UNIVERSITI SAINS MALAYSIA

Peperiksaan Semester Kedua
Sidang 1991/92

Mac/April 1992

EET 207 - Pemikroproses dan Peralatan Digit

Masa : [3 jam]

ARAHAN KEPADA CALON:

Sila pastikan bahawa kertas peperiksaan ini mengandungi 5 muka surat beserta Lampiran (4 muka surat) bercetak dan LIMA (5) soalan sebelum anda memulakan peperiksaan ini.

Jawab EMPAT (4) soalan.

Agihan markah bagi setiap soalan diberikan di sut sebelah kanan sebagai peratusan daripada markah keseluruhan yang diperuntukkan bagi soalan berkenaan.

Jawab kesemua soalan di dalam Bahasa Malaysia.

...2/-

1. (a) Dengan bantuan gambarajah, huraikan arkitektur dalaman suatu mikropemproses 8 bit yang tipikal.

(40%)

(b) Apakah yang dimaksudkan dengan sampukan bervektor?

(10%)

(c) Dua bait dikhaskan di alamat 2500 H (bait bawah) dan 2501 H (bait atas), untuk menyimpan satu pembilang 16 bit. Satu rutin sampukan yang berada di alamat 34 H, menokokkan pembilang tersebut sebanyak 1, setiap kali ianya dipanggil. Tuliskan rutin sampukan tersebut dan berikan litar logik yang boleh digunakan untuk menyampuk 8085 menerusi masukan INTR dan seterusnya membekalkan vektor yang berkenaan.

(50%)

2. Dengan menggunakan komponen-komponen yang disenaraikan dalam Lampiran I, tunjukkan bagaimana satu sistem mikropemproses asas dapat dihasilkan.

Keterangan yang perlu diberikan ialah

(a) Gambarajah skematik bagi sistem (sambungan terperinci tidak diperlukan.

(25%)

(b) Peta ingatan. (25%)

(c) Huraian kendalian sistem. (25%)

(d) Carta alir bagi aturcara pengawas untuk sistem tersebut.

(25%)

...3/-

3. (a) Tunjukkan bagaimana satu sistem pengumpulan data 8-saluran dan dikawal oleh 8085, dapat direalisasikan. Beri gambarajah blok dan contoh aturcara untuk mengawal proses tersebut.

(50%)

(b) Anda dikehendaki merekabentuk satu sistem kawalan berasaskan mikropemproses untuk mengawas keadaan di dalam pondok khas bagi tanaman-tanaman iklim sejuk. Terangkan dengan mendalam proses rekabentuk dan pelaksanaan sistem kawalan tersebut.

(50%)

4. Pemindahan data antara dua mikropemproses 8085 boleh dilaksanakan melalui 2 penyelak 8-bit 74LS374 seperti ditunjukkan dalam Rajah 1.

(a) Huraikan secara ringkas, bagaimana pemindahan tersebut dilakukan.

(25%)

(b) Tunjukkan bagaimana isyarat-isyarat CK1 & $\overline{OE1}$ dijanakan oleh 8085(1), dan CK2 & $\overline{OE2}$ oleh 8085(2).

(25%)

