Summer 2016 EXAMINATIONS

Subject Code: **17443** **Model Answer**

Important Instructions to examiners:

1) The answers should be examined by key words and not as word-to-word as given in the answer scheme.

2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.

3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills.

4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.

5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.

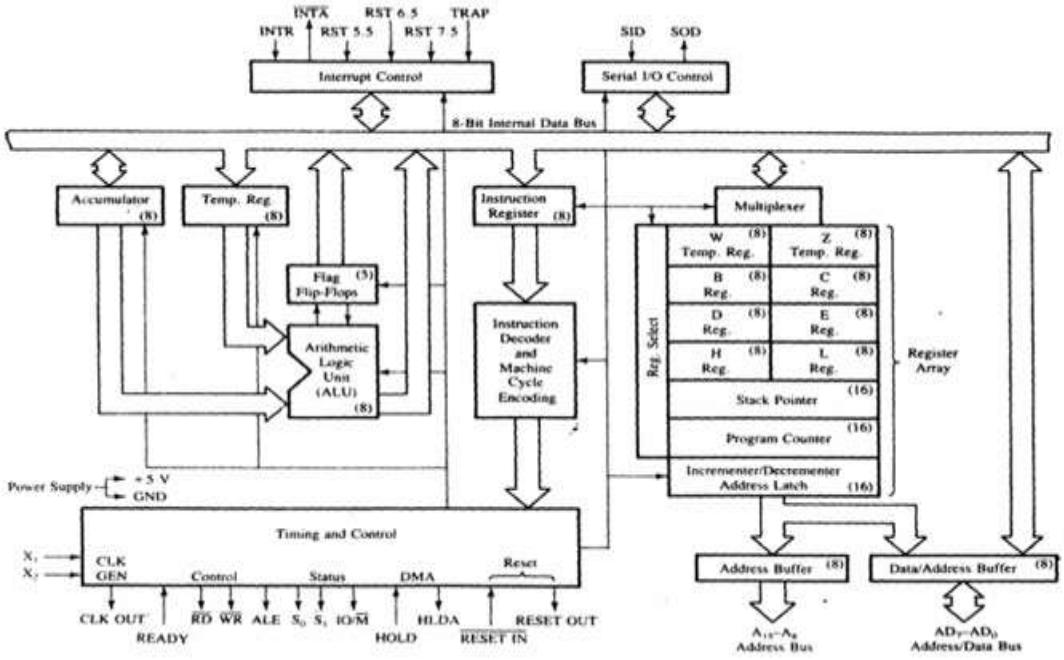
6) In case of some questions credit may be given by judgment on part of examiner of relevant answer based on candidate's understanding.

7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q1. Attempt any Five: 20

(a) Draw the architecture of 8085 microprocessor.

Ans: (4M- diagram)



(b) Write assembly language program to obtain two's complement of 8-bit number.

Ans: Any other correct program should be given marks (3M- program , 1M-comments)

```
Prog:   MOV A, 05H          ; Store a number in accumulator
        CPL  A              ; Take complement of that number
        INR A               ; Add 1 to get two's complement
        STA 2000H           ; Store at some location
```

(c) Write assembly language program to receive 8-bit serially on SID line. Store the byte at 1000 Memory location.

Ans: (4M-Program)

Program to receive 8- bits of data using SID pin

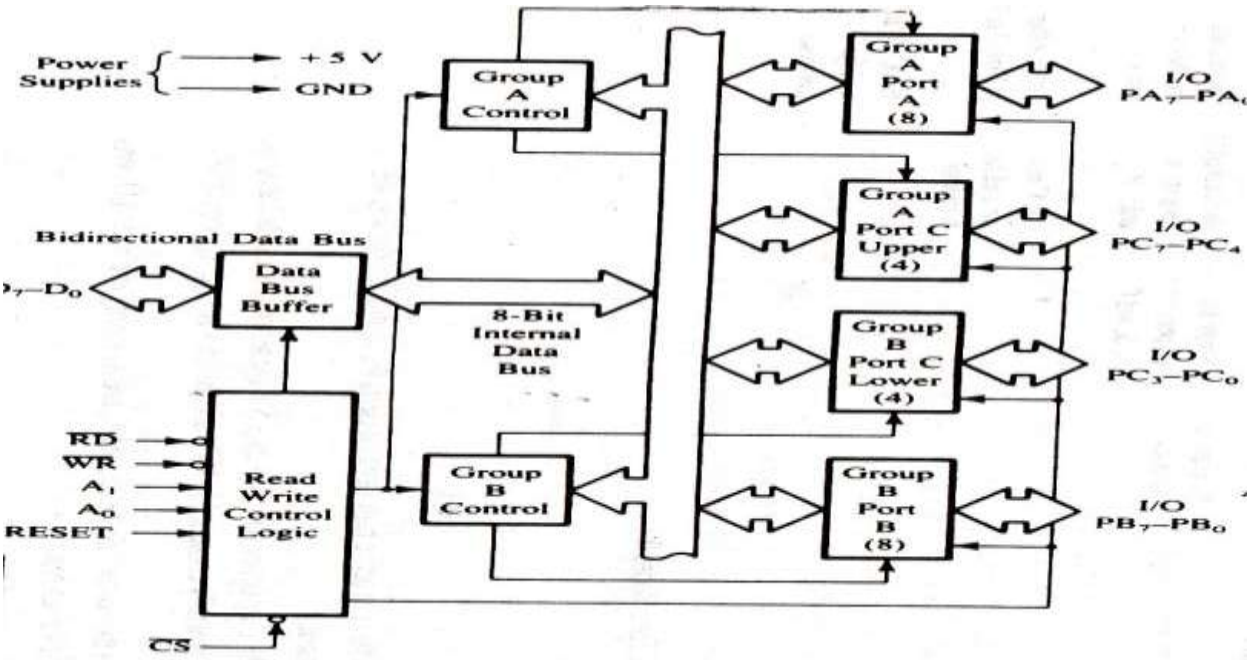
Program :-

```
        LXI   SP, FFFFH
        MVI B, 08H
        MVI D, 00H
UP :    RIM
        ANI   80H
        JNZ  UP
    CALL Delay 1
RIM
ANI 80H
JNZ UP
 UP 1 :   CALL Delay
RIM
```

```
ANI  80H
ORA  D
RRC
MOV D, A
DCR B
JNZ  UP 1
RLC
STA  1000H
ORA   A
JPO       Error
RIM
ANI 80H
CZ  Error
CALL Delay
RIM
ANI 80H
CZ    Error
CALL Delay
          HLT
```

(d) Draw the block diagram of 8255.

Ans: **(4M- diagram)**



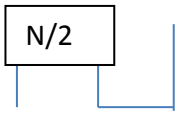
(e) With the help of suitable diagram explain the timer modes of 8155.

Ans: (1/2M- each waveform, 1/2M- each mode explanation)

8155 timer has four modes
1) Mode 0 – Single square wave cycle
2) Mode 1 – Continuous square wave
3) Mode 2- single pulse on terminal count

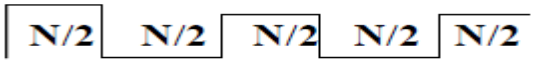
4) Mode 3 – Continuous pulse at the end of terminal count

Mode 0 M_2 M_1 = 0 N = Count
In this mode the timer output remains high for half the count and row for remaining half of the count.



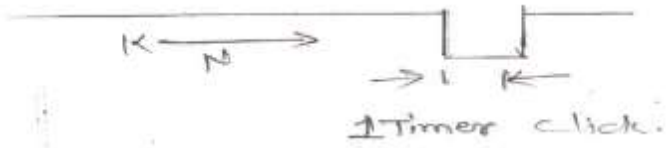
Mode 1 M_2 =0 M_1=0

In this mode timer O/P remains high for half and Low for half the period are repeated to generate the continuous square wave. The count is automatic reloaded.

