan carta aliran di dalam soalan c(i), tuliskan program di dalam bahasa asembli 8085.

(60/100)

5. (a) 8-bit mikropemproses Intel 8085 adalah ditempatkan di dalam 40-pin DIP (dual-in-line package) atau pakej bertalian dual). Gambarajah pinout bagi mikropemproses 8085 ditunjukkan dalam gambarajah di bawah:

...4/-



Gambarajah Pin Mikropemproses 8085

Nyatakan apakah pin-pin berikut dan jenis (samada input, output atau dwiarah) pin-pin berkenaan:

...5/-

(i) Pin 3 dan 36
(ii) Pin 4 dan 5
(iii) pin 7, 8 dan 9
(iv) pin 10 dan 11
(v) pin 12-19
(vi) pin 21-28.

(20/100)

(b) Huraikan mengenai mod-mod pengalamatan yang lazimnya tersedia dalam suatu mikropemproses. Berikan 1 (satu) contoh op kod bagi setiap mod pengalamatan yang dihuraikan.

(55/100)

(c) Jelaskan mengenai bas antara muka IEEE 488 dan apakah parameter-parameter komunikasi yang disetkan bagi piawai ini.

(25/100)

- oooOooo -

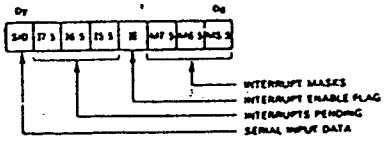
RESTART TABLE

Name	Code	Restart Address
RST 0	C7	0000_{16}
RST 1	CF	0008_{16}
RST 2	D7	0010_{16}
RST 3	DF	0018_{16}
RST 4	E7	0020_{16}
TRAP	Hardware* Function	0024_{16}
RST 5 *	EF	0028_{16}
RST 5.5	Hardware* Function	$002C_{16}$
RST 6	F7	0030_{16}
RST 6.5	Hardware* Function	0034_{16}
RST 7	FF	0038_{16}
RST 7.5	Hardware* Function	$003C_{16}$

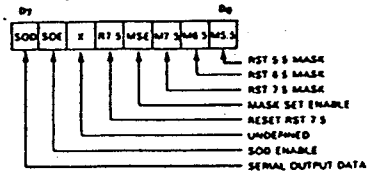
*NOTE The hardware functions refer to the on-chip interrupt feature of the 8085 only

USE OF THE A REGISTER BY RIM AND SIM INSTRUCTIONS (8085 ONLY)