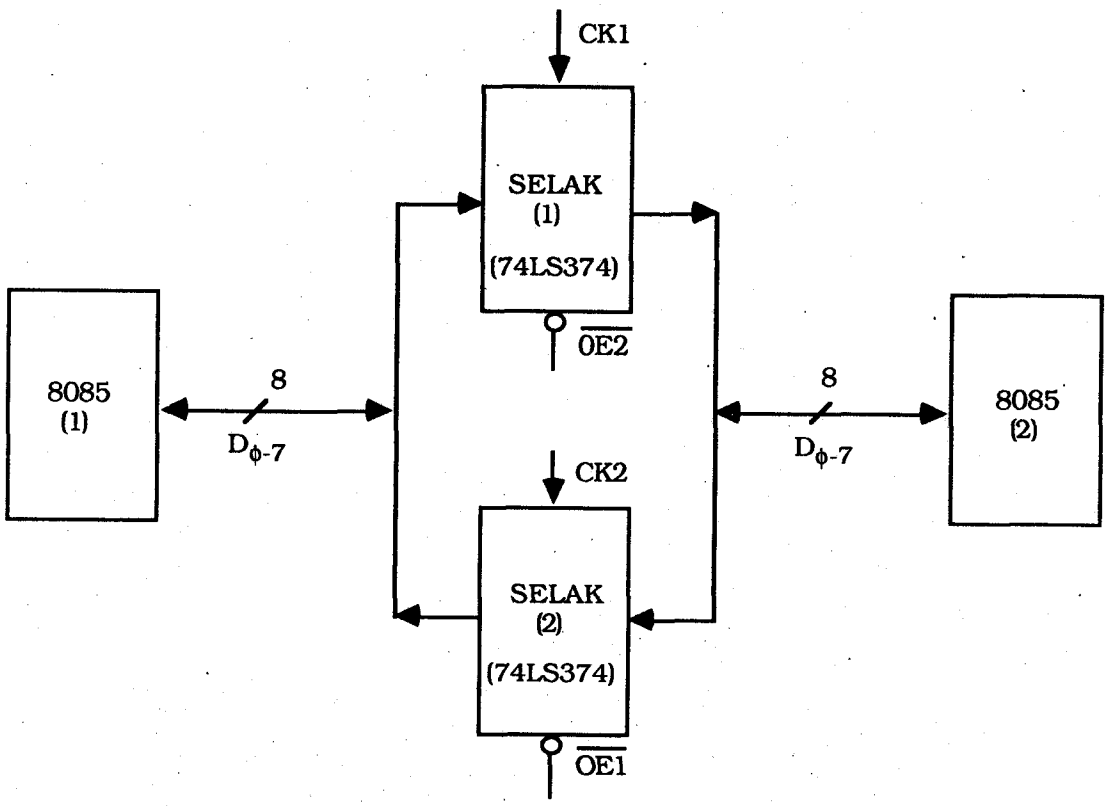
(c) Untuk memastikan pemindahan dilakukan dengan betul, jabat-tangan perlu diadakan. Terangkan cara, serta litar-litar tambahan yang perlu, untuk melaksanakan kaedah ini dalam sistem di atas.

(25%)

...4/-

(d) Tuliskan aturcara bagi kedua-dua mikropemproses untuk melaksanakan operasi pemindahan data dengan berjabat-tangan.

(25%)



$\overline{OE1}$, $\overline{OE2}$ - isyarat pemboleh

CK1, CK2 - denyutan jam (untuk menyelak)

$D_{\phi-7}$ - bas data

Rajah 1

...5/-

5. (a) Rekabentuk satu penjana bentukgelombang sinus yang dikawal oleh mikropemproses 8085. (Keterangan lengkap diperlukan termasuklah spesifikasi, konsep, kendalian litar, dan contoh aturcara).

(60%)

(b) Beri huraian ringkas tentang perkara-perkara berikut:-

(i) perhubungan siri RS232

(ii) teknik DMA

(iii) struktur tindan dalam 8085

(iv) perbezaan antara mikropemproses dengan mikropengawal

(40%)

- oooOooo -

Senarai komponen

Mikropemproses 8085	-	1 unit
Selak 74LS373	-	1 unit
Penyahkod 74LS138	-	1 unit
RAM 6264	-	2 unit
EPROM 2764	-	1 unit
PPI 8255	-	1 unit
Penjana Kadar Baud 4702	-	1 unit
USART 8251	-	1 unit
Penimbal 1488	-	1 unit
Penimbal 1489	-	1 unit
Hablur 6 MHz	-	1 unit
Hablur 2.45 MHz	-	1 unit