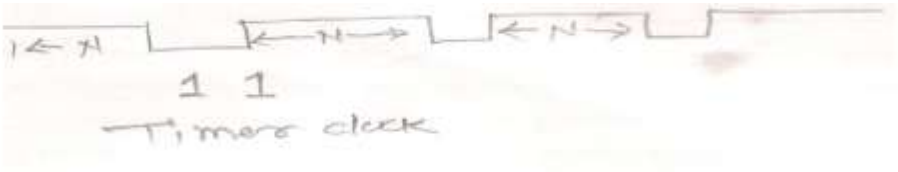


Mode 2
M_2= 1 M_1 = 0

This mode generates single clock pulse with variable on time. The pulse width is a function of clock frequency.



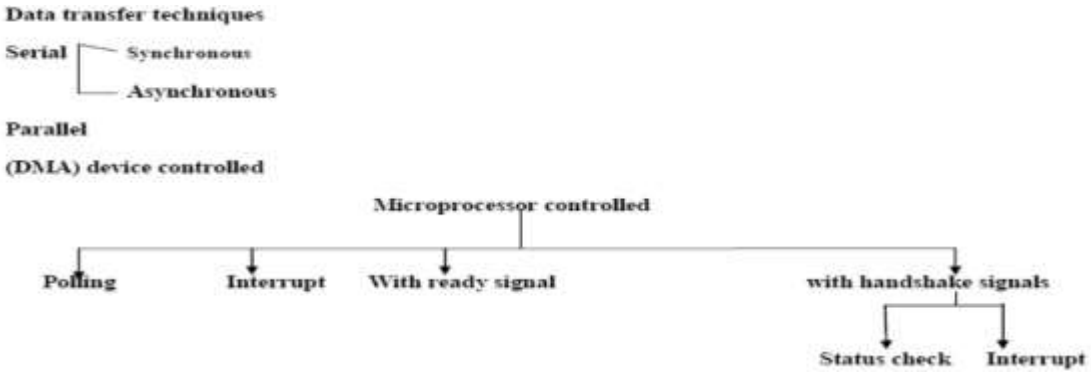
Mode 3 M_2 = 1 M_1 = 1
This mode generates pulse of the end of terminal count & repeated as shown in timing diagram

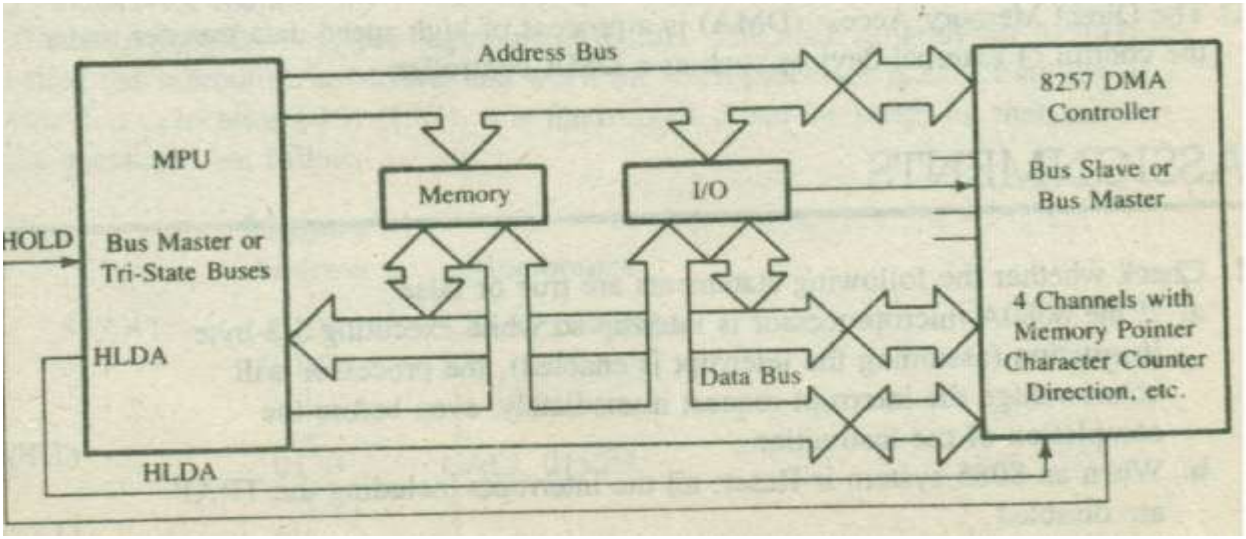


(f) List the different data transfer techniques and explain DMA controlled data transfer technique.

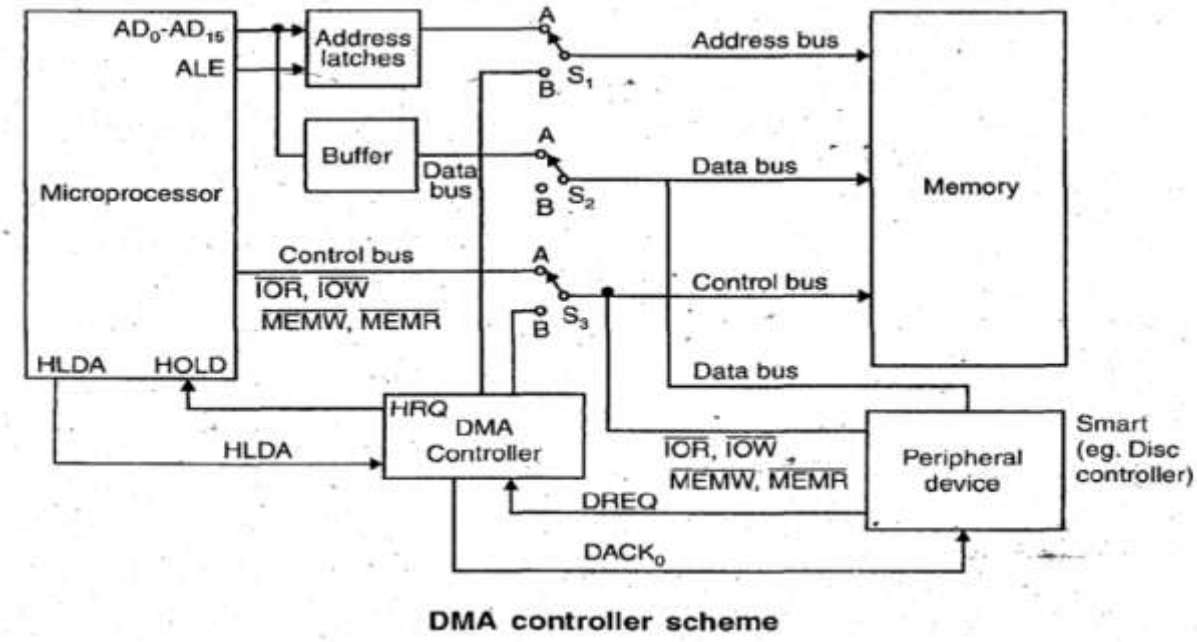
Ans: (1M- list,1M-diagram,2M-explanation)

Different data transfer techniques are:





OR



DMA controller scheme

In situation in which the microprocessor controlled data transfer is too slow, the DMA is generally used. E.g. data transfer between a floppy disk & R/W memory of the system. In this data transfer method, the data transfer operation is carried out by the DMA controller which is another master in the microprocessor based system. The data is transferred directly between I/O device and memory and data transfer is controlled by either I/O device or DMA controller. Microprocessor does not participate in this data transfer method. Whenever there is request from the I/O device, then DMA controller takes the control of all system buses i.e. address bus, data bus and control bus and perform data transfer operation directly between I/O device and memory. This method is used when the large amount of data is required to be transfer. IN IBM PC, Hard disk drive, floppy disk drive CD- ROM etc. devices uses this method of data transfer. In this method , when an I/O device wishes for data transfer, an I/O device itself generate request signal DREQ to DMA controller. In response to DREQ, the DMA controller send HOLD signal to the microprocessor. After receiving HOLD signal, microprocessor performs current operation completely and transfer the control of all system buses i.e. address bus, data bus and control bus to DMA controller. The DMA controller start data transfer operation. The speed of the data transfer is faster as compare to programmed I/O data transfer method. The three data transfer schemes of DMA are as given below:

