MICROPROCESSOR AND INTERFACING
(EE-329-F)

LAB MANUAL

V SEMESTER

Department of Electronics & Communication Engineering
Dronacharya College of Engineering
Khentawas, Gurgaon – 123506

MICROPROCESSOR AND INTERFACING LAB

LIST OF EXPERIMENTS
V SEM.(ECE, CSE, IT,BME)

EXPERIMENT NO. 1(A)

AIM : STUDY OF 8085-MICROPROCESSOR KIT.

APPARATUS: 8085 microprocessor kit.
 .
THEORY :
    Intel 8085 is an 8-bit microprocessor. It is 40-pin IC package fabricated on a single LSI chip. It uses a single +5 V supply. Its clock speed is about 3 MHz. It consists of three main sections: -

1.ALU (Arithmetic and logic unit):-

    The ALU performs the arithmetic and logical operation, addition, subtraction, logical AND, OR, EX-OR, Complement, Increment, Decrement, shift, clear.
2.Timing and Control Unit:-

    It generates timing and control signals, which are necessary for the execution of instruction.
3.Registers: -
    These are used for temporary storage of data and instruction. INTEL 8085 has following registers: -
i)   One 8 bit accumulator
ii)   Six 8 bit registers (B, C, D, E, H, L)
iii)   One 16 bit stack pointer, SP
iv)   One 16 bit program counter, PC
v)   Instruction register
vi)   Status register
vii)   Temporary registers

PC contains the address of next instruction.
IR holds the instruction until it is decoded.
SP holds the address of the stack top.
Accumulator is used during execution of program for temporary storage of data.

Status flags are as follows: -
i)   Carry (CS)
ii)   Zero (Z)
iii)   Sign (S)
iv)   Parity (P)
v)   Auxiliary Carry (AC)

PSW
        This 8-bit program status word includes status flags and three undefined bits.

Data and Address bus
        Data bus is 8- bit wide and 8 bits of data can be transmitted in parallel. It has 16-bit wide address bus as the memory addresses are of 16 bits.

CIRCUIT DIAGRAM(PIN DIAGRAM):-
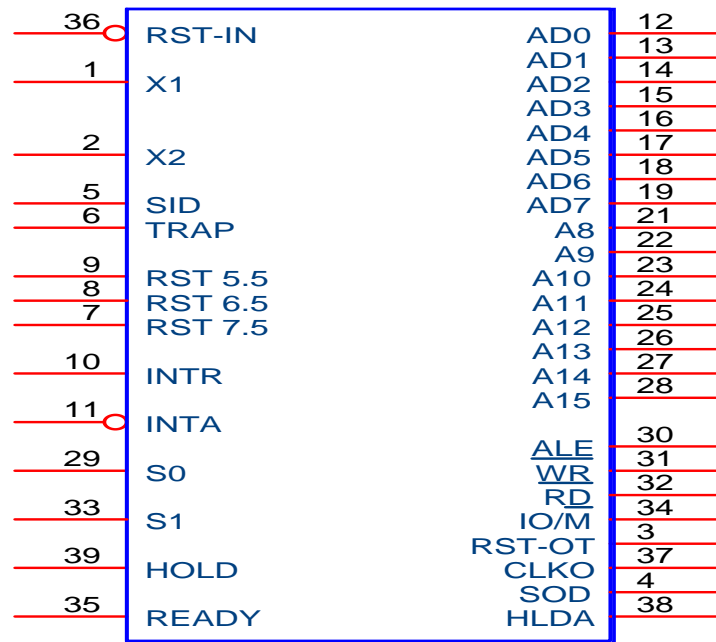
PIN CONFIGURATION

A8-A15 (Output):-
These are address bus and used for the most significant bits of memory address.

AD0-AD7 (Input/Output):-
These are time multiplexed address data bus. These are used for the least significant 8 bits of the memory address during first clock cycle and then for data during second and third clock cycle

ALE (Address Latch Enable)
It goes high during the 1st clock cycle of a machine. It enables the lower 8 bits of address to be latched either in the memory or external latch.

IO/M
It is status signal, when it goes high; the address on address bus is for I/O device, otherwise for memory.

So, S1
These are status signals to distinguish various types of operation

S1	So	Operation
0	0	Halt
0	1	Write
1	0	Read
1	1	Fetch

RD (output)
It is used to control read operation.

WR (output)
It is used to control write operation

HOLD (input)

It is used to indicate that another device is requesting the use of address & data bus.

HLDA (output)

It is acknowledgement signal used to indicate HOLD request has been received.

INTR (input)

When it goes high, microprocessor suspends its normal sequence of operations.

INTA (output)

It is interrupt acknowledgement signal sent by microprocessor after INTR is received.

RST 5.5,6.5,7.5 and TRAP

These are various interrupt signals. Among them TRAP is having highest priority

RESET IN (input)

It resets the PC to zero.

RESET OUT(output)

It indicates that CPU is being reset.

X1, X2 (input)

This circuitry is required to produce a suitable clock for the operation of microprocessor.   .

Clk (output)

It is clock output for user. Its frequency is same at which processor operates.

SID (input)

It is used for data line for serial input.

SOD (output)

It is used for data line for serial output.

Vcc

+5 volts supply

Vss

Ground reference

<div align="center">

EXPERIMENT NO .1 (B)

</div>

AIM : STUDY OF 8086 MICROPROCESSOR KIT.

APPARATUS: 8086 microprocessor kit.
.

THEORY : The 8086 is a 16-bit, N-channel, HMOS microprocessor. The term HMOS is used for high-speed MOS". The 8086 uses 20 address lines and 16 data lines. It can directly address up to $2^{20} =$ 1Mbytes of memory. The 16-bit data word is divided into a low-order byte and a high-order byte. The 20 address lines are time multiplexed lines. The 16 low-order address lines are time multiplexed with data, and the 4 high-order address lines are time multiplexed with status signals.

OPERATING MODES OF 8086
There are two modes of operation for Intel 8086, namely the minimum mode and the maximum mode. When only one 8086 CPU is to be used in a microcomputer system the 8086 is used in the minimum mode of operation. In this mode the CPU issues the control signals required by memory and I/O devices. In case of maximum mode of operation control signals are issued by Intel 8288 bus controller which is used with 8086 for this very purpose. When MN/MX is high the CPU operates in the minimum mode. When it is low the CPU operates in the maximum mode.

Pin Description For Minimum Mode
For the minimum mode of operation the pin MN/MX is connected to 5V d.c supply. The description of the pins from 24 to 31 for the minimum mode is as follows:

$\overline{INTA}$**(Output):** Pin no. 24 Interrupt acknowledge. On receiving interrupt signal the processor issues an interrupt acknowledge signal. It is active LOW.
ALE(Output) : Pin no. 25 Address latch enable. It goes HIGH during T1. The microprocessor sends this signal to latch the address into the Intel 8282/8283 latch.
$\overline{DEN}$**(Output) :** Pin no. 26 Data enable. When Intel 8286/8287 octal bus transceiver is used this signal acts as an output enable signal. It is active LOW.
DT/$\overline{R}$(Output) : Pin no. 27 Data Transmit/Receive. When Intel 8286/8287 octal bus transceiver is used this signal controls the direction of data flow through the transceiver. When it is High data are sent out. When it is LOW data are received.
M/$\overline{IO}$(Output) : Pin no. 28.Memory or I/O access. When it is HIGH the CPU wants to access memory. When it is LOW the CPU wants to access I/O device.
$\overline{WR}$ **(Output) :** Pin no. 29. Write. When it is LOW the CPU performs memory or I/O write Operation.
HLDA (Output) : Pin no. 30.HOLD acknowledge. It is issued by the processor when it receives HOLD signal. It is active HIGH signal. When HOLD request is removed HLDA goes LOW.
HOLD (Output) : Pin no. 31.Hold. when another device in microcomputer system wants to use the address and data bus, it sends a HOLD request to CPU through this pin. It is an active HIGH signal.

Pin Description For Maximum Mode

For the maximum mode of operation the pin MN/$\overline{\text{MX}}$ is made LOW. It is grounded. The description of the pins from 24 to 31 is as follows:

QS1, QS0(Output): Pin no. 24,25 Instruction Queue status. Logic are given below:

QS1	QS0	
0	0	No operation
0	1	1st byte of opcode from queue
1	0	Empty the queue
1	1	Subsequent byte from queue

$\overline{\textbf{S0}},\overline{\textbf{S1}},\overline{\textbf{S2}}$**(Output) :** Pin nos. 26,27,28.status signals. These signals are connected to the bus controller Intel 8288.The bus controller generates memory and I/O access control signals. Table for status signals is :

$\overline{\text{S2}}$	$\overline{\text{S1}}$	$\overline{\text{S0}}$	
0	0	0	Interrupt acknowledge
0	0	1	Read data from I/O port
0	1	0	Write data into I/O port
0	1	1	Halt
1	0	0	Opcode fetch
1	0	1	Memory read
1	1	0	Memory write
1	1	1	Passive state.

