TECHNICAL UNIVERSITY OF MOMBASA

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

UNIVERSITY EXAMINATION FOR:

BACHELOR OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING

EEE 2430: MICROPROCESSOR.

SPECIAL/SUPPLEMENTARY EXAMINATION

SERIES: SEPTEMBER 2018

TIME: 2 HOURS

DATE: SEPTEMBER 2018

Instructions to Candidates

You should have the following for this examination

-Answer Booklet, examination pass and student ID

This paper consists of FIVE questions. Question ONE is Compulsory attempt any other TWO questions.

Do not write on the question paper.

Question ONE

(a) (i.) Perform the following number systems conversions

I. 34.25_{10} to binary

II. $79B_{16}$ to Octal

(ii) Convert the decimal number $E9_{16}$ into

i. BCD Code

ii. Excess-3 code

iii. Gray Code

(ii) Perform the following binary subtraction operation.

I. $14_{10} - 16_{10}$ Using 2's complement method

II. $29_{10} - 25_{10}$ Using 1's complement method

12 marks

b.) With the aid a symbol and truth table, explain the operation of the FOUR types of positive edge triggered flip-flops. 6 marks

c.) i) Explain the function of any THREE dedicated registers in microprocessor.

ii) If the content of the accumulator and register B are C4H and 3A H respectively determine the content of the accumulator after execution of each individual instruction

I. ADD B
II. ORA B
III. SUB B
IV. MOV A, B
V. INR A
VI. CMA

12 marks

Question TWO

a) Draw the block diagram of a microprocessor controlled system and describe the function of:

i. main components
ii. bus highway system 10 marks

b) Write both assembly language and machine code for a program segment beginning at EF20 H to perform the following tasks.

- Move the data 1FH into register B
- Copy the content of the accumulator to register E
- Move data value 2297H into HL register pair
- Subtract the content of register B from accumulator
- Add the content of register E to the accumulator
- Increment the content of register H
- End the program 10 marks

Question THREE

a) Describe any THREE addressing modes stating a typical instruction in each case as applied to Intel 8085 microprocessor 9 marks

b) Explain the events that take place when the following instruction are executed

i. DCX D
ii. XRA B
iii. LDAX D 6 marks

c) Write an assembly language program starting at address 3000H to add TWO numbers, 3AH and 72H that are initially loaded in the accumulator and register C respectively, the result to be stored at a memory location 2D07H to end the program

5 marks

Question FOUR

a.) i. distinguish between asynchronous and synchronous counter

ii. Draw a 4 bit SIPO shift register and explain how a binary word 1011_2 is stored
10 marks

b.) i. Explain the function of any THREE flags in the status register

ii. State any TWO functional categories of Intel 8085 microprocessor instructions giving TWO examples for each 10 marks

Question FIVE

a. Develop a truth table for a two input XOR logic gate and hence implement its combinational logic Circuit. 5 marks

b. (i) Distinguish between the following terms

I. Combinational and Sequential logic circuits

II. Negative and positive edge triggered flip flops

(ii) Using K-map optimization approach Implement a logic circuit with THREE input variables that will produce a HIGH output only when any TWO variables are HIGH
15 marks

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	DB
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.