

UNIVERSITI SAINS MALAYSIA
Peperiksaan Semester Pertama
Sidang 1987/88

EET 307 Sistem Pemikroproses dan
Peralatan Digit

Tarikh: 5 November 1987

Masa: 9.00 pagi - 12.00 tengahari
(3 Jam)

ARAHAN KEPADA CALON:

Sila pastikan bahawa kertas peperiksaan ini mengandungi 4 muka surat berserta lampiran (2 muka surat) bercetak sebelum anda memulakan peperiksaan ini.

Jawab LIMA(5) soalan.

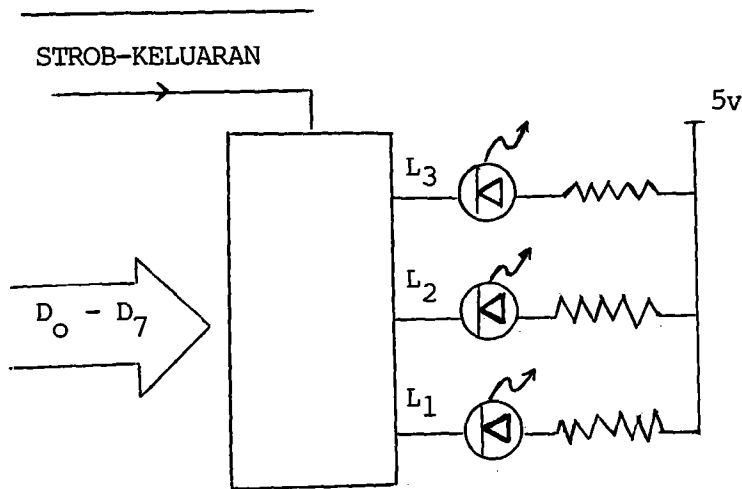
Agihan markah bagi setiap soalan diberikan di sut sebelah kanan sebagai peratusan daripada markah keseluruhan yang diperuntukkan bagi soalan berkenaan.

Jawab kesemua soalan di dalam Bahasa Malaysia.

...2/-

1. Rajah 1 menunjukkan satu liang keluaran sistem 8085 yang dipandu oleh hablur 2 MHz. Lampu L3 hendak dikelipkan dengan kadar 20 Hz (Nisbah tanda ke ruang adalah sama). Keadaan lampu-lampu lain mestilah tidak berubah. Tulis satu aturcara bahasa penghimpun 8085A untuk melaksanakan ini.

(80%)



Rajah 1

Jika satu masa tunggu dimasukkan kedalam setiap kitaran mesin apakah akibat ke atas pemasaan.

(20%)

2. Perkakasan penukar A/D boleh dihasilkan dari satu liang keluaran, satu liang masukan, penukar D/A dan pembandingan. Tunjukkan rajah blok untuk sistem ini.

(50%)

Tuliskan aturcara bahasa penghimpun 8085A untuk membuat penukaran A/D dengan mengguna kaedah penghampiran berturutan.

(50%)

...3/-

3. Penumpuk asalnya mengandungi nilai 34 H. Berikan kandungan penumpuk selepas setiap kendalian berikut dilakukan.

- (a) anjakan logik dua bit ke kiri.
- (b) anjakan aritmetik dua bit ke kiri.
- (c) anjakan logik dua bit ke kanan.
- (d) anjakan aritmetik dua bit ke kanan

(20%)

Bincangkan perwakilan nombor titik apung.

(30%)

Berikan satu sublazim untuk menukar nombor perduaan ke nombor perpuluhan.

(50%)

4. Arahan-arahan pemikroproses 8085A boleh dikumpulkan dalam lima kumpulan. Nyatakan kumpulan-kumpulan ini dan terangkan fungsi untuk setiap kumpulan.

(50%)

Satu jadual lompat diguna untuk menyimpan nilai-nilai alamat masukan kepada enambelas lazim. Nilai di antara 0 hingga 15 dalam penumpuk memilih lazim yang hendak dilaksanakan. Tulis satu aturcara dalam bahasa penghimpun 8085A untuk melakukan ini. Satu keadaan ralat akan dikembalikan dengan memasang bendera pembawa sekiranya nilai dalam penumpuk < 0 atau > 15 .

(50%)

...4/-

5. Dua byte ingatan dikhaskan untuk menyimpan nilai 16 bit. Nilai ini diguna untuk membilang berapa kali suatu peristiwa luaran yang berlaku. BIL dan BIL + 1 adalah alamat yang digunakan untuk nilai 16 bit ini. Suatu lazim khidmat sampukan di alamat 38 H akan meningkatkan kandungan BIL setiap kali ianya dipanggil.

Tulis lazim khidmat sampukan ini.

(40%)

Hasilkan perkakasan luaran untuk menyampuk 8085A dan memberi nilai vektor 38 H secara terus. Cuma guna masukan sampukan INTR.