$\overline{\textbf{LOCK}}$**(Output) :** Pin no. 29.It is an active LOW signal. When it is LOW all interrupts are masked and no HOLD request is granted. In a multiprocessor system all other processors are informed by this signal that they should not ask the CPU for relinquishing the bus control.

$\overline{\textbf{RQ}}$ **/** $\overline{\textbf{GT}_1}$, $\overline{\textbf{RQ}}$ **/** $\overline{\textbf{GT}_0}$ **(Bidirectional) :** Pin no. 30,31. Local bus Priority control. Other processors ask the CPU through these lines to release the local bus. $\overline{\text{RQ}}$ / $\overline{\text{GT}_1}$ has higher priority than $\overline{\text{RQ}}$ / $\overline{\text{GT}_0}$

U1

22 READY	16 AD0
19 CLK	15 AD1
21 RESET	14 AD2
	13 AD3
18 INTR	12 AD4
	11 AD5
	10 AD6
	9 AD7
	8 AD8
	7 AD9
	6 AD10
	5 AD11
	4 AD12
	3 AD13
	2 AD14
	39 AD15
	38 A16/S3
	37 A17/S4
	36 A18/S5
	35 A19/S6
	34 $\overline{BHE}$/S7
	26 S0
	27 S1
	28 S2
31 RQ/GT0	32 $\overline{RD}$
30 RQ/GT1	29 LOCK
17 NMI	25 QS0
23 TEST	24 QS1
33 MX	
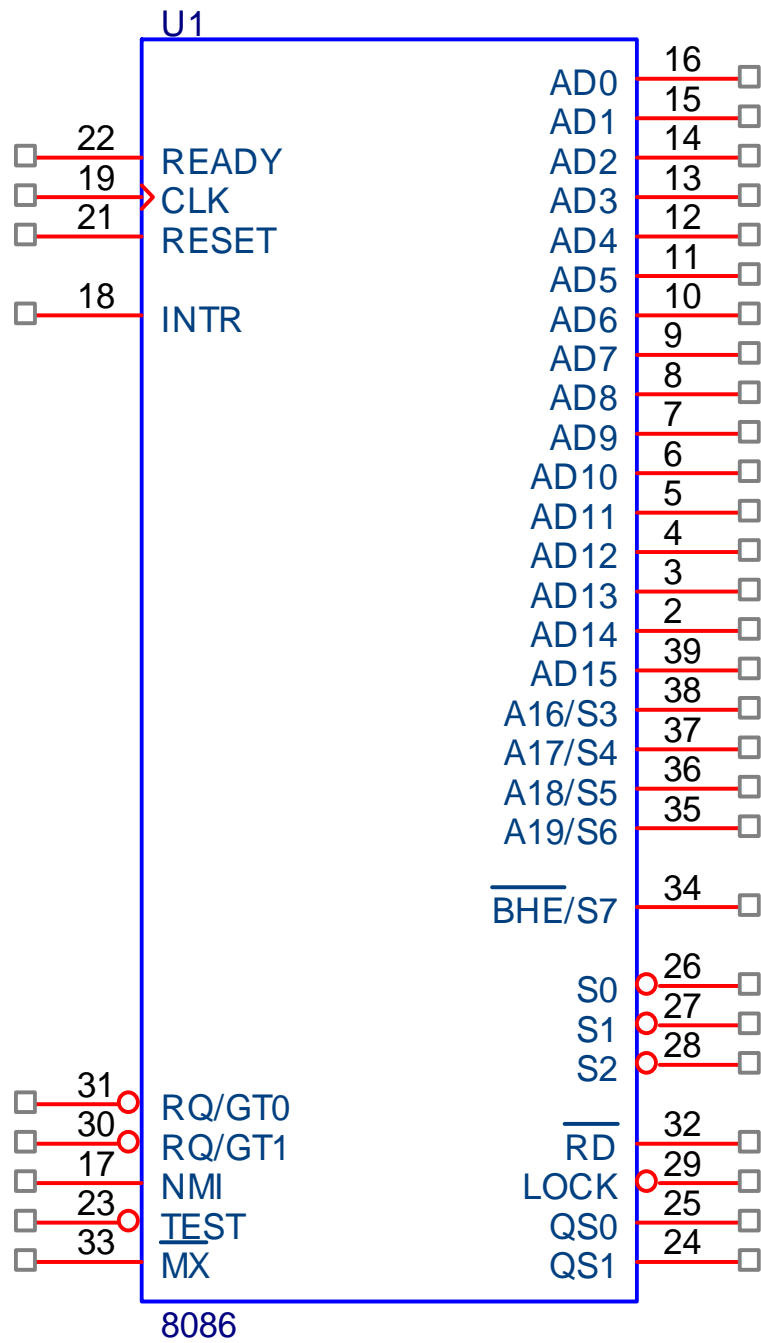
8086
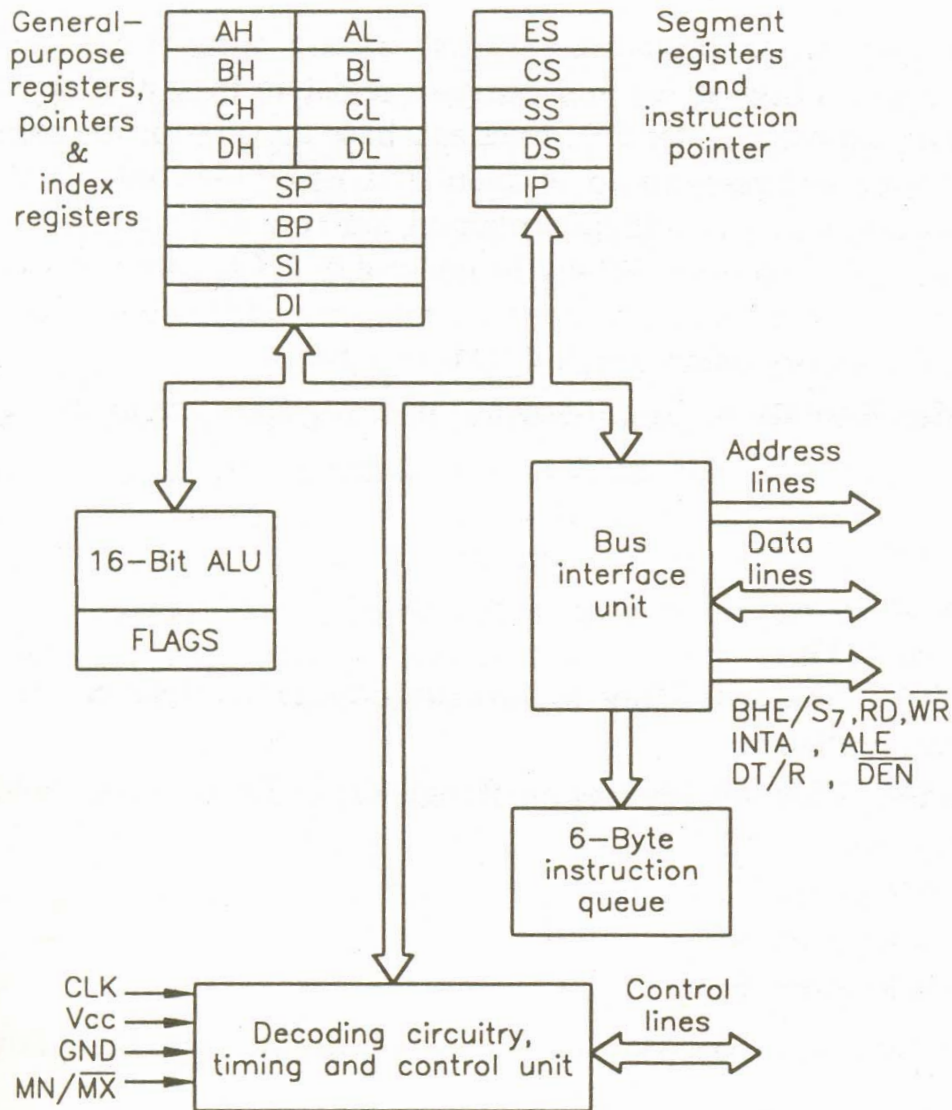
PIN DIAGRAM OF 8086

Block Diagram of Intel 8086 Microprocessor

BLOCK DIAGRAM OF 8086:

REGISTERS OF 8086 : The Intel 8086 contains the following registers:
   a) General Purpose Register
   b) Pointer and Index Registers
   c) Segment Registers
   d) Instruction Registers
   e) Status Flags

EXPERIMENT NO. 2(A)

AIM : WRITE A PROGRAM USING 8085 & VERIFY FOR :
   (a) ADDITION OF TWO 8-BIT NUMBERS.