8085A

8085A CPU INSTRUCTIONS IN OPERATION CODE SEQUENCE
Table 5-2

OP CODE	MNEMONIC	OP CODE	MNEMONIC	OP CODE	MNEMONIC	OP CODE	MNEMONIC	OP CODE	MNEMONIC	OP CODE	MNEMONIC
00	NOP	2B	DCX H	56	MOV D,M	81	ADD C	AC	XRA H	D7	RST 2
01	LXI B,D16	2C	INR L	57	MOV D,A	82	ADD D	AD	XRA L	D8	RC
02	STAX B	2D	DCR L	58	MOV E,B	83	ADD E	AE	XRA M	D9	–
03	INX B	2E	MVI L,D8	59	MOV E,C	84	ADD H	AF	XRA A	DA	JC Adr
04	INR B	2F	CMA	5A	MOV E,D	85	ADD L	B0	ORA B	DB	IN D8
05	DCR B	30	SIM	5B	MOV E,E	86	ADD M	B1	ORA C	DC	CC Adr
06	MVI B,D8	31	LXI SP,D16	5C	MOV E,H	87	ADD A	B2	ORA D	DD	–
07	RLC	32	STA Adr	5D	MOV E,L	88	ADC B	B3	ORA E	DE	SBI D8
08	–	33	INX SP	5E	MOV E,M	89	ADC C	B4	ORA H	DF	RST 3
09	DAD B	34	INR M	5F	MOV E,A	8A	ADC D	B5	ORA L	E0	RPO
0A	LDAX B	35	DCR M	60	MOV H,B	8B	ADC E	B6	ORA M	E1	POP H
0B	DCX B	36	MVI M,D8	61	MOV H,C	8C	ADC H	B7	ORA A	E2	JPO Adr
0C	INR C	37	STC	62	MOV H,D	8D	ADC L	B8	CMP B	E3	XTHL
0D	DCR C	38	–	63	MOV H,E	8E	ADC M	B9	CMP C	E4	CPO Adr
0E	MVI C,D8	39	DAD SP	64	MOV H,H	8F	ADC A	BA	CMP D	E5	PUSH H
0F	RRC	3A	LDA Adr	65	MOV H,L	90	SUB B	BB	CMP E	E6	ANI D8
10	–	3B	DCX SP	66	MOV H,M	91	SUB C	BC	CMP H	E7	RST 4
11	LXI D,D16	3C	INR A	67	MOV H,A	92	SUB D	BD	CMP L	E8	RPE
12	STAX D	3D	DCR A	68	MOV L,B	93	SUB E	BE	CMP M	E9	PCHL
13	INX D	3E	MVI A,D8	69	MOV L,C	94	SUB H	BF	CMP A	EA	JPE Adr
14	INR D	3F	CMC	6A	MOV L,D	95	SUB L	C0	RNZ	EB	XCHG
15	DCR D	40	MOV B,B	6B	MOV L,E	96	SUB M	C1	POP B	EC	CPE Adr
16	MVI D,D8	41	MOV B,C	6C	MOV L,H	97	SUB A	C2	JNZ Adr	ED	–
17	RAL	42	MOV B,D	6D	MOV L,L	98	SBB B	C3	JMP Adr	EE	XRI D8
18	–	43	MOV B,E	6E	MOV L,M	99	SBB C	C4	CNZ Adr	EF	RST 5
19	DAD D	44	MOV B,H	6F	MOV L,A	9A	SBB D	C5	PUSH B	F0	RP
1A	LDAX D	45	MOV B,L	70	MOV M,B	9B	SBB E	C6	ADI D8	F1	POP PSW
1B	DCX D	46	MOV B,M	71	MOV M,C	9C	SBB H	C7	RST 0	F2	JP Adr
1C	INR E	47	MOV B,A	72	MOV M,D	9D	SBB L	C8	RZ	F3	DI
1D	DCR E	48	MOV C,B	73	MOV M,E	9E	SBB M	C9	RET Adr	F4	CP Adr
1E	MVI E,D8	49	MOV C,C	74	MOV M,H	9F	SBB A	CA	JZ	F5	PUSH PSW
1F	RAR	4A	MOV C,D	75	MOV M,L	A0	ANA B	CB	–	F6	ORI D8
20	RIM	4B	MOV C,E	76	HLT	A1	ANA C	CC	CZ Adr	F7	RST 6
21	LXI H,D16	4C	MOV C,H	77	MOV M,A	A2	ANA D	CD	CALL Adr	F8	RM
22	SHLD Adr	4D	MOV C,L	78	MOV A,B	A3	ANA E	CE	ACI D8	F9	SPHL
23	INX H	4E	MOV C,M	79	MOV A,C	A4	ANA H	CF	RST 1	FA	JM Adr
24	INR H	4F	MOV C,A	7A	MOV A,D	A5	ANA L	D0	RNC	FB	EI
25	DCR H	50	MOV D,B	7B	MOV A,E	A6	ANA M	D1	POP D	FC	CM Adr
26	MVI H,D8	51	MOV D,C	7C	MOV A,H	A7	ANA A	D2	JNC Adr	FD	–
27	DAA	52	MOV D,D	7D	MOV A,L	A8	XRA B	D3	OUT D8	FE	CPI D8
28	–	53	MOV D,E	7E	MOV A,M	A9	XRA C	D4	CNC Adr	FF	RST 7
29	DAD H	54	MOV D,H	7F	MOV A,A	AA	XRA D	D5	PUSH D		
2A	LHLD Adr	55	MOV D,L	80	ADD B	AB	XRA E	D6	SUI D8		