A REGISTER AFTER EXECUTING RIM



A REGISTER BEFORE EXECUTING SIM



00	NOP		28	DCX	H	56	MOV	D,M	81	ADD	C		AC	XRA	H
01	LXI	B,dble	2C	INR	L	57	MOV	D,A	82	ADD	D		AD	XRA	L
02	STAX	B	2D	DCR	L	58	MOV	E,B	83	ADD	E		AE	XRA	M
03	INX	B	2E	MVI	L,byte	59	MOV	E,C	84	ADD	H		AF	XRA	A
04	INR	B	2F	CMA		5A	MOV	E,D	85	ADD	L		B0	ORA	B
05	DCR	B	30	SIM*		5B	MOV	E,E	86	ADD	M		B1	ORA	C
06	MVI	B,byte	31	LXI	SP,dble	5C	MOV	E,H	87	ADD	A		B2	ORA	D
07	RLC		32	STA	adr	5D	MOV	E,L	88	ADC	B		B3	ORA	E
08	---		33	INX	SP	5E	MOV	E,M	89	ADC	C		B4	ORA	H
09	DAD	B	34	INR	M	5F	MOV	E,A	8A	ADC	D		B5	ORA	L
0A	LDAX	B	35	DCR	M	60	MOV	H,B	8B	ADC	E		B6	ORA	M
0B	DCX	B	36	MVI	M,byte	61	MOV	H,C	8C	ADC	H		B7	ORA	A
0C	INR	C	37	STC		62	MOV	H,D	8D	ADC	L		B8	CMP	B
0D	DCR	C	38	---		63	MOV	H,E	8E	ADC	M		B9	CMP	C
0E	MVI	C,byte	39	DAD	SP	64	MOV	H,H	8F	ADC	A		BA	CMP	D
0F	RRC		3A	LDA	adr	65	MOV	H,L	90	SUB	B		BB	CMP	E
10	---		3B	DCX	SP	66	MOV	H,M	91	SUB	C		BC	CMP	H
11	LXI	D,dble	3C	INR	A	67	MOV	H,A	92	SUB	D		BD	CMP	L
12	STAX	D	3D	DCR	A	68	MOV	L,B	93	SUB	E		BE	CMP	M
13	INX	D	3E	MVI	A,byte	69	MOV	L,C	94	SUB	H		BF	CMP	A
14	INR	D	3F	CMC		6A	MOV	L,D	95	SUB	L		C0	RNZ	
15	DCR	D	40	MOV	B,B	6B	MOV	L,E	96	SUB	M		C1	POP	B
16	MVI	D,byte	41	MOV	B,C	6C	MOV	L,H	97	SUB	A		C2	JNZ	
17	RAL		42	MOV	B,D	6D	MOV	L,L	98	SBB	B		C3	JMP	
18	---		43	MOV	B,E	6E	MOV	L,M	99	SBB	C		C4	CNZ	
19	DAD	D	44	MOV	B,H	6F	MOV	L,A	9A	SBB	D		C5	PUSH	B
1A	LDAX	D	45	MOV	B,L	70	MOV	M,B	9B	SBB	E		C6	ADI	
1B	DCX	D	46	MOV	B,M	71	MOV	M,C	9C	SBB	H		C7	RST	0
1C	INR	E	47	MOV	B,A	72	MOV	M,D	9D	SBB	L		C8	RZ	
1D	DCR	E	48	MOV	C,B	73	MOV	M,E	9E	SBB	M		C9	RET	
1E	MVI	E,byte	49	MOV	C,C	74	MOV	M,H	9F	SBB	A		CA	JZ	
1F	RAR		4A	MOV	C,D	75	MOV	M,L	A0	ANA	B		CB	---	
20	RIM*		4B	MOV	C,E	76	HLT		A1	ANA	C		CC	CZ	
21	LXI	H,dble	4C	MOV	C,H	77	MOV	M,A	A2	ANA	D		CD	CALL	
22	SHLD	adr	4D	MOV	C,L	78	MOV	A,B	A3	ANA	E		CE	ACI	
23	INX	H	4E	MOV	C,M	79	MOV	A,C	A4	ANA	H		CF	RST	1
24	INR	H	4F	MOV	C,A	7A	MOV	A,D	A5	ANA	L		D0	RNC	
25	DCR	H	50	MOV	D,B	7B	MOV	A,E	A6	ANA	M		D1	POP	D
26	MVI	H,byte	51	MOV	D,C	7C	MOV	A,H	A7	ANA	A		D2	JNC	
27	DAA		52	MOV	D,D	7D	MOV	A,L	A8	XRA	B		D3	OUT	
28	---		53	MOV	D,E	7E	MOV	A,M	A9	XRA	C		D4	CNC	
29	DAD	H	54	MOV	D,H	7F	MOV	A,A	AA	XRA	D		D5	PUS	
2A	LHLD	adr	55	MOV	D,L	80	ADD	B	AB	XRA	E		D6	SUI	

*8085 Only

DATA TRANSFER GROUP

Move

MOV	A,A	7F
	A,B	78
	A,C	79
	A,D	7A
	A,E	7B
	A,H	7C
	A,L	7D
	A,M	7E
MOV	B,A	47
	B,B	40
	B,C	41
	B,D	42
	B,E	43
	B,H	44
	B,L	45
	B,M	46
MOV	C,A	4F
	C,B	48
	C,C	49
	C,D	4A
	C,E	4B
	C,H	4C
	C,L	4D
	C,M	4E
MOV	D,A	57
	D,B	50
	D,C	51
	D,D	52
	D,E	53
	D,H	54
	D,L	55
	D,M	56