APPARATUS : 8085 microprocessor kit, 5V power supply, Keyboard.

THEORY (Program)

Memory address	Machine code	Mnemonics	Operands	Commands
7000	21,01,75	LXI	H,7501	Get address of 1st no. in HL pair
7003	7E	MOV	A,M	Move Ist no. in accumulator
7004	23	INX	H	HL points the address 7502H
7005	86	ADD	M	Add the 2nd no.
7006	23	INX	H	HL points 7503H
7007	77	MOV	M,A	Store result in 7503H.
7008	CF	RST 1		Terminate
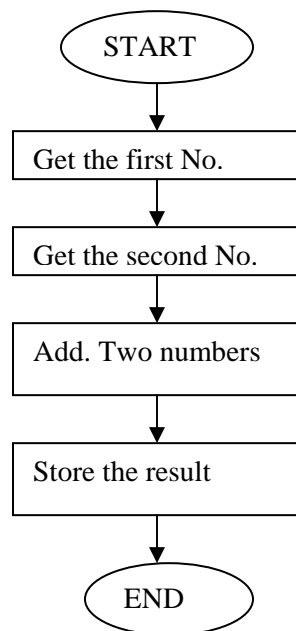
CIRCUIT DIAGRAM / BLOCK DIAGRAM:-

PROCEDURE:-

ANSHUMAN	SCIENTECH
	Reset
S	Exmem
Enter Enter	Starting Address
Program Address	Next
Write Program	Write Program
Execution Steps	**Execution Steps**
Esc	Reset
G	GO
Enter-enter	Starting Address
Prog. Address	Fill
Enter	Reset
S	Exmem
Enter	Result Address
Any key-2	
Enter-2	
Register Name	

INPUT DATA
        7501- 13H
        7502- 12H

OUTPUT DATA
        7503- 25H

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.

EXPERIMENT NO. 2(B)

AIM : WRITE A PROGRAM USING 8085 & VERIFY FOR :
      (b) ADDITION OF TWO 16-BIT NUMBERS(WITH CARRY).

APPARATUS : 8085 microprocessor kit, 5V power supply, Keyboard.

THEORY (Program)

Memory address	Label	Machine code	Mnemonics	Operands	Commands
7000		2A,01,76	LHLD	7601H	Get 1st no. in HL pair from memory (7601)
7003		EB	XCHG		Exchange cont. of DE ↔HL
7004		2A,03,76	LHLD	7603H	Get 2st no. in HL pair from location 7603
7007		0E,00	MVI	C,00H	Clear reg. C.
7009		19	DAD	D	Get HL+DE & store result in HL
700A		D2,12,70	JNC	7012(loop)	If no carry move to loop/if carry then move to next step.
700D		0C	INR	C	Increment reg C
700E		79	MOV	A,C	Move carry from reg. C to reg. A
7011		32,02,75	STA	7502	Store carry at 7502H
7012	loop	22,00,75	SHLD	7500	Store result in 7500H.
7015		CF	RST1		Terminate
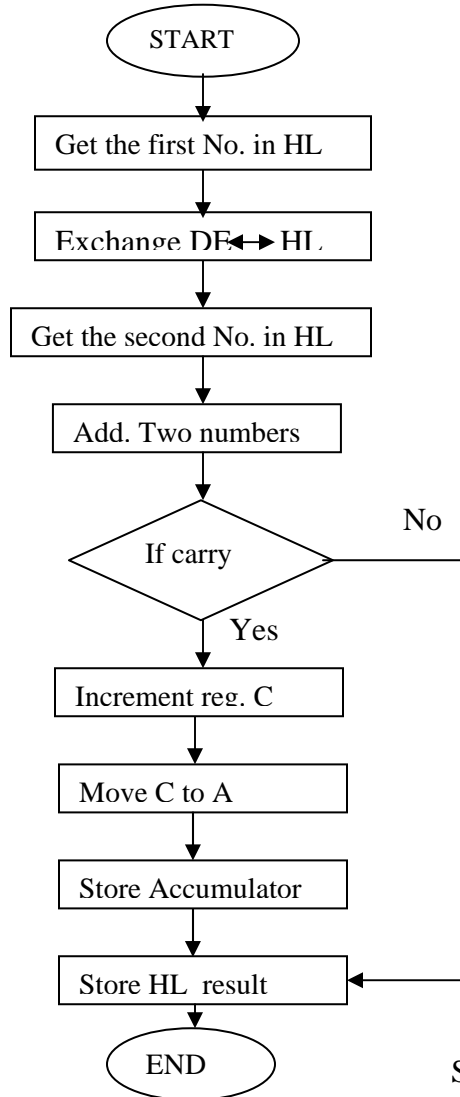
CIRCUIT DIAGRAM / BLOCK DIAGRAM:-

```
                    ( START )
                        |
                        v
        +-------------------------------+
        | Get the first No. in HL       |
        +-------------------------------+
                        |
                        v
        +-------------------------------+
        | Exchange DE <--> HL           |
        +-------------------------------+
                        |
                        v
        +-------------------------------+
        | Get the second No. in HL      |
        +-------------------------------+
                        |
                        v
        +-------------------------------+
        | Add. Two numbers              |
        +-------------------------------+
                        |
                        v
                   < If carry >  ----No---->
                        |                  |
                       Yes                 |
                        v                  |
        +-------------------------------+  |
        | Increment reg. C              |  |
        +-------------------------------+  |
                        |                  |
                        v                  |
        +-------------------------------+  |
        | Move C to A                   |  |
        +-------------------------------+  |
                        |                  |
                        v                  |
        +-------------------------------+  |
        | Store Accumulator             |  |
        +-------------------------------+  |
                        |                  |
                        v                  |
        +-------------------------------+  |
        | Store HL  result              | <+
        +-------------------------------+
                        |
                        v
                    ( END )
```

PROCEDURE:-

ANSHUMAN	SCIENTECH
S	Reset
Enter Enter	Exmem
Program Address	Starting Address
Write Program	Next
Execution Steps	Write Program
Esc	**Execution Steps**
G	Reset
Enter-enter	GO
Prog. Address	Starting Address
Enter	Fill
S	Reset
Enter	Exmem
Any key-2	Result Address
Enter-2 Register Name	

INPUT DATA

    7601 : 13H
    7602 : 31H
    7603 : 12H
    7604 : 10H

OUTPUT DATA

    7500 : 25H
    7501 : 41H
    7502 : 00H

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.

Question & Answer:

Q.l Explain MOV r,M ?
Ans: Move the content of memory to register.

Q.2 How many T-state are in MOV instruction?
Ans: 4 T-state.

Q.3 Explain the addressing mode of MOV r,M?
Ans: Register indirect.

Q.4 How many machine cycles are in MOV instruction?
Ans: 2 machine cycle.

Q.5 What is MOV M,r ?
Ans: move the content of register to memory

Q.6 Which flag is affected in MOV instruction?
Ans: none

Q.7 What is MVI r,data?
Ans: move immediate data to register

Q.8 How many T-state are in MVI instruction?
Ans: seven T-states.

Q.9 Explain the addressing mode of MVI r,data?
Ans: immediate

Q.10 How many machine cycles are in MVI instruction?
Ans: 3 machine cycles.

EXPERIMENT NO. 3(A)

AIM : WRITE A PROGRAM USING 8085 & VERIFY FOR :
   A. SUBTRACTION OF TWO 8-BIT NUMBERS. (DISPLAY OF BARROW).

APPARATUS : 8085 microprocessor kit, 5V power supply, Keyboard.