D8 = constant, or logical/arithmetic expression that evaluates to an 8-bit data quantity.

D16 = constant, or logical/arithmetic expression that evaluates to a 16-bit data quantity.

Adr = 16-bit address.

8085A INSTRUCTION SET SUMMARY BY FUNCTIONAL GROUPING
Table 5-3

Mnemonic	Description	Instruction Code (1) D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Page
MOVE, LOAD, AND STORE										
MOV r1 r2	Move register to register	0	1	D	D	D	S	S	S	5-4
MOV M,r	Move register to memory	0	1	1	1	0	S	S	S	5-4
MOV r,M	Move memory to register	0	1	D	D	D	1	1	0	5-4
MVI r	Move immediate register	0	0	D	D	D	1	1	0	5-4
MVI M	Move immediate memory	0	0	1	1	0	1	1	0	5-4
LXI B	Load immediate register Pair B & C	0	0	0	0	0	0	0	1	5-5
LXI D	Load immediate register Pair D & E	0	0	0	1	0	0	0	1	5-5
LXI H	Load immediate register Pair H & L	0	0	1	0	0	0	0	1	5-5
STAX B	Store A indirect	0	0	0	0	0	0	1	0	5-6
STAX D	Store A indirect	0	0	0	1	0	0	1	0	5-6
LDAX B	Load A indirect	0	0	0	0	1	0	1	0	5-5
LDAX D	Load A indirect	0	0	0	1	1	0	1	0	5-5
STA	Store A direct	0	0	1	1	0	0	1	0	5-5
LDA	Load A direct	0	0	1	1	1	0	1	0	5-5
SHLD	Store H & L direct	0	0	1	0	0	0	1	0	5-5
LHLD	Load H & L direct	0	0	1	0	1	0	1	0	5-5
XCHG	Exchange D & E, H & L Registers	1	1	1	0	1	0	1	1	5-6
STACK OPS										
PUSH B	Push register Pair B & C on stack	1	1	0	0	0	1	0	1	5-15
PUSH D	Push register Pair D & E on stack	1	1	0	1	0	1	0	1	5-15
PUSH H	Push register Pair H & L on stack	1	1	1	0	0	1	0	1	5-15
PUSH PSW	Push A and Flags on stack	1	1	1	1	0	1	0	1	5-15
POP B	Pop register Pair B & C off stack	1	1	0	0	0	0	0	1	5-15
POP D	Pop register Pair D & E off stack	1	1	0	1	0	0	0	1	5-15
POP H	Pop register Pair H & L off stack	1	1	1	0	0	0	0	1	5-15
POP PSW	Pop A and Flags off stack	1	1	1	1	0	0	0	1	5-15
XTHL	Exchange top of stack, H & L	1	1	1	0	0	0	1	1	5-16
SPHL	H & L to stack pointer	1	1	1	1	1	0	0	1	5-16
LXI SP	Load immediate stack pointer	0	0	1	1	0	0	0	1	5-5
INX SP	Increment stack pointer	0	0	1	1	0	0	1	1	5-9
DCX SP	Decrement stack pointer	0	0	1	1	1	0	1	1	5-9
JUMP										
JMP	Jump unconditional	1	1	0	0	0	0	1	1	5-13
JC	Jump on carry	1	1	0	1	1	0	1	0	5-13
JNC	Jump on no carry	1	1	0	1	0	0	1	0	5-13
JZ	Jump on zero	1	1	0	0	1	0	1	0	5-13
JNZ	Jump on no zero	1	1	0	0	0	0	1	0	5-13
JP	Jump on positive	1	1	1	1	0	0	1	0	5-13
JM	Jump on minus	1	1	1	1	1	0	1	0	5-13