i. Single Byte Transfer

ii. Block Transfer
iii. Hidden or Transparent DMA

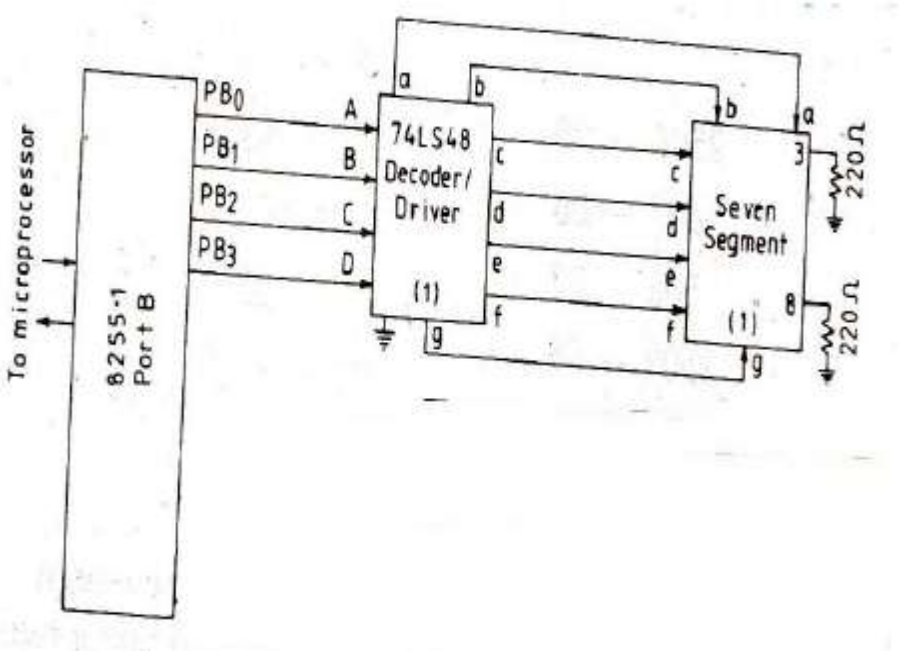
i. Single Byte Transfer:
Only one byte data is transferred at a time – data transfer speed is low. In this mode of DMA data transfer, only one byte of data is transferred at a time, hence the data transfer speed is slow. DMA controller send HOLD signal to microprocessor and wait for HLDA (Acknowledge) signal. After receiving HLDA signal from microprocessor, the DMA enter into master mode and gain the control of all system buses and execute only one DMA cycle to transfer one byte of data. After transferring one byte of data, the DMA controller disable HOLD signal, enter into slave mode and transfer the control of system buses to the microprocessor. This process is repeated to transfer all data bytes. Means, the DMA controller enables and disable HOLD signal for all data bytes transfer.

ii. Block Transfer: In this mode of DMA data transfer, the block of data bytes is transferred continuously. During the DMA data transfer, the microprocessor is disconnected from the system buses, hence the microprocessor cannot execute its own programs. N number of DMA cycle is added into the machine cycle of the microprocessor where N indicates numbers of bytes to be transferred. In this mode, the DMA controller sends HOLD signal to the microprocessor to gain the control of the system buses and wait for HLDA signal. After receiving HLDA signal, the DMA controller enters into the master mode and starts data transfer operation. After transferring all data bytes of the block, the DMA controller disable HOLD signal and enter into the slave mode. This mode DMA data transfer is faster than single byte mode.

iii. Hidden or Transparent DMA: In the machine cycle of microprocessor, there are some states during which all buses are not used by the microprocessor means it floats system buses. During these states, the microprocessor is isolated from the system buses and DMA controller transfer data between I/O device and memory. This is slowest DMA data transfer. In this method, additional logic i.e. hardware is required to detect the idle states when the microprocessor floats its buses.

(g) Draw the interfacing diagram of 7-segments display to 8085 through 8255.

Ans: **(4M- diagram)**



2. Attempt any Four:

(a) State any eight important features of 8085.

Ans: Any other correct feature should be considered. (1/2M-each feature).

i. It has 8 bit data bus, 8bit ALU. So it is 8 bit microprocessor.
ii. It has 16 bit address bus.
iii. It can access 64 kb external memory.
iv. It requires +5v power supply.
v. It requires 6 MHz crystal oscillator. 3 MHz is operating frequency.
vi. It offers 5 hardware interrupts & 8 software interrupts.
vii. It supports DMA feature using HOLD & HLDA Pins.
viii. Serial communication is possible through the pins SID & SOD.
ix. $2^8 = 256$ input & output devices can be interfaced with 8085

(b)Write assembly language program to add 8-bit numbers.

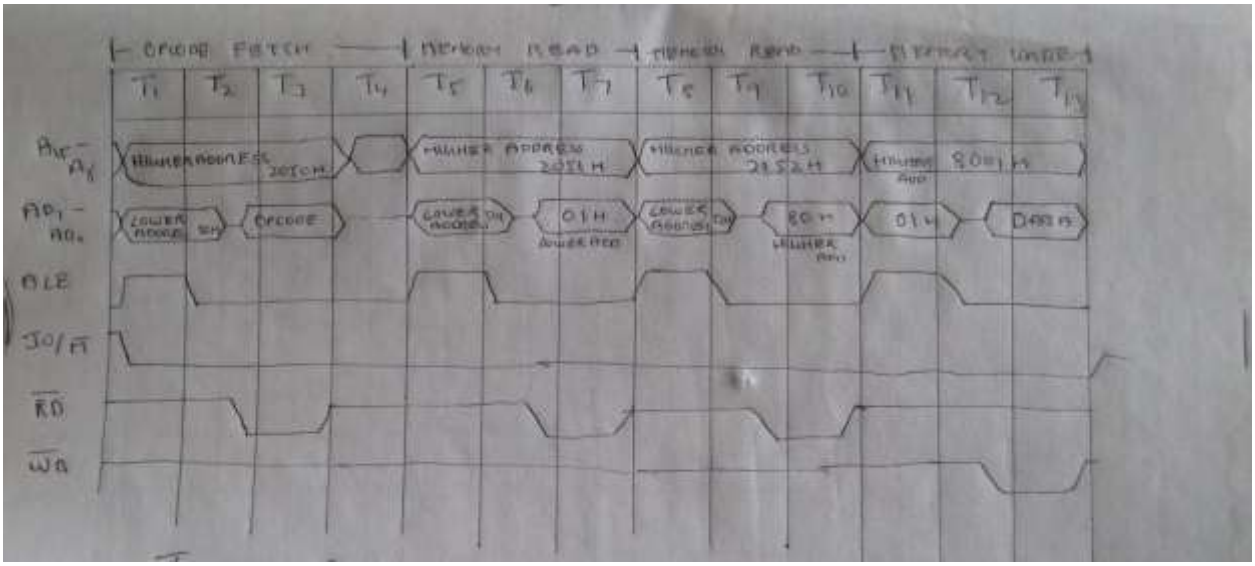
Ans: Any other correct program should be given marks (3M- program, 1M- comments)

Prog:

```
MOV A, 05H      ; Take 1ST number in accumulator
MOV B, 03H      ; Move 2ND number in register B
ADD A, B        ; add two numbers
STA 2500H       ; store answer at 2500H location
HLT
```

(c) Draw the timing diagram of STA 7000H instruction.

Ans: **NOTE: Timing diagram is same for any given memory location .In the diagram only 8001 should be replaced with 7000H, rest remains same**



(d) List the hardware interrupts used in 8085 and mention their vector location and priority.