THEORY(Program) :

Memory address	Opcode	Mnemonics	Operands	Comments
7000	21,01,75	LXI	H, 7501	Get address of ist no. in HL pair
7003	7E	MOV	A, M	Move Ist no. in accumulator
7004	23	INX	H	HL points 7502H.
7005	96	SUB	M	Substract 2nd no. from Ist no.
7006	23	INX	H	HL points 7503 H.
7007	77	MOV	M, A	Move contents of acc. to memory
7008	CF	RST 1		Stop

CIRCUIT DIAGRAM / BLOCK DIAGRAM :-

PROCEDURE:-

	SCIENTECH
ANSHUMAN	Reset
S	Exmem
Enter Enter	Starting Address
Program Address	Next
Write Program	Write Program
Execution Steps	**Execution Steps**
Esc	Reset
G	GO
Enter-enter	Starting Address
Prog. Address	Fill
Enter	Reset
S	Exmem
Enter	Result Address
Any key-2	
Enter-2	
Register Name	

INPUT DATA
             7501 : 20H
             7502 : 10H

OUTPUT DATA
      7503 : 10H

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.
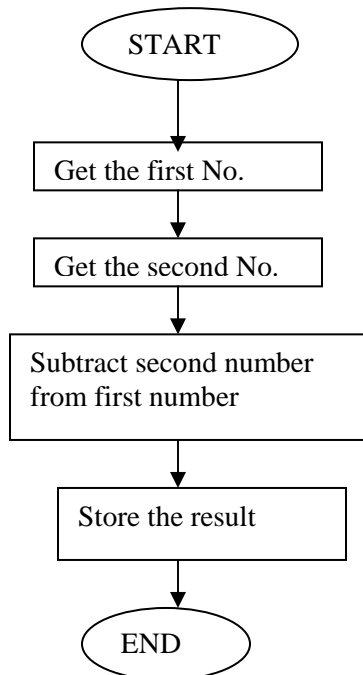
EXPERIMENT NO. 3 (B)

AIM : WRITE A PROGRAM USING 8085 & VERIFY FOR :
     B. SUBTRACTION OF TWO 16-BIT NUMBERS. (DISPLAY OF BARROW)

APPARATUS : 8085 microprocessor kit, 5V power supply, Keyboard.

THEORY (Program) :

Memory Address	Machine Code	Mnemonics	Operands	Comments
7000	2A, 01,75	LHLD	7501 H	Get 1st 16 bit no. in HL pair
7003	EB	XCHG		Exchange HL pair with DE.
7004	2A, 03,75	LHLD	7503 H	Get 2nd 16 bit no. in HL pair
7007	7B	MOV	A, E	Get lower byte of ist no.
7008	95	SUB	L	Subtract lower byte of 2nd no.
7009	6F	MOV	L, A	Store the result in reg. L
700A	7A	MOV	A, D	Get higher byte of Ist no.
700B	96	SBB	H	Subtract higher byte of 2nd no. with borrow
700C	67	MOV	H,A	Move from acc. To H
700D,E, F	22,05,75	SHLD	7505H	Store 16 bit result at 7505&7506
7010	CF	RST 1		Terminate
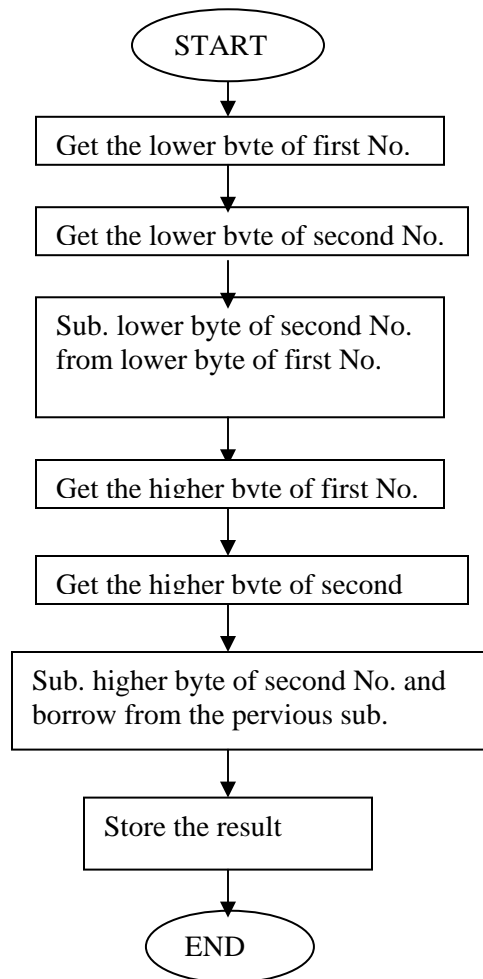
CIRCUIT DIAGRAM / BLOCK DIAGRAM :-

```
                    ( START )
                        |
                        v
        +-------------------------------+
        | Get the lower byte of first No.|
        +-------------------------------+
                        |
                        v
        +--------------------------------+
        | Get the lower byte of second No.|
        +--------------------------------+
                        |
                        v
        +--------------------------------+
        | Sub. lower byte of second No.  |
        | from lower byte of first No.   |
        +--------------------------------+
                        |
                        v
        +-------------------------------+
        | Get the higher byte of first No.|
        +-------------------------------+
                        |
                        v
        +-------------------------------+
        | Get the higher byte of second |
        +-------------------------------+
                        |
                        v
        +-------------------------------+
        | Sub. higher byte of second No. and|
        | borrow from the pervious sub. |
        +-------------------------------+
                        |
                        v
        +-------------------------------+
        |      Store the result         |
        +-------------------------------+
                        |
                        v
                    (  END  )
```

PROCEDURE:-

	SCIENTECH
ANSHUMAN	Reset
S	Exmem
Enter Enter	Starting Address
Program Address	Next
Write Program	Write Program
Execution Steps	**Execution Steps**
Esc	Reset
G	GO
Enter-enter	Starting Address
Prog. Address	Fill
Enter	Reset
S	Exmem
Enter	Result Address
Any key-2	

Enter -2
Register Name
INPUT DATA

      7501 : 30H
      7502 : 40H
      7503 : 10H
      7504 : 20H

OUTPUT DATA

      7505 : 20H
      7506 : 20H

PRECAUTIONS:-
Make sure that all the machine codes should be as per specified in the program.

Question & Answer:
Q.1 Explain LXI rp,data 16 ?
Ans: load register pair immediate.

Q.2 How many T-state are in LXI instruction?
Ans: 10 T –states.

Q.3 Explain the addressing mode of LXI rp,data?
Ans: Immediate

Q.4 How many machine cycles are in LXI instruction?
Ans: 3 machine cycles.

Q.5 What is LDA addr ?
Ans: load accumulator direct.

Q.6 How many T-state are in LDA instruction?
Ans: 13 T –states.

Q.7 Explain the addressing mode ofLDA addr?
Ans: Direct

Q.8 How many machine cycles are in LDA instruction?
Ans: 4

Q.9 What is STA addr?
Ans: store accumulator direct

Q.10 How many T-state are in STA instruction?
Ans: 13

EXPERIMENT NO. 4(A)

AIM : WRITE A PROGRAM USING 8085 FOR MULTIPLICATION OF TWO 8-BIT NUMBERS
BY BIT ROTATION METHOD & VERIFY.

APPARATUS : 8085 microprocessor kit,5 V power supply, Keyboard.

THEORY(Program)

Memory Address	Label	Machine Code	Mnemonics	Operands	Comments
7000		2A,01,75	LHLD	7501 H	Get Multiplicand in H-L pair.
7003		EB	XCHG		Exchange HL pair with DE pair
7004		3A,03,75	LDA	7503 H	Get 2nd no. in acc.
7007		21,00,00	LXI	H,0000	Initial product in HL=00
700A		0E,08	MVI	C,08H	Count=08 in reg .C
700C	Loop	29	DAD	H	Shift partial product left by 1 bit
700D		17	RAL		Rotate multiplication by 1 bit. Is multiplier = 1?
700E		D2,12,70	JNC	Ahead(7012)	No, go to ahead
7011		19	DAD	D	Product=Product + Multiplicand
7012	Ahead	0D	DCR	C	Decrement Count
7013		C2,0C,70	JNZ	Loop(700C)	
7016		22,04,75	SHLD	7504	Store result
7019		CF	RST 1		Terminate
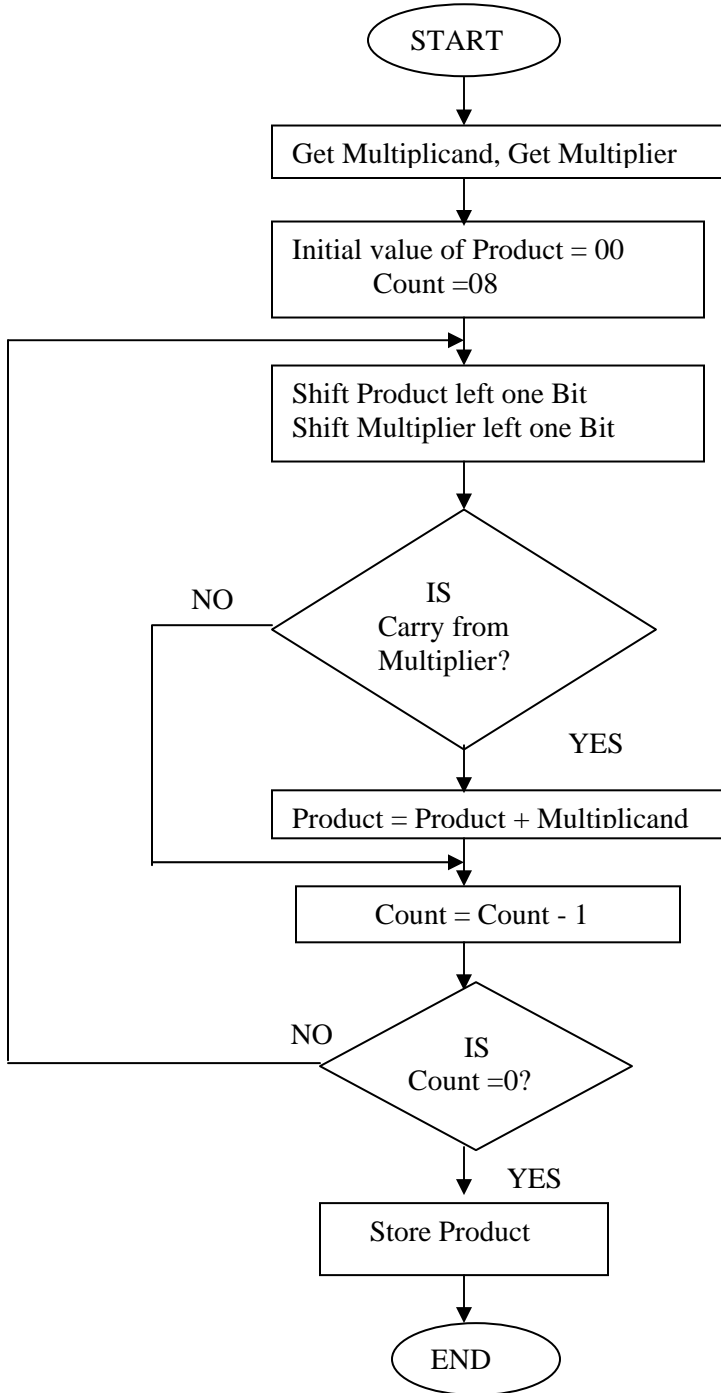
CIRCUIT DIAGRAM / BLOCK DIAGRAM :-