(60%)

6. Nyatakan perbezaan-perbezaan di antara bahasa tinggi dan bahasa penghimpun.

(30%)

Untuk membangun perisian pemikroproses, beberapa kemudahan perkakasan dan perisian diperlukan. Terangkan kemudahan-kemudahan ini.

(40%)

Huraikan sintaks penghimpun ASM85.

(30%)

-ooo0ooo-

LAMPIRAN A

INSTRUCTION TIMINGS

Mnemonic	Description	Clock(2) Cycles	Mnemonic	Description	Clock(2) Cycles	Mnemonic	Description	Clock(2) Cycles
MOVE, LOAD, AND STORE								
MOV r, r2	Move register to register	4	CNC	Call on no carry	9/18	SBB r	Subtract register from A with borrow	4
MOV M, r	Move register to memory	7	CZ	Call on zero	9/18	SUB M	Subtract memory from A	7
MOV r, M	Move memory to register	7	CNZ	Call on no zero	9/18	SBB M	Subtract memory from A with borrow	7
MVI r	Move immediate register	7	CP	Call on positive	9/18	SUI	Subtract immediate from A	7
MVI M	Move immediate memory	10	CM	Call on minus	9/18	SBI	Subtract immediate from A with borrow	7
LXI B	Load immediate register Pair B & C	10	CPE	Call on parity even	9/18	LOGICAL		
LXI D	Load immediate register Pair D & E	10	CPO	Call on parity odd	9/18	ANA r	And register with A	4
LXI H	Load immediate register Pair H & L	10	RETURN			XRA r	Exclusive OR register with A	4
LXI SP	Load immediate stack pointer	10	RET	Return	10	ORA r	OR register with A	4
STAX B	Store A indirect	7	RC	Return on carry	6/12	CMP r	Compare register with A	4
STAX D	Store A indirect	7	RNC	Return on no carry	6/12	ANA M	And memory with A	7
LDAX B	Load A indirect	7	RZ	Return on zero	6/12	XRA M	Exclusive OR memory with A	7
LDAX D	Load A indirect	7	RNZ	Return on no zero	6/12	ORA M	OR memory with A	7
STA	Store A direct	13	RP	Return on positive	6/12	CMP M	Compare memory with A	7
LDA	Load A direct	13	RM	Return on minus	6/12	ANI	And immediate with A	7
SHLD	Store H & L direct	16	RPE	Return on parity even	6/12	XRI	Exclusive OR immediate with A	7
LHLD	Load H & L direct	16	RPO	Return on parity odd	6/12	ORI	OR immediate with A	7
XCHG	Exchange D & E, H & L Registers	4	RESTART			CPI	Compare immediate with A	7
STACK OPS								
PUSH B	Push register Pair B & C on stack	12	RST	Restart	12	ROTATE		
PUSH D	Push register Pair D & E on stack	12	INPUT/OUTPUT			RLC	Rotate A left	4
PUSH H	Push register Pair H & L on stack	12	IN	Input	10	RRC	Rotate A right	4
PUSH PSW	Push A and Flags on stack	12	OUT	Output	10	RAL	Rotate A left through carry	4
POP B	Pop register Pair B & C off stack	10	INCREMENT AND DECREMENT			RAR	Rotate A right through carry	4
POP D	Pop register Pair D & E off stack	10	INR r	Increment register	4	SPECIALS		
POP H	Pop register Pair H & L off stack	10	DCR r	Decrement register	4	CMA	Complement A	4
POP PSW	Pop A and Flags off stack	10	INR M	Increment memory	10	STC	Set carry	4
XTHL	Exchange top of stack, H & L	16	DCR M	Decrement memory	10	CNC	Complement carry	4
SPHL	H & L to stack pointer	6	INX B	Increment B & C registers	6	DAA	Decimal adjust A	4
JUMP								
JMP	Jump unconditional	10	INX D	Increment D & E registers	6	CONTROL		
JC	Jump on carry	7/10	INX H	Increment H & L registers	6	EI	Enable interrupt	4
JNC	Jump on no carry	7/10	INX SP	Increment stack pointer	6	DI	Disable interrupt	4
JZ	Jump on zero	7/10	DCX B	Decrement B & C	6	NOP	No-operation	4
JNZ	Jump on no zero	7/10	DCX D	Decrement D & E	6	HLT	Halt	5
JP	Jump on positive	7/10	DCX H	Decrement H & L	6	NEW 8085A INSTRUCTIONS		
JM	Jump on minus	7/10	DCX SP	Decrement stack pointer	6	RIM	Read Interrupt Mask	4
JPE	Jump on parity even	7/10	ADD			SIM	Set Interrupt Mask	4
JPO	Jump on parity odd	7/10	ADD r	Add register to A	4			
PCHL	H & L to program counter	6	AOC r	Add register to A with carry	4			
CALL								
CALL	Call unconditional	18	ADD M	Add memory to A	7			
CC	Call on carry	9/18	AOC M	Add memory to A with carry	7			
SUBTRACT								
			SUB r	Subtract register from A	4			

NOTES: 1. D05 or S55: B 000, C 001, D 010, E 011, H 100, L 101, Memory 110, A 111.
 2. Two possible cycle times. (6/12) indicate instruction cycles dependent on condition flags.