Ans: (1M- list, 3M- vector location and priority)

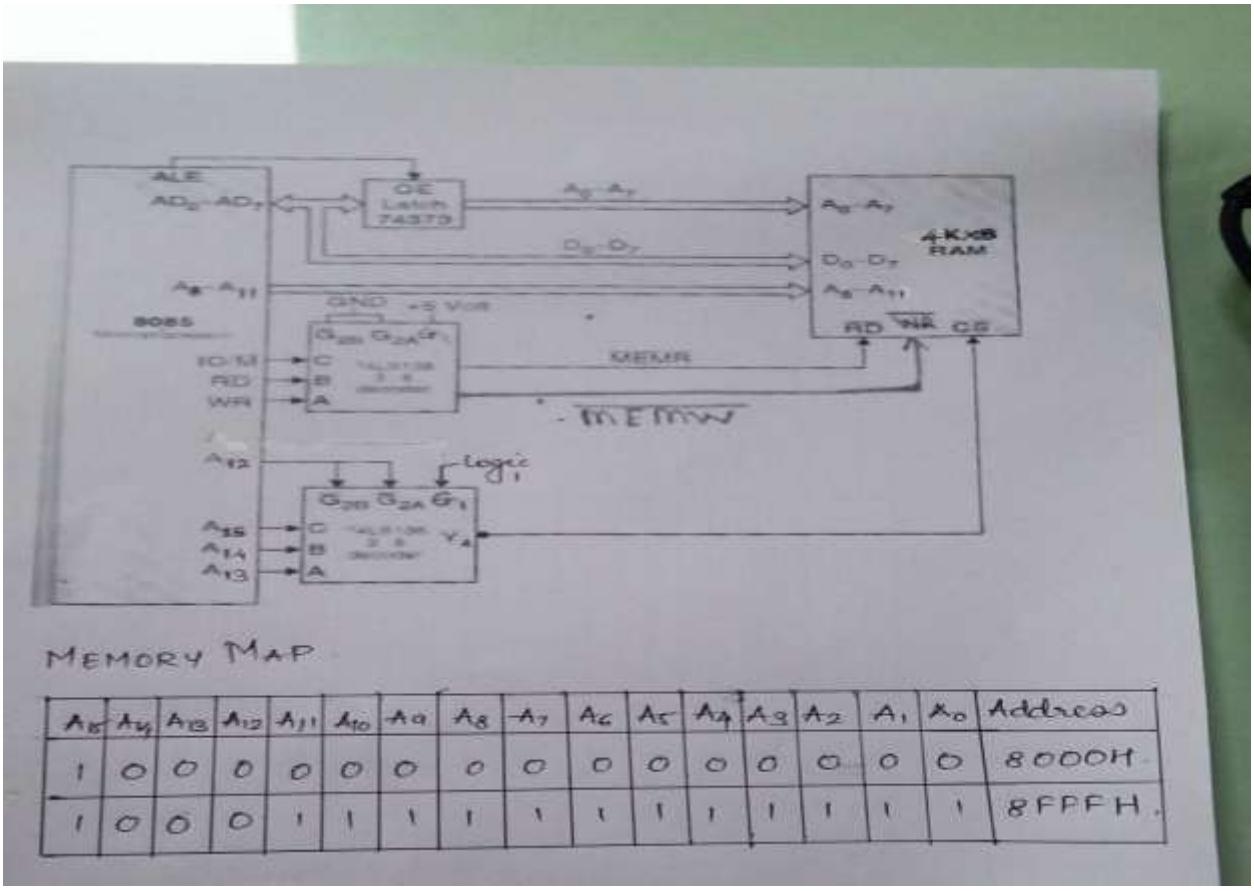
It has five hardware interrupts:

1. TRAP 2.RST 7.5 3.RST 6.5 4.RST 5.5 INTR

Interrupt type	Trigger	Priority	Maskable	Vector address
TRAP	Edge and Level	1st	No	0024H
RST 7.5	Edge	2nd	Yes	003CH
RST 6.5	Level	3rd	Yes	0034H
RST 5.5	Level	4th	Yes	002CH
INTR	Level	5th	Yes	-

(e) Interface 4K byte of RAM to 8085. Draw the memory map.

Ans: (3M- diagram, 1M- memory map)



MEMORY MAP

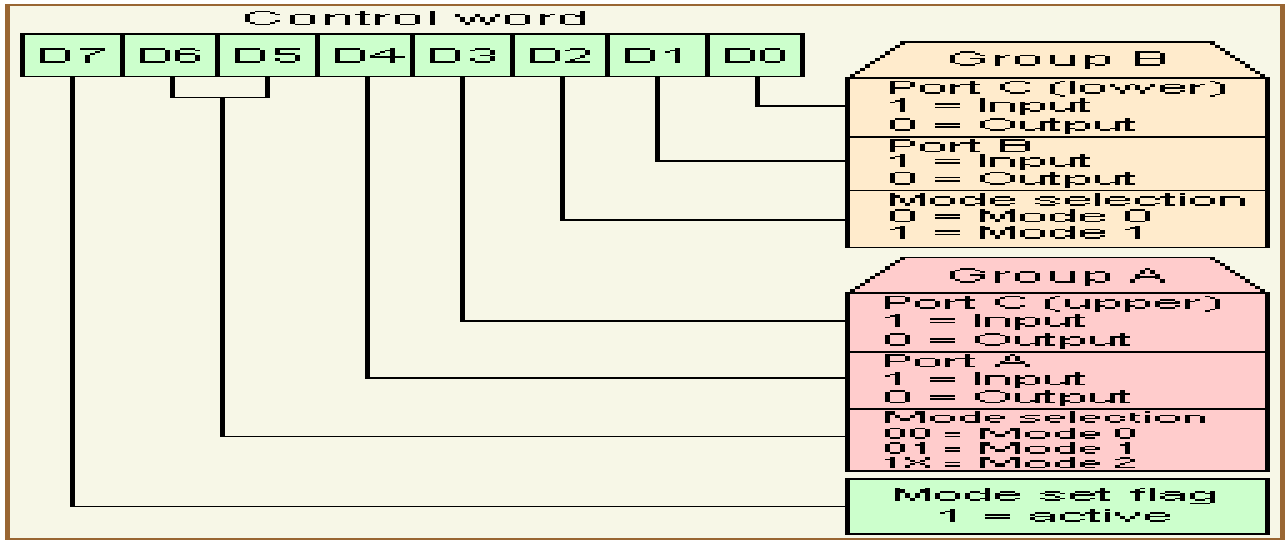
A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	Address
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8000H
1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	8FFFH

(f) Draw the control word format of 8255 for the following:

(i) All port as output in mode 0.

(ii) Port A and Port B input in model and port C as output in mode 0.

Ans: (2M- each)



i) All port as output in mode 0.

1	0	0	0	0	0	0	0

= 80H

ii) Port A and Port B input in mode and port C as output in mode 0.

1	0	0	1	0	0	1	0

= 92H

3. Attempt any FOUR:

(a) Draw the flag register of 8085 and explain the function of each bit.

Ans. (Format: 2 marks, explanation: 2 marks)

S	Z	X	AC	X	P	X	CY

Flag Register is given by:

S: Sign flag is set when D7 bit of ACC is set otherwise reset.

Z: Zero flag is set when result of an operation is 0 otherwise reset.

Ac: Auxiliary carry flag is set when there is a carry out of lower nibble or lower four bits of the operation otherwise reset.

CY: flag is set when there is carry generated by an operation otherwise reset.

P: Parity flag is set when result contains even number of 1's otherwise reset.

(b**) Explain the addressing modes of 8085 with suitable example**.

Ans.(any 4 addressing modes ½ mks explain ,1/2 mks example for each)

1) Immediate Addressing mode:-

In this mode of addressing the 8 bit or 16 bit operand (data) is a part of instruction . MVI A,20H

2) Register Addressing mode:-

In this mode of addressing the operand (data) is in one of the general purpose register or accumulator.

MOV B, A

3) Direct Addressing mode:-

In this mode of addressing the address of operand (data) is a part of instruction. LDA 6020H

4) Indirect Addressing mode:-

In this mode of addressing the address of the operand (data) is specified by a register pair. MOV B, M .

5) Implicit / Implied Addressing mode:-

In this mode of addressing the operand (data) is in accumulator. RAR

Note: any valid example can be considered.