START

Get Multiplicand, Get Multiplier

Initial value of Product = 00
Count =08

Shift Product left one Bit
Shift Multiplier left one Bit

IS
Carry from
Multiplier?

NO

YES

Product = Product + Multiplicand

Count = Count - 1

IS
Count =0?

NO

YES

Store Product

END

PROCEDURE:-

ANSHUMAN	SCIENTECH
S	Reset
Enter Enter	Exmem
Program Address	Starting Address
Write Program	Next
Execution Steps	Write Program
	Execution Steps

Esc	Reset
G	GO
Enter-enter	Starting Address
Prog. Address	Fill
Enter	Reset
S	Exmem
Enter	Result Address
Any key-2	
Enter-2	
Register Name	

INPUT DATA

> 7501- 25H
> 7502- 00H
> 7503- 05H

OUTPUT DATA

> 7504- B9H
> 7505- 00H

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.

EXPERIMENT NO. 4(B)

AIM : WRITE A PROGRAM USING 8085 FOR DIVISION OF TWO 8-BIT NUMBERS BY REPEATED SUBTRACTION METHOD & TEST FOR TYPICAL DATA.

APPARATUS : 8085 microprocessor kit, 5V power supply, Key board.

THEORY (Program) :

Memory Address	Label	Machine Code	Mnemonics	Operands	Comments
7000		3A,01,75	LDA	Divisor(7501)	
7003		47	MOV B,A		Take divisor in reg,B
7004		3A,02,75	LDA	Dividend(7502)	Take dividend in reg,A
7007		0E,00	MVI	C,00	Quotient=00
7009		B8	CMP	B	
700A		DA,13,70	JC	Loop(7013)	
700D	loop1	90	SUB	B	Dividend-divisor=>A
700E		0C	INR	C	C=C+1
700F		B8	CMP	B	Is dividend < divisor
7010		D2,0D,70	JNC	Loop1(700D)	If not,go back
7013	loop	32,03,75	STA	Remainder(7503)	Store Remainder
7016		79	MOV	A,C	
7017		32,04,75	STA	Quotient(7504)	Store Quotient
701A		CF	RST 1		Terminate.
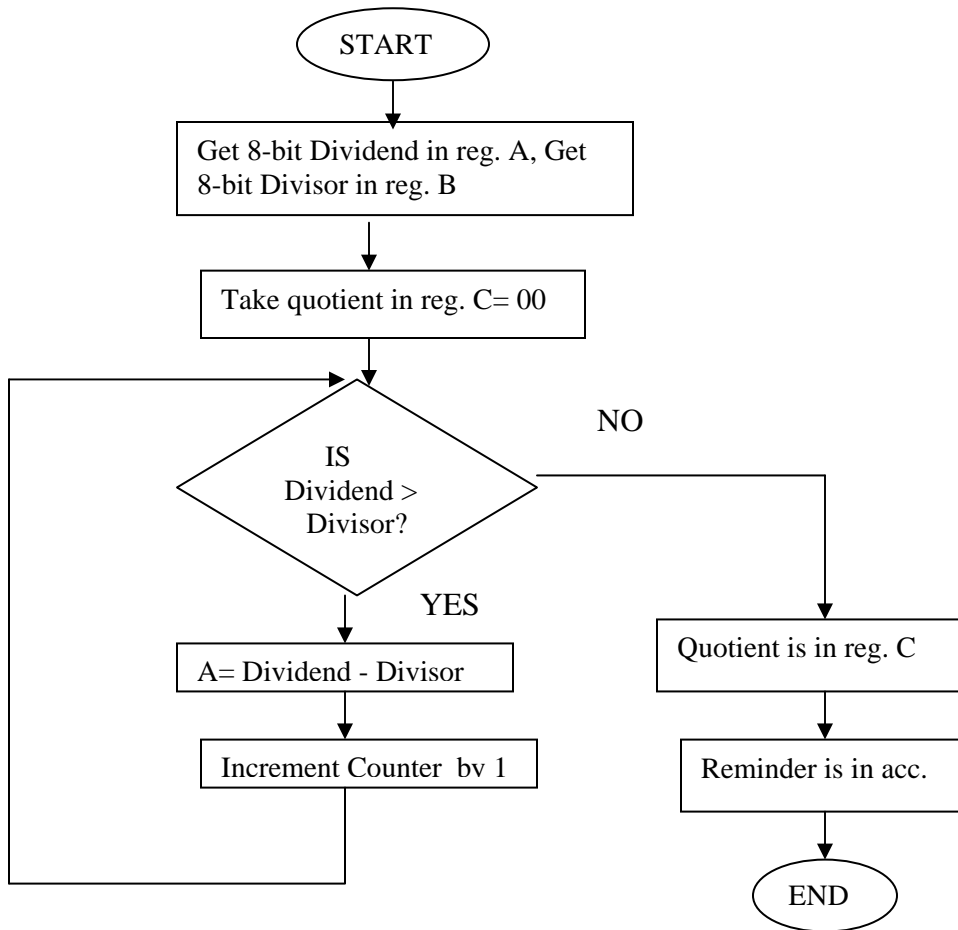
CIRCUIT DIAGRAM / BLOCK DIAGRAM:-

```
                    ( START )
                        |
                        v
        +------------------------------------+
        | Get 8-bit Dividend in reg. A, Get  |
        | 8-bit Divisor in reg. B            |
        +------------------------------------+
                        |
                        v
        +------------------------------------+
        | Take quotient in reg. C= 00        |
        +------------------------------------+
                        |
                        v
                    /         \          NO
                  /    IS       \  ------------------+
                 <   Dividend >   >                  |
                  \   Divisor?  /                    |
                    \         /                      |
                        | YES                        v
                        v                +-------------------------+
        +----------------------------+   | Quotient is in reg. C   |
        | A= Dividend - Divisor      |   +-------------------------+
        +----------------------------+                |
                        |                             v
                        v                +-------------------------+
        +----------------------------+   | Reminder is in acc.     |
        | Increment Counter  bv 1    |   +-------------------------+
        +----------------------------+                |
                (loops back)                          v
                                                  ( END )
```

PROCEDURE:-

ANSHUMAN	SCIENTECH
S	Reset
Enter Enter	Exmem
Program Address	Starting Address
Write Program	Next
Execution Steps	Write Program
Esc	**Execution Steps**
G	Reset
Enter-enter	GO
Prog. Address	Starting Address
Enter	Fill
S	Reset
Enter	Exmem
Any key-2	Result Address
Enter-2	
Register Name	

INPUT DATA

        7501- Divisor
        7502-Dividend

OUTPUT DATA

        7503-Remainder
        7504-Quotient

PRECAUTIONS:-

 Make sure that all the machine codes should be as per specified in the program.

Question & Answer

Q.1 Explain the addressing mode ofSTA addr?
Ans: Direct

Q.2 How many machine cycles are in STA instruction?
Ans: 4

 Q.3 What is LHLD addr?
Ans: load H - L pair direct.