Move (cont)

MOV	E,A	5F
	E,B	58
	E,C	59
	E,D	5A
	E,E	5B
	E,H	5C
	E,L	5D
	E,M	5E
MOV	H,A	67
	H,B	60
	H,C	61
	H,D	62
	H,E	63
	H,H	64
	H,L	65
	H,M	66
MOV	L,A	6F
	L,B	68
	L,C	69
	L,D	6A
	L,E	6B
	L,H	6C
	L,L	6D
	L,M	6E
MOV	M,A	77
	M,B	70
	M,C	71
	M,D	72
	M,E	73
	M,H	74
	M,L	75

XCHG EB

Move Immediate

MVI	A, byte	3E
	B, byte	06
	C, byte	0E
	D, byte	16
	E, byte	1E
	H, byte	26
	L, byte	2E
	M, byte	36

Load Immediate

LXI	B, dble	01
	D, dble	11
	H, dble	21
	SP, dble	31

Load Store

LDAX B	0A
LDAX D	1A
LHLD adr	2A
LDA adr	3A
STAX B	02
STAX D	12
SHLD adr	22
STA adr	32

byte — constant, or logical/arithmetic expression that evaluates to an 8-bit data quantity (Second byte of 2-byte instructions)

dble — constant, or logical/arithmetic expression that evaluates to a 16-bit data quantity (Second and Third bytes of 3-byte instructions)

adr — 16-bit address (Second and Third bytes of 3-byte instructions)

* all flags (C, Z, S, P, AC) affected

** all flags except CARRY affected (exception: INR and DCR affect no flags)

† only CARRY affected

ARITHMETIC AND LOGICAL GROUP

Add*

ADD	A	87
	B	80
	C	81
	D	82
	E	83
	H	84
	L	85
	M	86
ADC	A	8F
	B	88
	C	89
	D	8A
	E	8B
	H	8C
	L	8D
	M	8E

Subtract*

SUB	A	97
	B	90
	C	91
	D	92
	E	93
	H	94
	L	95
	M	96
SBB	A	9F
	B	98
	C	99
	D	9A
	E	9B
	H	9C
	L	9D
	M	9E

Double Add

DAD	B	09
	D	19
	H	29
	SP	39

Increment**

INR	A	3C
	B	04
	C	0C
	D	14
	E	1C
	H	24
	L	2C
	M	34
INX	B	03
	D	13
	H	23
	SP	33

Decrement**

DCR	A	3D
	B	05
	C	0D
	D	15
	E	1D
	H	25
	L	2D
	M	35
DCX	B	0B
	D	1B
	H	2B
	SP	3B

Special

DAA*	27
CMA*	2F
STC†	37
CMC†	3F

Rotate

RLC	07
RRC	0F
RAL	17
RAR	1F

Logical*

ANA	A	A7
	B	A0
	C	A1
	D	A2
	E	A3
	H	A4
	L	A5
	M	A6
XRA	A	AF
	B	A8
	C	A9
	D	AA
	E	AB
	H	AC
	L	AD
	M	AE
ORA	A	B7
	B	B0
	C	B1
	D	B2
	E	B3
	H	B4
	L	B5
	M	B6
CMP	A	BF
	B	B8
	C	B9
	D	BA
	E	BB
	H	BC
	L	BD
	M	BE

Arith & Logical with Immediate*

ADI byte	C6
ACI byte	CE
SUI byte	D6
SBI byte	DE
ANI byte	E6
XRI byte	EE
ORI byte	F6
CPI byte	FE

BRANCH CONTROL GROUP

Jump

JMP adr	C3
JNZ adr	C2
JZ adr	CA
JNC adr	D2
JC adr	DA
JPO adr	E2
JPE adr	EA
JP adr	F2
JM adr	FA
PCHL	E9