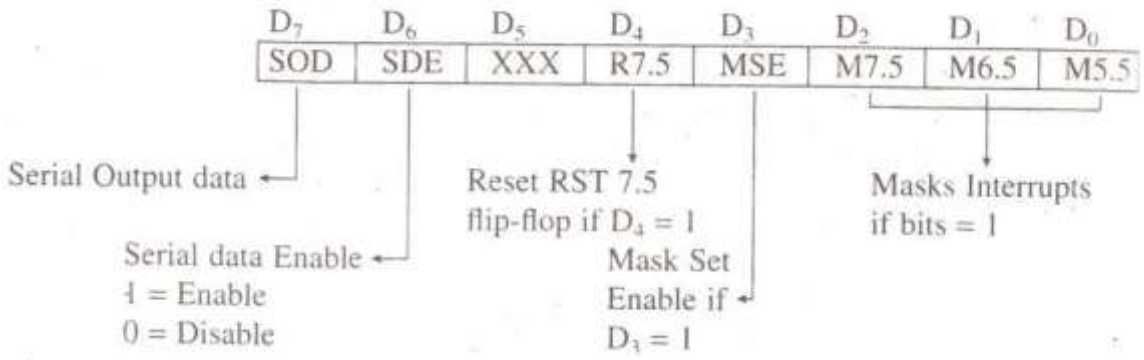
(c) Write assembly language program to exchange the lower and upper nibble of byte.

Ans:(3 mks program, 1 mks commentAny correct program with comments-4 Marks)

```
MVI C,00                ; Initialize Counter C with 00
LDA 4200                ; Load Content of memory loc 4200H into accumulator
MOV C,A                 ; move contents of accumulator to register C
ANI OF                  ; Mask the lower nibble of byte
MOV B,A                 :move contents of accumulator to reg. B
MOV A,C                 ; move contents of Reg C into Reg A
ANI F0                  ; Mask the Higher Nibble of Byte
ADD B                   ; Add contents of R Rotate Contents of Accumulator right eg B with
contents of Accumulator
STA 4202                ;store the result in 4202H
HLT                     ;halt
```

(d) Draw the format of SIM instruction and explain the function of each bit.

Ans:((2 Marks—format,2 Marksexplaination)



SOD – Serial Output Data: Bit D7 of the accumulator is latched into the SOD output line and made available to a serial peripheral if bit D6=1;

SDE- Serial Data Enable: If this bit=1, it enables the serial output. To implement serial output, this bit needs to be enabled.

XXX- Don't care

R7.5-Reset RST 7.5 if this bit=1, RST 7.5 flip-Flop is reset. This is an additional control to reset RST 7.5.

MSE- Mask Set Enable: if this bit is high, it enables the functions of bits D2, D1, D0. This id master control over all the interrupt masking bits. If this bit is low, bits D2, D1 and D0 do not

have any effect on the masks. M7.5-D2=0,
RST 7.5 is enabled

=1, 7.5 is masked or disabled

M6.5-D1=0, RST 6.5 is enabled

=1, 6.5 is masked or disabled

M5.5-D0=0, RST 5.5 is enabled

=1, 5.5 is masked or disabled

(e) Differentiate between I/O mapped I/O and mapped I/O (any four points).

Ans: (Any 4 points. Each point carries 1 mark)

Sr. No	Memory mapped I/O	I/O mapped I/O
1	In this technique I/O and memory both are treated as Memory.	In this technique I/O is treated as I/O and memory is treated as Memory
2	In this case both I/O and memory have a 16 bit address.	In this case I/O has an 8 bit address and memory has a 16 bit address.
3	All memory related instructions are applicable for I/O devices and memory.	IN and OUT instructions are used for I/O devices and memory related instructions for memory.

4	Size of memory is reduced	Size of Memory is not reduced.
5	Arithmetic and logical operations can be directly performed on the I/O ports	Arithmetic and logical operations cannot be directly performed on the I/O ports
6	Can interface maximum memory of 64 KB which also includes the I/O ports.	Can interface maximum memory of 64 KB and 256 I/O ports.
7	The data transfer is possible between any register and I/O port	The data transfer is possible only between Acc and I/O port.

(f) State any four important feature of 8355.

Ans; (Any 4 points,1 Mark each point)

- It has 2 I/O ports: Port A, Port B,
- Supports single mode: Simple I/O mode
- Each I/O pin can be individually programmed with the help of data direction register(DDR)
- It has 2K of ROM

4. Attempt any FOUR:

(a) State the function of program counter and stack counter.

Ans: (2 Marks—Each function)

Program Counter (PC): Program is a sequence of instructions. As mentioned earlier, microprocessor fetches these instructions from the memory and executes them sequentially. The program counter is a special purpose register which, at a given time, stores the address of the next instruction to be fetched. Program counter acts as a pointer to the next instructions. How processor increments program counter depends on the nature of the instructions; for one byte instruction it increments program counter by two and for three byte instructions it increments program counter by three such that program counter always points to the address of the next instructions.

In case of JUMP and CALL instructions, address followed by JUMP and CALL instructions is placed in the program counter. The processor then fetches the next instrucions from the new address specified by JUMP or CALL instruction. In conditional JUMP and conditional CALL instructions, if the condition is not satisfied, the processor increments program counter by three so that it points the instruction followed by conditional JUMP or CALL instruction; otherwise processor fetches the next instruction from the new address specified by JUMP or CALL instruction.

Stack counter:

The **stack** is a sequence of memory locations set aside by a programmer to store/retrieve the contents of accumulator, flags, program counter and general purpose registers during the execution of a program. The **stack pointer controls** addressing of the stack. The SP holds the address of the top element of data stored in the stack.

(b) State the function of LDA address and SHLD address instruction.

Ans. (Each instruction: 2 Marks)

LDA addr: Load data into A register directly from the address given within the instruction.

This instruction copies the contents of the memory location whose address is given within the instruction into the accumulator. The contents of the memory location remain unchanged.

Example: LDA 2000H :This instruction will copy the contents of the memory location 2000H into the Accumulator.

LDA 1200H : This instruction will copy the contents of the memory location 1200H into the

Accumulator..

SHLD 16-bit address: This instruction copies contents of register H and L to two consecutive memory locations without modifying H and L.

The 16 bit address of the first memory location is specified along with the instruction. It is a 3 byte instruction.

No flags are affected, the address mode is direct.

Example: SHLD 0300H: Copies the contents of L register to memory location 0300H and H to memory location 0301H

(c) Write assembly language program to transfer 5bytes of data starting from 1000H TO 2000H

Onwards.

Ans: **(Program with suitable Comments-- 4 M)**

M VI C, 05h --------- Counter for no of blocks to be transferred

LXI B, 1000h ------ Initializing source address in BC pair

LXI H, 2000h -------- Initializing destination address in HL pair

L1: LDAX B -------- Source address ----- (ACC)

MOV M, A ---------- (ACC) ----- Destination address

INX B ---------- Get next source address

INX H ---------- get next destination address

DCR C ---------- Decrement counter by 1

JNZ L1 ----------If counter $\neq 0$ continue to transfer, else stop

HLT

(**d) List the interrupt related instruction. Explain any two**.

Ans:((List –1 Mark,Explain any two--3 Marks)

The interrupt relate instructions are:

RST n

EI- Enable interrupt

DI- Disable interrupt

RIM- Read input mask

SIM- Set input mask

Restart

RST 0-7 : The RST instruction is equivalent to a 1-byte call instruction to one of eight memory locations depending upon the number. The instructions are generally used in conjunction with interrupts and inserted using external hardware. However these can be used as software instructions in a program to transfer program execution to one of the eight locations. The addresses are:

EI- (Enable interrupts)

It is 1-byte instruction which set the

- Interrupt enable Flip-flop & enables the interrupt process.
- System reset or an interrupt disables the interrupt process.