Q.4 How many T-state are in LHLD instruction?
Ans: 24 sixteen T –states..

 Q.5 Explain the addressing mode of LHLD addr?
Ans: Direct

Q.6 How many machine cycles are in LHLD instruction?
Ans: 5

Q.7 What is SHLD addr ?
Ans: store H-L pair direct.

Q.8 How many T-state are in SHLD instruction?
Ans: 16

Q.9 Explain the addressing mode of SHLD addr?
Ans: Direct

Q.10 How many machine cycles are in SHLD instruction?
Ans: 5.

EXPERIMENT NO. 5

AIM : WRITE A PROGRAM USING 8085 FOR FINDING SQUARE-ROOT OF A NUMBER

APPARATUS : 8085 microprocessor kit, 5V power supply, Keyboard.

THEORY(Program):

Memory Address	Label	Machine Code	Mnemonics	Operands	Comments
2000		0E,01	MVI	C,01H	Place 01 in reg.C
2002		06,01	MVI	B,01H	Place odd number 1 in reg.B
2004		3E,36	MVI	A,36	Load accumulator with the given number
2006	Loop	90	SUB	B	Subtract odd number from the accumulator
2007		CA,10,20	JZ	Ahead(2010)	If accumulator contents are zero, go to Ahead
200A		0C	INR	C	Increment reg. C
200B		04	INR	B	Increment odd number
200C		04	INR	B	Increment odd number
200D		C3,06,20	JMP	Loop(2006)	Repeat subtraction
2010	Ahead	79	MOV	A,C	Move the contents of C to A
2011		32,50,20	STA	2050H	Store the result in the memory location 2050H.
2014		CF	RST1		Stop

PROCEDURE:-

	SCIENTECH
ANSHUMAN	Reset
S	Exmem
Enter Enter	Starting Address
Program Address	Next
Write Program	Write Program
Execution Steps	**Execution Steps**
Esc	Reset
G	GO
Enter-enter	Starting Address
Prog. Address	Fill
Enter	Reset

S   Exmem
Enter   Result Address
Any key-2
Enter
Name
Register

CIRCUIT DIAGRAM / BLOCK DIAGRAM:-
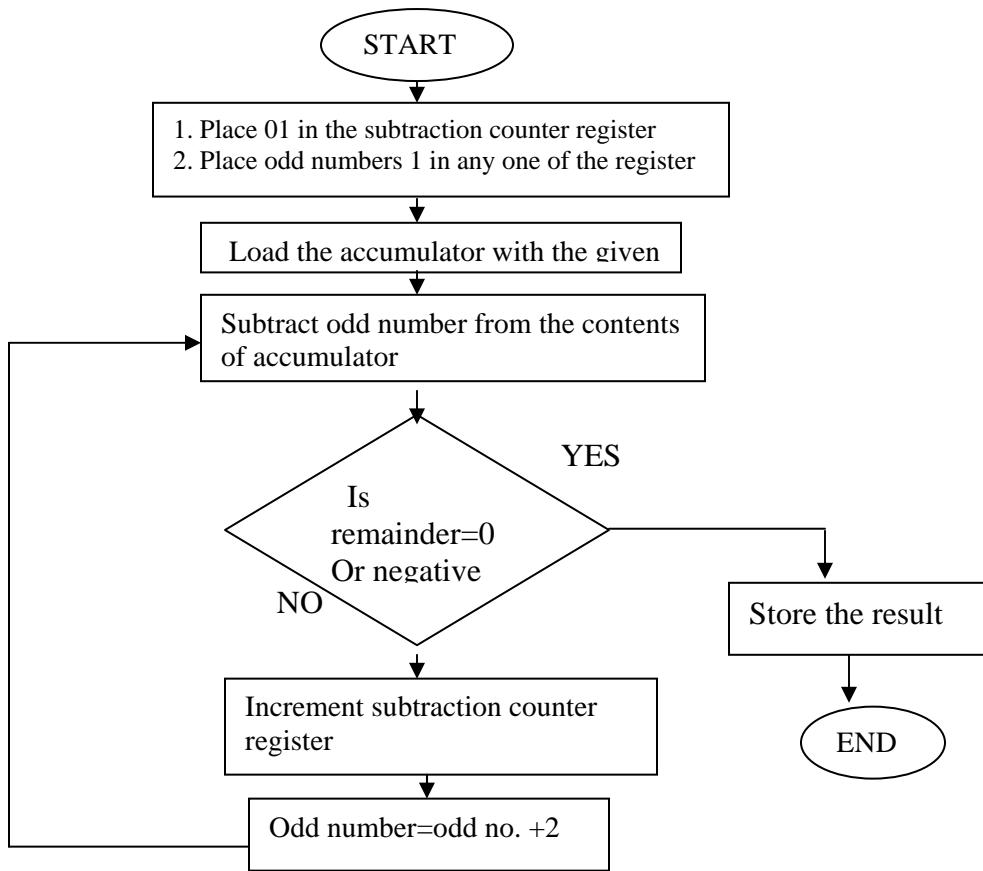
INPUT DATA

        2500-10H
        2501- 00H

OUTPUT DATA

        2550- 04H

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.

Question & Answer ;

Q.1 What is LDAX rp?
Ans: Load accumulator indirect.

Q.2 How many T-state are in LDAX instruction?
Ans: 7

Q.3 Explain the addressing mode ofLDAX rp?
Ans: Register indirect .

Q.4 How many machine cycles are in LDAX instruction?
Ans: 2

Q.5 What is STAX rp ?
Ans: Store accumulator indirect

Q.6 How many T-state are in STAX instruction?
Ans: 7

Q.7 Explain the addressing mode ofSTAX rp?
Ans: Register indirect.

Q.8 How many machine cycles are in STAX instruction?
Ans: 2

Q.9 What is XCHG ?
Ans: Exchange the contents of H-L pair with D-E pair

Q.10 How many T-state are in XCHG instruction?
Ans: 4

EXPERIMENT NO. 6

AIM : WRITE A PROGRAM USING 8086 FOR DIVISION OF A DEFINED DOUBLE WORD
     BYANOTHER WORD & VERIFY.

APPARATUS : 8086 microprocessor kit, 5V power supply, Keyboard.

THEORY(Program)

Memory Address	Machine Code	Mnemonics	Operands	Comments
1000	B8,78,56	MOV	AX,5678H	Move 5678 to AX
1003	BA,34,12	MOV	DX,1234H	Move 1234 to DX
1006	B9,25,25	MOV	CX,2525	Move 2525 to CX
1009	F7,F1	DIV	CX	Divide AX&DX by CX
100b	CD,A5	INT	A5	

CIRCUIT DIAGRAM / BLOCK DIAGRAM:-,

START

Get LSB of Dividend

Get MSB of Dividend

Get Divisor in CX

Divide & Store result

END

PROCEDURE:-
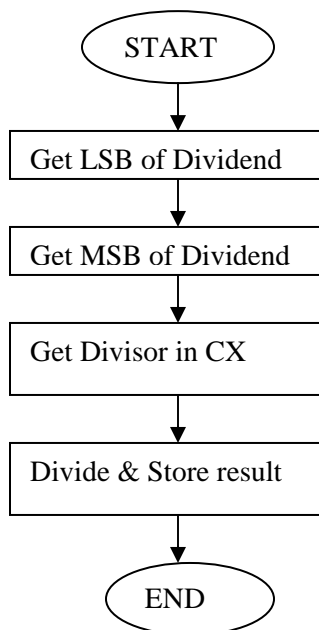
ANSHUMAN	SCIENTECH
	Reset
S	O
Enter Enter	EB/AX
SRC-SEGM Address	
Enter	Starting Address
Program Address	Next
Write Program	Write Program

Execution Steps

Execution Steps

Esc	Reset
G	GO
Enter-enter	Starting Address
SRC-SEGM Add	Fill
Enter	Reset
Prog. Address	O
Enter	EB/AX
S	
Enter	Result Address
Any key-2	
Enter-2	
Register Name	

INPUT DATA
    AX : 5678H
    DX : 1234H
    CX : 2525H

OUTPUT DATA
    AX : 7D77(Quotient)
    DX : 0145(Remainder)

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.

Question & Answer:

Q.l Explain the addressing mode of XCHG?
Ans: Register

Q.2 How many machine cycles are in XCHG instruction?
Ans: 1

Q.3 What is ADD r ?
Ans: Add register to accumulator.

Q.4 How many T-state are in ADD instruction?
Ans: 4

Q.5 Explain the addressing mode of ADD?
Ans: Register

Q.6 How many machine cycles are in ADD instruction?
Ans: 2

Q.7 What is ADC r?
Ans: Add register with carry to accumulator.