JPE	Jump on parity even	1	1	1	0	1	0	1	0	5-13
JPO	Jump on parity odd	1	1	1	0	0	0	1	0	5-13
PCHL	H & L to program counter	1	1	1	0	1	0	0	1	5-15
CALL										
CALL	Call unconditional	1	1	0	0	1	1	0	1	5-13
CC	Call on carry	1	1	0	1	1	1	0	0	5-14
CNC	Call on no carry	1	1	0	1	0	1	0	0	5-14
CZ	Call on zero	1	1	0	0	1	1	0	0	5-14
CNZ	Call on no zero	1	1	0	0	0	1	0	0	5-14
CP	Call on positive	1	1	1	1	0	1	0	0	5-14
CM	Call on minus	1	1	1	1	1	1	0	0	5-14
CPE	Call on parity even	1	1	1	0	1	1	0	0	5-14
CPO	Call on parity odd	1	1	1	0	0	1	0	0	5-14
RETURN										
RET	Return	1	1	0	0	1	0	0	1	5-14
RC	Return on carry	1	1	0	1	1	0	0	0	5-14
RNC	Return on no carry	1	1	0	1	0	0	0	0	5-14
RZ	Return on zero	1	1	0	0	1	0	0	0	5-14
RNZ	Return on no zero	1	1	0	0	0	0	0	0	5-14
RP	Return on positive	1	1	1	1	0	0	0	0	5-14
RM	Return on minus	1	1	1	1	1	0	0	0	5-14
RPE	Return on parity even	1	1	1	0	1	0	0	0	5-14
RPO	Return on parity odd	1	1	1	0	0	0	0	0	5-14
RESTART										
RST	Restart	1	1	A	A	A	1	1	1	5-14
INPUT/OUTPUT										
IN	Input	1	1	0	1	1	0	1	1	5-16
OUT	Output	1	1	0	1	0	0	1	1	5-16
INCREMENT AND DECREMENT										
INR r	Increment register	0	0	D	D	D	1	0	0	5-8
DCR r	Decrement register	0	0	D	D	D	1	0	1	5-8
INR M	Increment memory	0	0	1	1	0	1	0	0	5-8
DCR M	Decrement memory	0	0	1	1	0	1	0	1	5-8
INX B	Increment B & C registers	0	0	0	0	0	0	1	1	5-9
INX D	Increment D & E registers	0	0	0	1	0	0	1	1	5-9
INX H	Increment H & L registers	0	0	1	0	0	0	1	1	5-9
DCX B	Decrement B & C	0	0	0	0	1	0	1	1	5-9
DCX D	Decrement D & E	0	0	0	1	1	0	1	1	5-9
DCX H	Decrement H & L	0	0	1	0	1	0	1	1	5-9
ADD										
ADD r	Add register to A	1	0	0	0	0	S	S	S	5-6
ADC r	Add register to A with carry	1	0	0	0	1	S	S	S	5-6
ADD M	Add memory to A	1	0	0	0	0	1	1	0	5-6
ADC M	Add memory to A with carry	1	0	0	0	1	1	1	0	5-7
ADI	Add immediate to A	1	1	0	0	0	1	1	0	5-6
ACI	Add immediate to A with carry	1	1	0	0	1	1	1	0	5-7
DAD B	Add B & C to H & L	0	0	0	0	1	0	0	1	5-9
DAD D	Add D & E to H & L	0	0	0	1	1	0	0	1	5-9
DAD H	Add H & L to H & L	0	0	1	0	1	0	0	1	5-9
DAD SP	Add stack pointer to H & L	0	0	1	1	1	0	0	1	5-9
SUBTRACT										
SUB r	Subtract register from A	1	0	0	1	0	S	S	S	5-7
SBB r	Subtract register from A with borrow	1	0	0	1	1	S	S	S	5-7
SUB M	Subtract memory from A	1	0	0	1	0	1	1	0	5-7
SBB M	Subtract memory from A with borrow	1	0	0	1	1	1	1	0	5-8
SUI	Subtract immediate from A	1	1	0	1	0	1	1	0	5-7