DI - (Disable interrupt)

It is 1 bite instruction which reset interrupt enable flip-flop & disables the interrupt

RIM – (read interrupt mask)

This is 1 byte multipurpose instruction used to read the status of interrupts 7.5, 6.5, & 5.5 and also

to read serial data input bit

OR Format

D7	D6	D5	D4	D3	D2	D1	D0
SID	I7	I6	I5	IE	7.5	6.5	55

(**e) Write assembly language program to generate square wave on SOD line.**

Ans:((Any correct program with comments–4 Marks)

Source program :

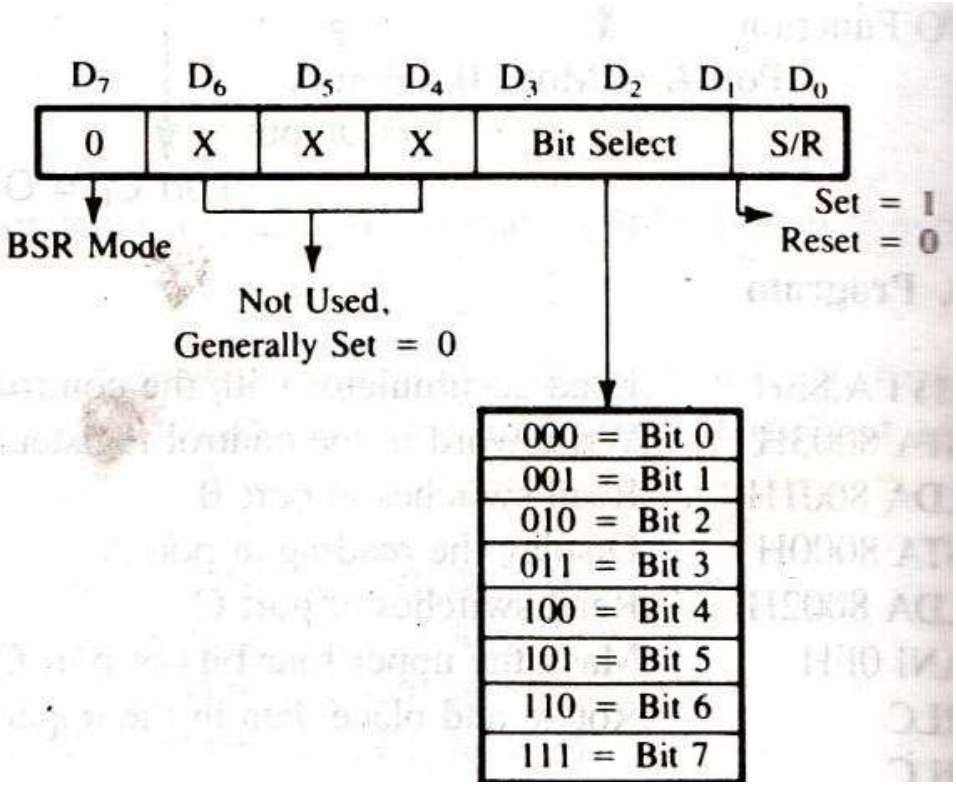
```
LXI SP, 27FFH        : Initialize stack pointer
LXI B, 1388H         : Initialize counter with count 5000.
BACK: MVI A, COH
 SIM                 : Send high on SOD pin
CALL DELAY           : Wait for 0.5 msec
MVI A, 40H           : Send low on SOD pin
SIM
CALL DELAY           : wait for. 5 msec
DCX B                : Decrement count by 1
MOV A, C
ORA B                : Check if count = 0
JNZ BACK             : If not, repeat
HLT                  : Stop program execution
Delay subroutine:
 DELAY               : LXI D, Count Back:
 DCX D
 MOV A, D
 ORA E
 JNZ Back
 RET
```

(f) Explain the operating modes of 8255.

Ans: BSR mode and I/O mode

BIT SET/RESET MODE:

- The PORT C can be Set or Reset by sending OUT instruction to the CONTROL registers.

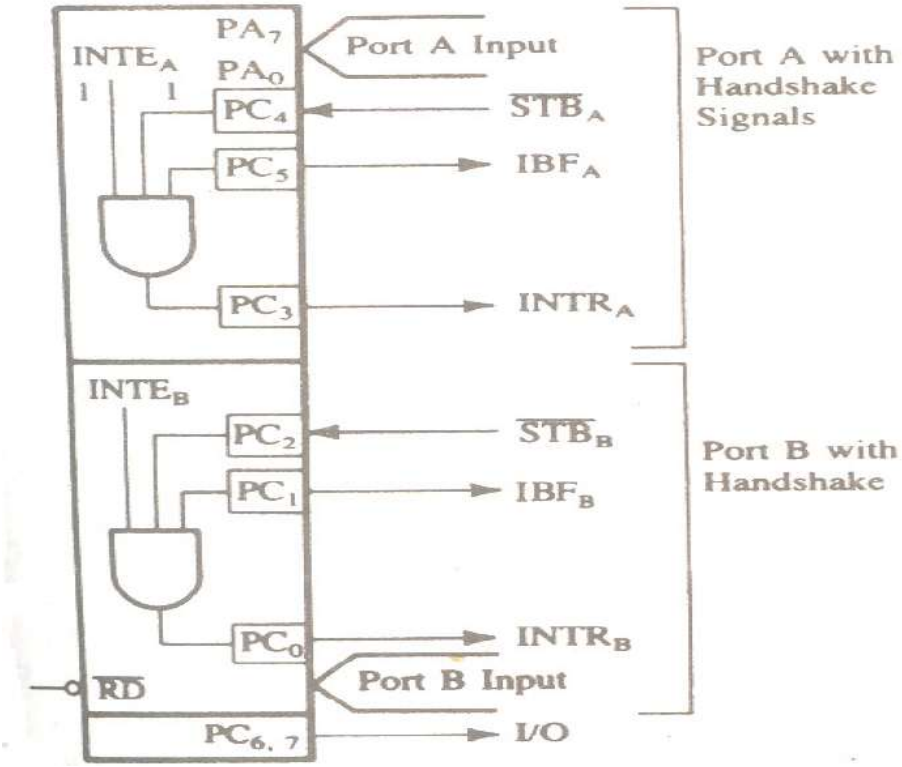


I/O MODES:

- MODE 0(Simple input / Output):
- In this mode , port A, port B and port C is used as individually (Simply).
- Features:
- Outputs are latched , Inputs are buffered not latched.
- Ports do not have Handshake or interrupt capability.

MODE 1 :(Input/output with Hand shake):

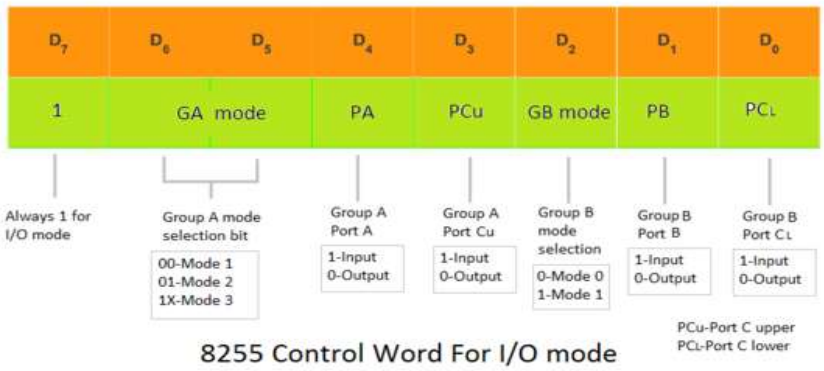
- In this mode, input or output is transferred by hand shaking Signals.
- Handshaking signals is used to transfer data between whose data transfer is not same.



MODE 2:bi-directional I/O data transfer:

- This mode allows bidirectional data transfer over a single 8-bit data bus using handshake signals.
- This feature is possible only Group A
- Port A is working as 8-biy bidirectional.
- PC3-PC7 is used for handshaking purpose.
- The data is sent by CPU through this port , when the peripheral request it.