Q.8 How many T-state are in ADC r instruction?
Ans: 4

Q.9 Explain the addressing mode of ADC ?
Ans: Register

Q.10 How many machine cycles are in ADC instruction?
Ans: 1

EXPERIMENT NO. 7

AIM : WRITE A PROGRAM USING 8086 FOR COPYING 12 BYTES OF DATA FROM SOURCE TO DESTINATION & VERIFY.

APPARATUS : 8086 microprocessor kit, 5V power supply, Keyboard.

THEORY(Program)

Memory Address	Label	Machine Code	Mnemonics	Operands	Comments
0101		FC	CLD		Clear direction flag DF
0102		BE,00,03	MOV	SI,0300	Source address in SI
0105		BF,02,02	MOV	DI,0202	Destination address in DI
0108		8B,0C	MOV	CX,[SI]	Count in CX
010A		46	INC	SI	Increment SI
010B		46	INC	SI	Increment SI
010C	BACK	A4	MOV	SB	Move byte
010D		E2,FD	LOOP	BACK	Jump to BACK until CX becomes zero
010F		CC	INT		Interrupt program

CIRCUIT DIAGRAM / BLOCK DIAGRAM:-

```
            ┌─────────┐
            │  START  │
            └────┬────┘
                 │
      ┌──────────▼──────────┐
      │  Get source address │
      └──────────┬──────────┘
                 │
      ┌──────────▼────────────┐
      │ Get destination address│
      └──────────┬────────────┘
                 │
      ┌──────────▼──────────┐
      │   Get Count in CX    │
      └──────────┬──────────┘
                 │
      ┌──────────▼──────────┐
      │  Increment in SI     │
      │  Increment in SI     │
      └──────────┬──────────┘
                 │
      ┌──────────▼──────────┐
      │    Move bytes        │◄──────┐
      └──────────┬──────────┘        │
                 │                    │
           ◄─────▼─────►   NO         │
          /   IS         \ ───────────┘
          \  Count =0?   /
           ◄─────┬─────►
                 │ YES
            ┌────▼────┐
            │   END   │
            └─────────┘
```
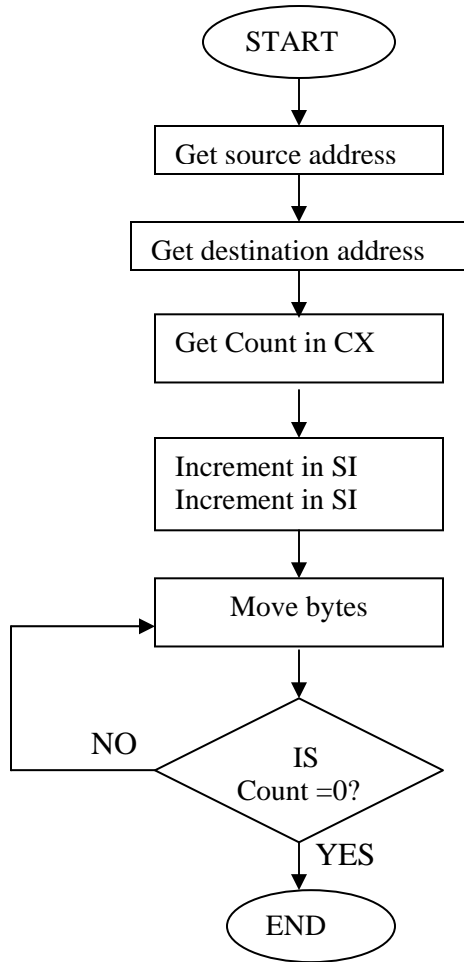
INPUT DATA			
0300	: 0B	030B	: 0A
0301	: 00	030C	: 0B
0302	: 03	030D	: 0E
0303	: 04		
0304	: 05	**OUTPUT DATA**	
0305	: 06	0202	: 03
0306	: 15	0203	: 04
0307	: 07	0204	: 05
0308	: 12	0205	: 06
0309	: 08	0206	: 15
030A	: 09	0207	: 07

0208	: 12	020B	: 0A
0209	: 08	020C	: 0B
020A	: 09	020D	: 0E

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.

Question & Answer:

Q.1 Explain ADI data?
Ans: Add immediate data to accumulator

Q.2 How many T-states are in ADI instruction?
Ans: 7

Q.3 Explain the addressing mode of ADI?
Ans: Immediate

Q.4 How many machine cycles are in ADI instruction?
Ans: 2

Q 5 Explain DAD rp ?
Ans: Add register pair to HL pair.

Q.6 How many T-states are in DAD instruction?
Ans: 10

Q.7 Explain the addressing mode of DAD.
Ans: Register

Q.8 How many machine cycles are in DAD instruction?
Ans: 3

Q.9 Explain DAA.
Ans: Decimal adjust accumulator

Q.10 What is INX rp?
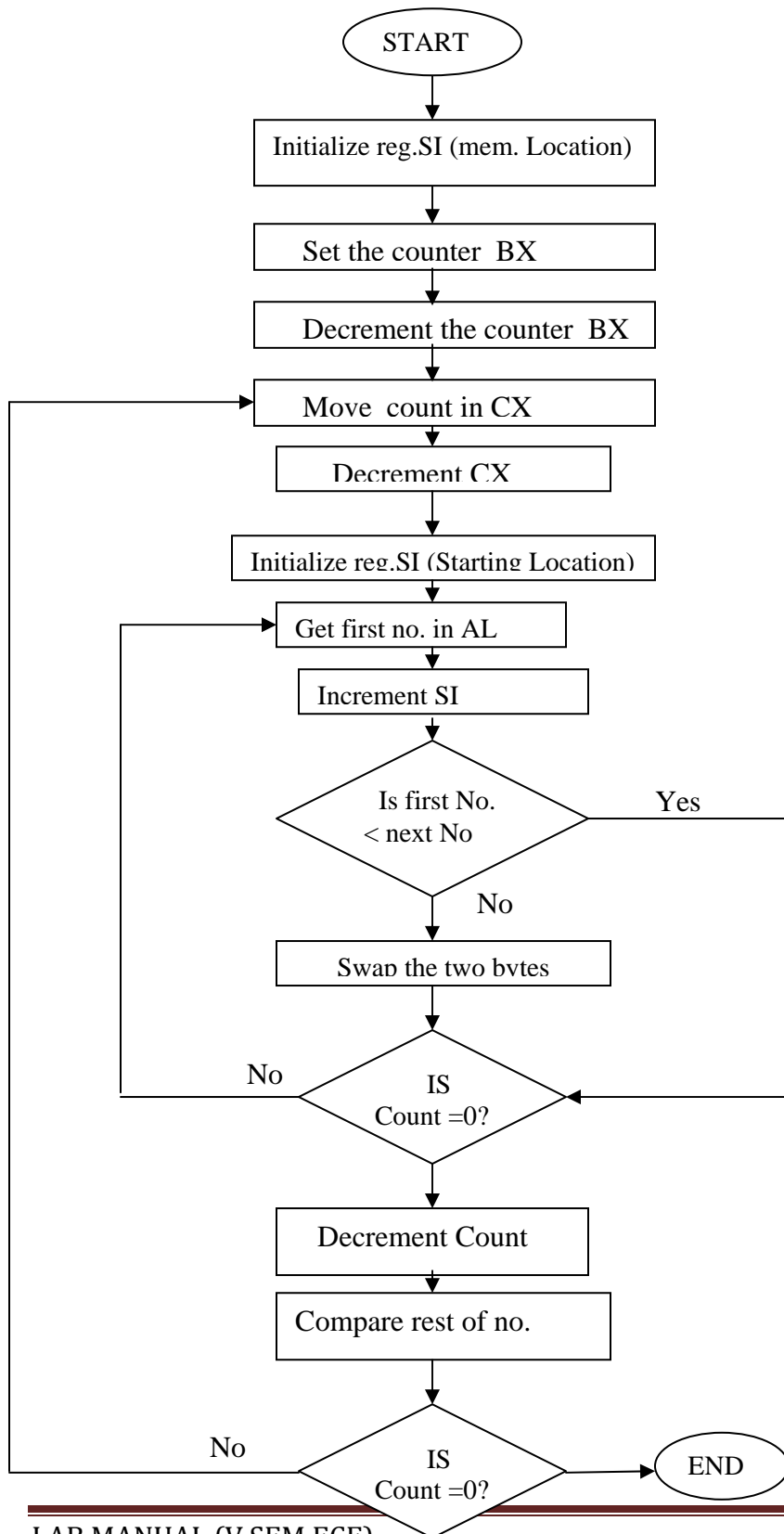Ans: Increment register pair

EXPERIMENT NO.8

AIM : WRITE A PROGRAM USING 8086 FOR ARRANGING AN ARRAY OF NUMBERS IN DESCENDING ORDER & VERIFY.