Call

CALL adr	CD
CNZ adr	C4
CZ adr	CC
CNC adr	D4
CC adr	DC
CPO adr	E4
CPE adr	EC
CP adr	F4
CM adr	FC

Return

RET	C9
RNZ	C0
RZ	C8
RNC	D0
RC	D8
RPO	E0
RPE	E8
RP	F0
RM	F8

Restart

RST	0	C7
	1	CF
	2	D7
	3	DF
	4	E7
	5	EF
	6	F7
	7	FF

D7	RST	2
D8	RC	
D9	---	
DA	JC	adr
DB	IN	byte
DC	CC	adr
DD	---	
DE	SBI	byte
DF	RST	3
E0	RPO	
E1	POP	H
E2	JPO	adr
E3	XTHL	
E4	CPO	adr
E5	PUSH	H
E6	ANI	byte
E7	RST	4
E8	RPE	
E9	PCHL	
EA	JPE	adr
EB	XCHG	
EC	CPE	adr
ED	---	
EE	XRI	byte
EF	RST	5
F0	RP	
F1	POP	PSW
F2	JP	adr
F3	DI	
F4	CP	adr
F5	PUSH	PSW
F6	ORI	byte
F7	RST	6
F8	RM	
F9	SPHL	
FA	JM	adr
FB	EI	
FC	CM	adr
FD	---	
FE	CPI	byte
FF	RST	7

HEX-ASCII TABLE

00	NUL		21	!	42	B		63	c		
01	SOH		22	"	43	C		64	d		
02	STX		23	#	44	D		65	e		
03	ETX		24	$	45	E		66	f		
04	EOT		25	%	46	F		67	g		
05	ENQ		26	&	47	G		68	h		
06	ACK		27	'	48	H		69	i		
07	BEL		28	(	49	I		6A	j		
08	BS		29	)	4A	J		6B	k		
09	HT		2A	*	4B	K		6C	l		
0A	LF		2B	+	4C	L		6D	m		
0B	VT		2C	,	4D	M		6E	n		
0C	FF		2D	-	4E	N		6F	o		
0D	CR		2E	.	4F	O		70	p		
0E	SO		2F	/	50	P		71	q		
0F	SI		30	0	51	Q		72	r		
10	DLE		31	1	52	R		73	s		
11	DC1	(X-ON)	32	2	53	S		74	t		
12	DC2	(TAPE)	33	3	54	T		75	u		
13	DC3	(X-OFF)	34	4	55	U		76	v		
14	DC4	(TAPE)	35	5	56	V		77	w		
15	NAK		36	6	57	W		78	x		
16	SYN		37	7	58	X		79	y		
17	ETB		38	8	59	Y		7A	z		
18	CAN		39	9	5A	Z		7B	{		
19	EM		3A	:	5B	[		7C	|		
1A	SUB		3B	;	5C	\		7D	}		
1B	ESC		3C	<	5D	]			(ALT MODE)		
1C	FS		3D	=	5E	^	(↑)	7E	~		
1D	GS		3E	>	5F	—	(←)	7F	DEL		
1E	RS		3F	?	60	`			(RUB OUT)		
1F	US		40	@	61	a					
20	SP		41	A	62	b					

INTEL CORPORATION
3065 Bowers Avenue
Santa Clara, California 95051
Tel: (408) 987-8080

INTEL JAPAN CORPORATION
Flower Hill-Shinmachi East Bldg
1-23-9, Shinmachi, Setagaya-ku
Tokyo 154, Japan
Tel. (03) 426-9261

INTEL INTERNATIONAL
Rue du Moulin a Papier
51-Boite 1
B-1160 Brussels, Belgium
Tel. (02) 660 30 10

Printed in U.S.A./A348/0481/100K BL

intel

8085/8080 Assembly Language Reference Card

March 1979

9800438D

...AND
E CONTROL

...ack Ops

B	C5
D	D5
H	E5
PSW	F5
B	C1
D	D1
H	E1
PSW*	F1

...THL E3
...PHL F9

...Input Output

...IT byte D3
...byte DB

...Control

	F3
	FB
...P	00
...T	76

...ctions
...ly)

...BLER
...ENCE

...tors

...H
...SHL SHR

ASSEMBLER REFERENCE (Cont.)