CONTROL WORD FORMATS:



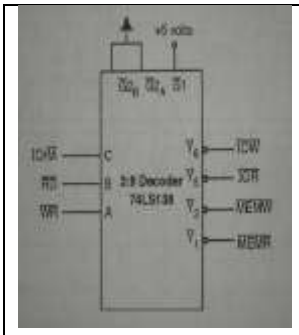
8255 Control Word For I/O mode

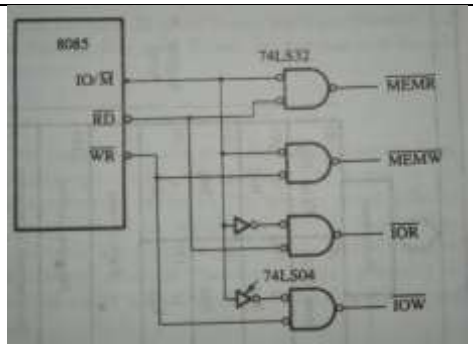
5. Attempt any FOUR: 16

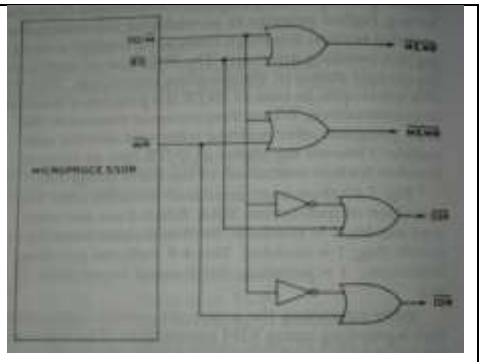
(a) List the different control signals in 8085 and draw the suitable diagram to generate control Control.

Ans: Any one correct diagram should be given marks (1M- names, 1M for each diagram)

i. IOR
ii. IOW
iii. MEMR
iv. MEMW







(b) Write assembly language program to find largest number out of five numbers stored from 2000H onwards and store the result at 4000H.

Ans: (3M- correct program, 1M-comments)

Program:

Sample data

(2000H) = 04
(2001H) = 34H
(2002H) = A9H
(2003H) = 78H
(2004H) =56H
Result = (4000H) = A9H

Program

```
LDA 2000H
MOV C, A          : Initialize counter
XRA A             : Maximum = Minimum possible value = 0
LXI H, 2201H      : Initialize pointer
BACK: CMP M       : Is number> maximum
JNC SKIP          : Yes, replace maximum
MOV A, M
SKIP: INX H
DCR C
JNZ BACK
STA 4000H         : Store maximum number
HLT               : Terminate program execution
```

(c) What is stack and sub routine? State any two advantage of sub routine.

Ans: (2M-each definition, 2M-advantages)

The **stack** is a sequence of memory locations set aside by a programmer to store/retrieve the contents of accumulator, flags, program counter and general purpose registers during the execution of a program. The **stack pointer controls** addressing of the stack. The SP holds the address of the top element of data stored in the stack.

Subroutine

Set of instructions which need to be executed frequently are stored separately from main program is known as subroutine.

(Note: Any relevant definition should be given marks)

Instruction

- CALL addr of subroutine
- e.g. CALL 2500h
- RET

Advantages of subroutines (any 2)

1. Large programs are lined into modules.
2. Different modules of programs in the form of subroutine are written, tested and debugs separately.
3. It improves the efficiency or the program by reducing errors.
4. Repeated group of instruction are written into the subroutines are called whenever required in the main program.
5. It save memory space and reduce time, size of program.
6. It reduces the time of market.

(d) State the function of SID and SOD line and give two advantages of serial communication.

Ans: (2M-each definition, 2M-advantages)

SID (Serial input data): This input signal is used to accept serial data bit by bit from the external device.

SOD (Serial output data): This is an output signal which enables the transmission of serial data bit by bit to the external device

Advantages of serial communication:

1. Bit by bit data transmission is possible for long distance communication and communication with cassette tapes or a CRT terminal .
2. Reduces cost of cabling.

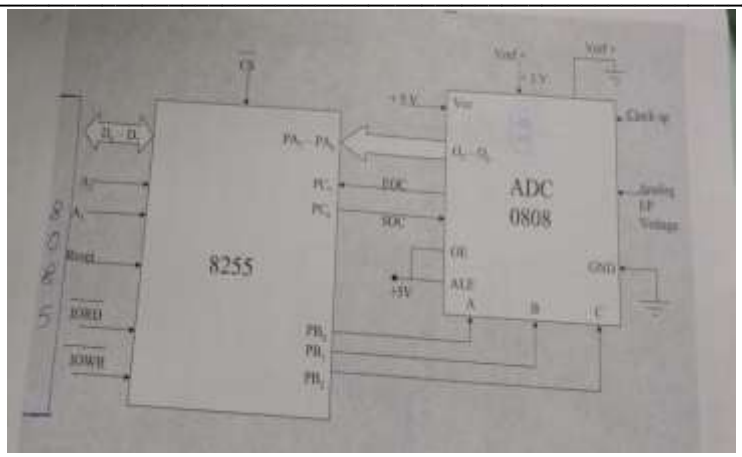
e) Compare 8155, 8255 and 8355 (any four points).

Ans: (1M-each point)

Sr.	8255	8155	8355
1.	Three 8 bit I/O Ports i.e. Port A, Port B and Port C	Two 8 bit I/0 ports i.e. Port A, Port B and one 6 bit I/ 0P orts i.e.Port C	Two 8 bit I/0 ports i.e. Port A, Port B
2.	No timer	Inbuilt 14- bit timer	No timer
3.	Separate data bus D_0-D_7 and address lines A0 and A1	Multiplexed AD_0-AD_7 bus	Multiplexed AD_0-AD_7 bus
4.	IO/M, ALE signals are not available.	IO/M, ALE signals are available	IO/M, ALE signals are available
5.	No memory	Inbuilt 256 byte of RAM	Inbuilt 2Kbytes of ROM

(f) Draw the interfacing of DAC with 8085 through 8255. Write assembly language to generate Square waveform using DAC.

Ans: (2M-diagram,2M-program)



Program:-

Control word

D7							D0
1	MA_I	MA_D	PA	PC_U	MB	PB	PC_C
1	0	0	0	0	0	0	0

=804

LABEL	MNEMONIC	OPERAND	COMMENTS
UP	MVIA	80H	**Initialize 8255**
	OUT	CWR	
	MVIA	00H	Out 00H to Port A and call delay to generate square wave.
	OUT	PORTA	
	CALL	DELAY	
	MVI	FFH	Out FFH to PA & call delay.
	OUT	PORTA	
	CALL	DELAY	
	JMP	UP	
DELAY	MVI C	85H	
BACK	DCR	H	
	JNZ	BACK	
	RET		

6. Attempt any FOUR: 16

(a) State the necessity of de-multiplexing of low order address/ data bus. Explain with suitable Diagram.

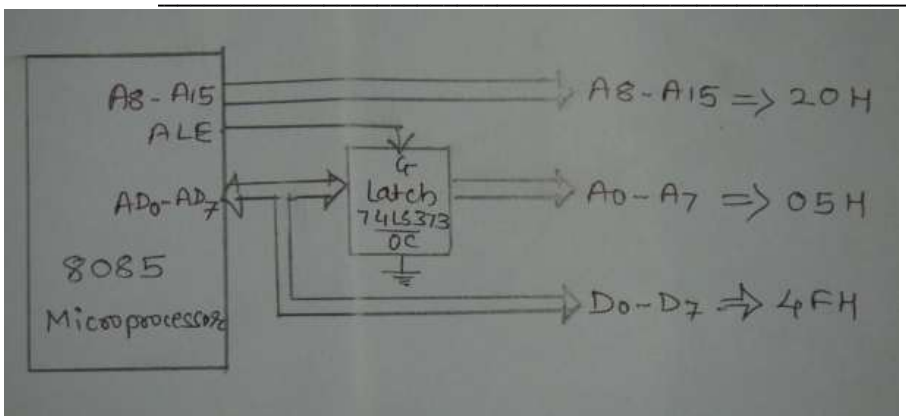
Ans: (1M- diagram, 3M-explanation)

In the 8085 higher order address lines A8 –A15 are directly available. The lower order address lines are multiplexed with data bus. i.e.AD0- AD7.

The de-multiplexing of address / data bus can be implemented by using tristate octal latch 74LS373 and the latch is controlled by ALE signal. When ALE goes high the address signals will be latched in octal latch 74LS373 and the o/p of latch will be A0 –A7.

When ALE goes low the latch will be disabled and the data is available on data bus D0-D7

for example. Address = 2005 H & Data = 4f H



(b) Write assembly language program to arrange five numbers in ascending order.