*All mnemonics copyright © Intel Corporation 1977

LAMPIRAN B

HEXADECIMAL CODING CHART

DATA TRANSFER GROUP

Move		Move (cont)		Move Immediate	
MOV	A.A 7F	MOV	E.A 5F	MVI	A. byte 3E
	A.B 78		E.B 58		B. byte 0E
	A.C 79		E.C 59		C. byte 3E
	A.D 7A		E.D 5A		D. byte 1E
	A.E 7B		E.E 5B		E. byte 1E
	A.H 7C		E.H 5C		H. byte 2E
A.L 7D	E.L 5D	L. byte 2E			
A.M 7E	E.M 5E	M. byte 3E			
MOV	B.A 47	MOV	H.A 67	LXI	B. dble 01
	B.B 40		H.B 60		D. dble 11
	B.C 41		H.C 61		H. dble 21
	B.D 42		H.D 62		SP. dble 31
	B.E 43		H.E 63		
	B.H 44		H.H 64		
B.L 45	H.L 65				
B.M 46	H.M 66				
MOV	C.A 4F	MOV	L.A 8F	Load/Store	
	C.B 48		L.B 88	LDAX B 0A	
	C.C 49		L.C 89	LDAX D 1A	
	C.D 4A		L.D 8A	LHLD adr 2A	
	C.E 4B		L.E 8B	LDA adr 3A	
	C.H 4C		L.H 8C	STAX B 02	
C.L 4D	L.L 8D	STAX D 12			
C.M 4E	L.M 8E	SHLD adr 22			
MOV	D.A 57	MOV	M.A 77	STA adr 32	
	D.B 50		M.B 70		
	D.C 51		M.C 71		
	D.D 52		M.D 72		
	D.E 53		M.E 73		
	D.H 54		M.H 74		
D.L 55	M.L 75				
D.M 56	M.L 75				

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).

ARITHMETIC AND LOGICAL GROUP

Add		Increment		Logical	
ADD	A 87	INR	A 3C	ANA	A A7
	B 80		B 04		B A0
	C 81		C 0C		C A1
	D 82		D 14		D A2
	E 83		E 1C		E A3
	H 84		H 24		H A4
L 85	L 2C	L A5			
M 86	M 34	M A6			
ADC	A' 9F	INX	B 03	XRA	A AF
	B 88		D 13		B A8
	C 89		H 23		C A9
	D 8A		SP 33		D AA
	E 8B				E AB
	H 8C				H AC
L 8D		L AD			
M 8E		M AE			
SUB	A 97	Decrement		ORA	A B7
	B 90	A 3D	B 05		B B0
	C 91	B C 0D	C 00		C B1
	D 92	D 15	D 1D		D B2
	E 93	H 25	L 2D		E B3
	H 94	L M 35	M 35		H B4
L 95	OCX	B 08	L B5	L B5	
M 96		D 18	M B6		
SBB		A 9F	H 28	A B7	
		B 98	SP 38	B B8	
		C 99		C B9	
		D 9A	Specials		D BA
	E 9B	DAA 27	E BB		
	H 9C	CMA 2F	H BC		
L 9D	STC 37	L BD			
M 9E	CMC 3F	M BE			
DAD	B 09	Rotate		CMP	A B7
	D 19	RLC 07	B B8		
	H 29	RRC 0F	C B9		
	SP 39	RAL 17	D BA		
		RAR 1F	E BB		
			H BC		

Arith & Logical Immediate
 ADI byte C6
 ACI byte CE
 SUI byte D8
 SBI byte DE
 ANI byte E8
 XRI byte EE
 ORI byte F6
 CPI byte FE

BRANCH CONTROL GROUP

Jump		Return	
JMP adr C3	RET C9		
JNZ adr C2	RNZ C0		
JZ adr CA	RZ C8		
JNC adr D2	RNC D0		
JC adr DA	RC D8		
JPO adr E2	RPO E0		
JPE adr EA	RPE E8		
JP adr F2	RP F0		
JM adr FA	RM F8		
PCHL E9			
Call			
CALL adr C0			
CNZ adr C4			
CZ adr CC			
CNC adr D4			
CC adr DC			
CPO adr E4			
CPE adr EC			
CP adr F4			
CM adr FC			
Restart			
RST 0 C7			
RST 1 CF			
RST 2 D7			
RST 3 DF			
RST 4 E7			
RST 5 EF			
RST 6 F7			
RST 7 FF			

I/O AND MACHINE CONTROL

Stack Ops		Control	
PUSH	B C5	DI F3	
	D D5	EI FB	
	H E5	NOP 00	
	PSW F5	HLT 76	
POP	B C1	New Instructions (8085 Only)	
	D D1	RIM 20	
	H E1	SIM 30	
	PSW F1		
XTHL E3			
SPHL F9			
Input-Output			
OUT byte D3			
IN byte D3			