Pseudo Instruction

General:

ORG
END
EQU
SET
DS
DB
DW

Macros:

MACRO
ENDM
LOCAL
REPT
IRP
IRPC
EXITM

Relocation:

ASEG NAME
DSEG STKLN
CSEG STACK
PUBLIC MEMORY
EXTRN

Conditional Assembly:

IF
ELSE
ENDIF

Constant Delimiter

08DH, 1AH	Hex
105D, 105	Decimal
720, 720	Octal
11011B, 001108	Binary
'TEST', 'A', 'B'	ASCII

INTEL® 8080/8085 INSTRUCTION SET REFERENCE TABLES

INTERNAL REGISTER ORGANIZATION

A Reg. (8)	

B Reg. (8)	C Reg. (8)
D Reg. (8)	E Reg. (8)
H Reg. (8)	L Reg. (8)
Program Counter (16)	
Stack Pointer (16)	

REGISTER PAIR ORGANIZATION

PSW

A (8)	FLAGS (8)

NOTE: Leftmost Byte is high-order byte for arithmetic operations and addressing. Left byte is pushed on stack first. Right byte is popped first.

B	(B/C) (16)
D	(D/E) (16)
H	(H/L) (16)
Prog. Ctr.	(16)
Stack Ptr.	(16)

BRANCH CONTROL INSTRUCTIONS

Flag Condition	Jump		Call		Return	
Zero=True	JZ	CA	CZ	CC	RZ	C8
Zero=False	JNZ	C2	CNZ	C4	RNZ	C0
Carry=True	JC	DA	CC	DC	RC	D8
Carry=False	JNC	D2	CNC	D4	RNC	D0
Sign=Positive	JP	F2	CP	F4	RP	F0
Sign=Negative	JM	FA	CM	FC	RM	F8
Parity=Even	JPE	EA	CPE	EC	RPE	E8
Parity=Odd	JPO	E2	CPO	E4	RPO	E0
Unconditional	JMP	C3	CALL	CD	RET	C9

ACCUMULATOR OPERATIONS

	Code	Function
XRA A	AF	Clear A and Clear Carry
ORA A	B7	Clear Carry
CMC	3F	Complement Carry
CMA	2F	Complement Accumulator
STC	37	Set Carry
RLC	07	Rotate Left
RRC	0F	Rotate Right
RAL	17	Rotate Left Thru Carry
RAR	1F	Rotate Right Thru Carry
DAA	27	Decimal Adjust Accum

REGISTER PAIR AND STACK OPERATIONS

	Register Pair PSW (A/F)	B (B/C)	D (D/E)	H (H/L)	SP	PC	Function
INX		03	13	23	33		Increment Register Pair
DCX		0B	1B	2B	3B		Decrement Register Pair
LDAX		0A	1A	7E(1)			Load A Indirect (Reg. Pair holds Adrs)
STAX		02	12	77(2)			Store A Indirect (Reg. Pair holds Adrs)
LHLD				2A			Load H:L Direct (Bytes 2 and 3 hold Adrs)
SHLD				22			Store H:L Direct (Bytes 2 and 3 hold Adrs)
LXI		01	11	21	31	C3(3)	Load Reg. Pair Immediate (Bytes 2 and 3 hold immediate data)
PCHL						F9	Load PC with H:L (Branch to Adrs in H:L)
XCHG				EB			Exchange Reg. Pairs D, E and H:L
DAD		09	19	29	39		Add Reg. Pair to H:L
PUSH	F5	C5	D5	E5			Push Reg. Pair on Stack
POP	F1	C1	D1	E1			Pop Reg. Pair off Stack
XTHL				E3			Exchange H:L with Top of Stack
SPHL					F9		Load SP with H:L

Notes: 1. This is MOV A,M. 2. This is MOV M,A. 3. This is JMP.