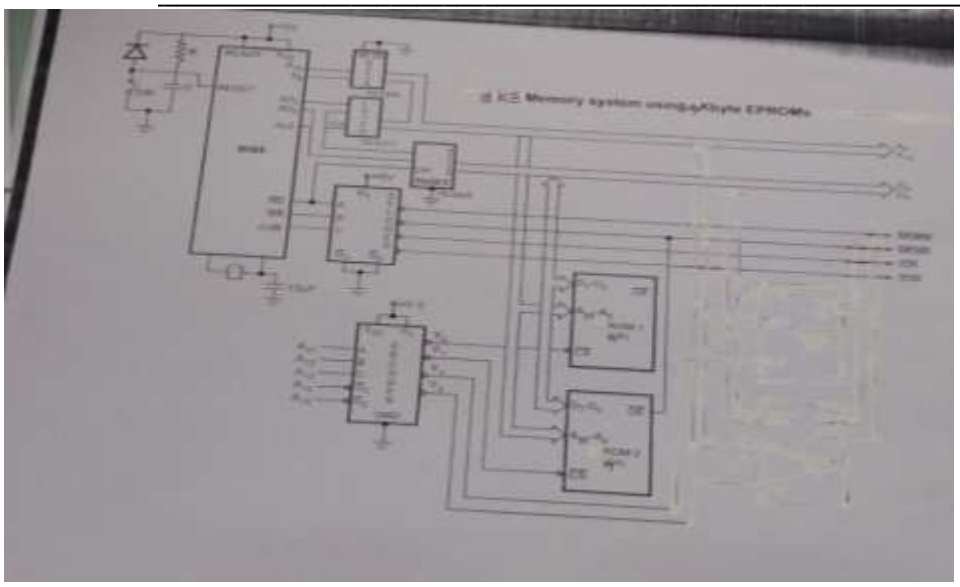
Ans: (3M-program, 1M -comments)

```
MVI B, 05              : Initialize counter

START  : LXI H, 2200H        : Initialize memory pointer
MVI C, 05H             : Initialize counter 2

BACK: MOV A, M               : Get the number
INX H                  : Increment memory pointer
CMP M                  : Compare number with next number
JC SKIP                : If less, don't interchange
JZ SKIP                : If equal, don't interchange
MOV D, M
MOV M, A
DCX H
MOV M, D
INX H                  : Interchange two numbers

SKIP:   DCR C          : Decrement counter 2
JNZ BACK               : If not zero, repeat
DCR B                  : Decrement counter 1
JNZ START
HLT                    : Terminate program execution
```

(c) Interface 8 K byte of ROM by using 4 K byte of memory chips.

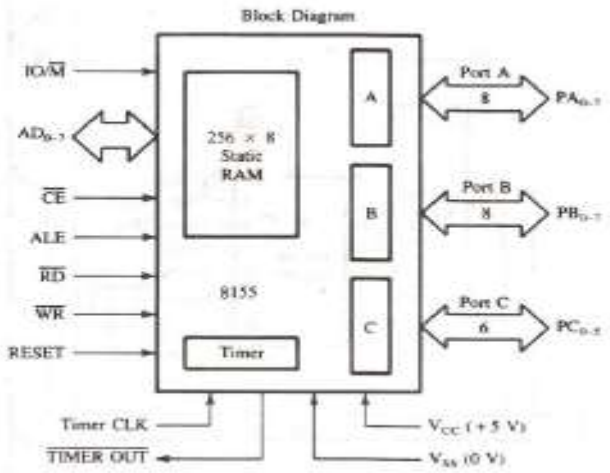
Ans: (4M-diagram)

4KB memory ,2^{12} = 4K so 12 address lines are required from A0 to A11



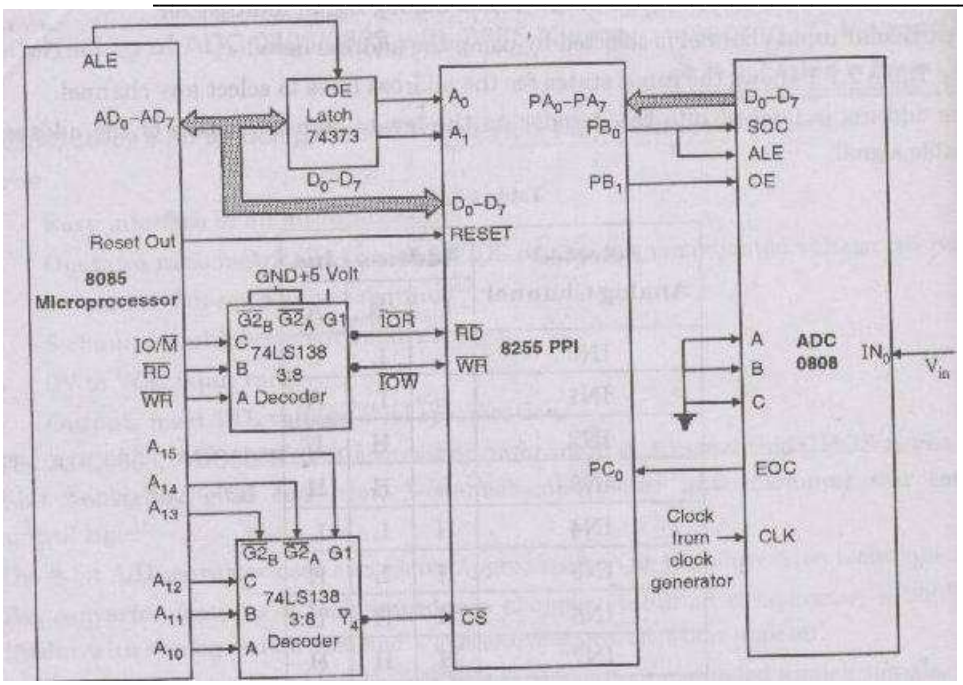
(d) Draw the block diagram of 8155.

Ans: (4M-diagram)



(e) Draw the interfacing of ADC to 8085 through 8255.

Ans: (4M-diagram)



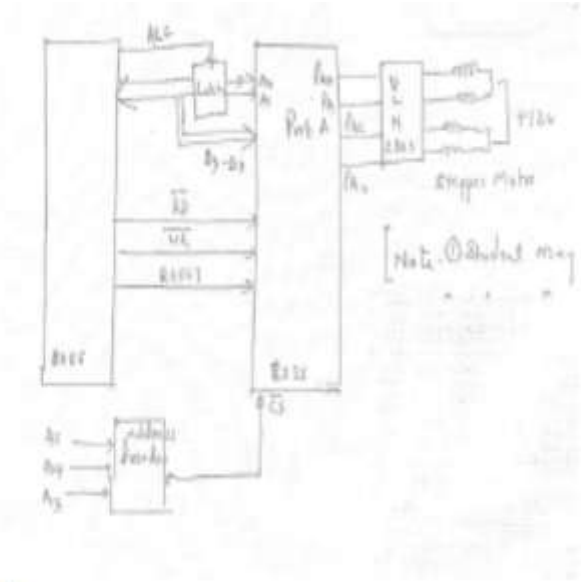
(f) Draw the interfacing of stepper motor with 8085 through 8255. Write assembly language Program to rotate stepper motor clockwise with 4-step sequence.

Ans: (2M-diagram,2M-program)

(Note 1 Student may use any port lines form PA, PB or PC

2. Motor driver transistor logic may be used

3. Program continues rotation; clockwise student may assume suitable data for program)



Look table
C100 – 0AH
C101 – 09H
C102 – 05H
C103- 06H

} input for stepper motor 8255 CWR -8011 – all port in O/P & motor

Program

Label	Mnemonics	Comments
	LX1 SP D0004	
	MV1 A, 804 OUT CWR	8255 initialization
AGAIN :	LXI H, C100	Lookup table
	MV 1 B , 04H	Step counter for I/P
UP:	MOV A,M	
	OUT Port A	
	CALL DELAY	
	INH X	
	DCR B	
	JNZ UP	
	JMP AGAIN	
DELAY:	LXI D, FFFFH	
GO:	DCX D	
	MOV A,E	
	ORA D	
	JNZ GO	
	RE T	