8085A INSTRUCTION SET SUMMARY (Cont'd)
Table 5-3

Mnemonic	Description	Instruction Code (1) D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Page
SBI	Subtract immediate from A with borrow	1	1	0	1	1	1	1	0	5-8
LOGICAL										
ANA r	And register with A	1	0	1	0	0	S	S	S	5-9
XRA r	Exclusive OR register with A	1	0	1	0	1	S	S	S	5-10
ORA r	OR register with A	1	0	1	1	0	S	S	S	5-10
CMP r	Compare register with A	1	0	1	1	1	S	S	S	5-11
ANA M	And memory with A	1	0	1	0	0	1	1	0	5-10
XRA M	Exclusive OR memory with A	1	0	1	0	1	1	1	0	5-10
ORA M	OR memory with A	1	0	1	1	0	1	1	0	5-11
CMP M	Compare memory with A	1	0	1	1	1	1	1	0	5-11
ANI	And immediate with A	1	1	1	0	0	1	1	0	5-10
XRI	Exclusive OR immediate with A	1	1	1	0	1	1	1	0	5-10
ORI	OR immediate with A	1	1	1	1	0	1	1	0	5-11
CPI	Compare immediate with A	1	1	1	1	1	1	1	0	5-11
ROTATE										
RLC	Rotate A left	0	0	0	0	0	1	1	1	5-11
RRC	Rotate A right	0	0	0	0	1	1	1	1	5-12
RAL	Rotate A left through carry	0	0	0	1	0	1	1	1	5-12
RAR	Rotate A right through carry	0	0	0	1	1	1	1	1	5-12
SPECIALS										
CMA	Complement A	0	0	1	0	1	1	1	1	5-12
STC	Set carry	0	0	1	1	0	1	1	1	5-12
CMC	Complement carry	0	0	1	1	1	1	1	1	5-12
DAA	Decimal adjust A	0	0	1	0	0	1	1	1	5-9
CONTROL										
EI	Enable Interrupts	1	1	1	1	1	0	1	1	5-17
DI	Disable Interrupt	1	1	1	1	0	0	1	1	5-17
NOP	No-operation	0	0	0	0	0	0	0	0	5-17
HLT	Halt	0	1	1	1	0	1	1	0	5-17
NEW 8085A INSTRUCTIONS										
RIM	Read Interrupt Mask	0	0	1	0	0	0	0	0	5-17
SIM	Set Interrupt Mask	0	0	1	1	0	0	0	0	5-18

NOTES: 1. DDS or SSS: B 000, C 001, D 010, E011, H 100, L 101, Memory 110, A 111.
2. Two possible cycle times. (6/12) indicate instruction cycles dependent on condition flags.