APPARATUS : 8086 microprocessor kit, 5V power supply, Keyboard.

THEORY(Program)

Memory Address	Label	Machine Code	Mnemonics	Operands	Comments
0200		BE,00,03	MOV	SI,0300	Initialize SI Reg. with Memory Location. 0300.
0203		8B,1C	MOV	BX,[SI]	BX has no. of bytes
0205		4B	DEC	BX	Decrement the no. of bytes by one
0206	(3)	8B 0C	MOV	CX (SI)	Move no. of bytes in CX
0208		49	DEC	CX	Decrement the no. of bytes by one
0209		BE,02,03	MOV	SI,0303	Initialize SI reg. with the starting address of string
020C	(2)	8A,04	MOV	AL,[SI]	Move first data byte of string into AL
020E		46	INC	SI	Point at the next bytes of the string
020F		3A,04	COMP	AL,[SI]	Com. the two bytes of string.
0211		73,06	JAE	(1)	If two bytes are equal or 1st byte is above that the second byte branch to (1)
0213		86,04	XCHG	AL,[SI]	Else
0215		4E	DEC	SI	Second byte is less than first byte and swap the two bytes.
0216		88,04	MOV	[SI],AL	
0218		46	INC	SI	Point at next location of string
0219	(1)	E2,F1	LOOP	(2)	Loop if CX is not zero
021B		4B	DEC	BX	
021C		BE,00,03	MOV	SI,0300	
021F		75,E5	JNZ	(3)	
0221		F4	HLT		Halt.

CIRCUIT DIAGRAM / BLOCK DIAGRAM:-

```
                    ┌──────────┐
                    │  START   │
                    └──────────┘
                         │
          ┌──────────────────────────────────┐
          │ Initialize reg.SI (mem. Location) │
          └──────────────────────────────────┘
                         │
              ┌────────────────────────┐
              │   Set the counter  BX  │
              └────────────────────────┘
                         │
            ┌──────────────────────────────┐
            │  Decrement the counter  BX   │
            └──────────────────────────────┘
                         │
             ┌──────────────────────────┐
             │    Move  count in CX     │
             └──────────────────────────┘
                         │
                ┌──────────────────┐
                │   Decrement CX   │
                └──────────────────┘
                         │
          ┌────────────────────────────────────┐
          │ Initialize reg.SI (Starting Location) │
          └────────────────────────────────────┘
                         │
               ┌──────────────────────┐
               │  Get first no. in AL │
               └──────────────────────┘
                         │
                  ┌──────────────┐
                  │ Increment SI │
                  └──────────────┘
                         │
                      ╱╲
                     ╱    ╲
                    ╱ Is first No. ╲    Yes
                    ╲ < next No    ╱──────────┐
                     ╲          ╱             │
                      ╲╱                      │
                         │ No                 │
               ┌──────────────────────┐       │
               │  Swap the two bytes  │       │
               └──────────────────────┘       │
                         │                    │
       No             ╱╲                      │
     ┌───────────────╱    ╲◄──────────────────┘
     │               ╲ IS Count =0? ╱
     │                ╲          ╱
     │                 ╲╱
     │                  │
     │          ┌──────────────────┐
     │          │  Decrement Count │
     │          └──────────────────┘
     │                  │
     │          ┌──────────────────────┐
     │          │ Compare rest of no.  │
     │          └──────────────────────┘
     │                  │
     │    No          ╱╲
     └───────────────╱    ╲         ┌───────┐
                     ╲ IS Count =0? ╱──────►│  END  │
                      ╲          ╱          └───────┘
                       ╲╱
```

PROCEDURE:-

ANSHUMAN	SCIENTECH
S	Reset
Enter Enter	Exmem
Program Address	Starting Address
Write Program	Next
Execution Steps	Write Program
	Execution Steps

Esc	Reset
G	GO
Enter-enter	Starting Address
Prog. Address	Fill
Enter	Reset
S	Exmem
Enter	Result Address
Any key-2	
Enter-2	
Register Name	

INPUT DATA

0300	: 05
0301	: 00
0302	: 20
0303	: 25
0304	: 28
0305	: 15
0306	: 07

OUTPUT DATA

0302	: 28
0303	: 25
0304	: 20
0305	: 15
0306	: 07

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.

Question & Answer:

1.   What is microprocessor?

Ans   It is a program controlled semi conductor device (IC), which fetches, decodes and execute instructions.

2.   What are the basic units of microprocessor?

Ans   The basic units or blocks of microprocessor are ALU, an array of registers and control unit.

3.   What is a bus?

Ans   Bus is a group of conducting lines that carries data, address and control signals.

4.   Why data bus is bi-directional?

Ans   The microprocessor is to fetch (read) the data from memory or input device for processing and after processing it has to store (write) the data to memory or output devices. Hence the data bus is bi-directional.

5.   Why data bus is bi-directional?

Ans   The address is an identification number used by the microprocessor to identify or access a memory location or input/output device. It is an output signal from the processor. Hence the address bus is unidirectional.

6.   Define machine cycle?

Ans   Machine cycle is defined as the time required to complete one operation of accessing memory input/output, or acknowledging an external request. This cycle may consists of three to six T-states.

7.   Define T-state?

Ans   T-state is defined as one subdivision of operation performed in one clock period. These subdivisions are internal states synchronized with the system clock, and each T-state is precisely equal to one clock period.

8.   What is an instruction cycle?

Ans   The sequence of operations that a processor has to carry out while executing the instruction is called instruction cycle. Each instruction cycle of processor contains a number of machine cycles.

9.   What is fetch and execute cycle?

Ans   The instruction cycle is divided in to fetch and execute cycles. The fetch cycle is executed to fetch the opcode from memory. The execute cycle is executed to decode the instruction and to perform the work instructed by the instruction.

10.   List the flags of 8085?

Ans   There are five flags in 8085.They are sign flag, zero flag, auxiliary carry flag, parity flag and carry flag.

EXPERIMENT NO.9

AIM : WRITE A PROGRAM TO INTERFACE ADC & DAC WITH 8085 & DEMONSTRATE GENERATION OF SQUARE WAVE.
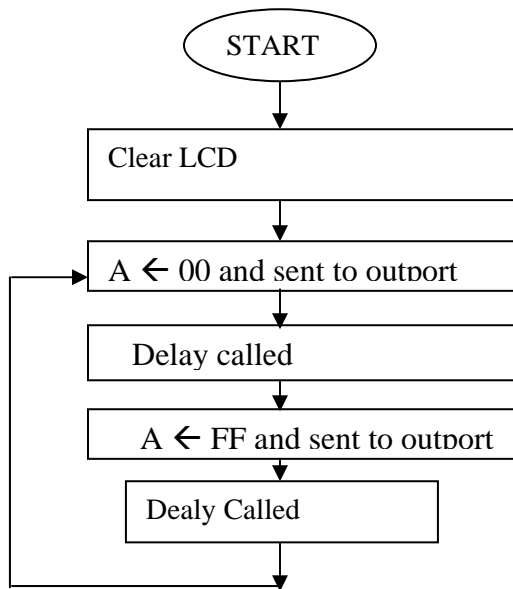
APPARATUS : 8085 microprocessor kit, 5V power supply, Keyboard.

DESCRIPTION: A D/A converter chip DAC 0800 has been provided on the board of M85-07 to enable the user to have analog output. This can be used for generating various waveforms or for any closed loop applications. The chip has been used in I/O mapped mode and has an address of (A0-A7), i.e any of A0 to A7 can be used as an address. This chip has been designed to give an output of 0 to 8 Volts. The output of DAC 0800 is coming at Pin No.13 of connector CN11.

THEORY(Program)

Memory Address	Label	Machine Code	Mnemonics	Operands	Comments
2000		CD 4D 0F	CALL	LECHO	CLEAR LCD DISPLAY
2003		06 0E	MVI	B,0EH	
2005		21 1F 20	LXI	H,WAVE	
2008		CD 47 17	CALL	PRINTF	DISPLAY MESGAE
200B	DAC	3E 00	MVI	A,00H	
200D		D3 38	OUT	38H	
200F		CD 31 0F	CALL	DELAY1	
2012		06 14	MVI	B,14H	
2014		CD 47 17	CALL	PRINTF	DISPLAY MESSAGE
2017		3E FF	MVI	A,FFH	
2019		D3 A0	OUT	0A0H	
201B		CD 31 0F	CALL	DELAY1	
201E		C3 0B 20	JMP	DAC	LOOP
2021		52 41 4D 50 20			WAVE OUTPUT AT
2026		50 49 4E 20 4E			PIN NO. 2 CONN. C9

CIRCUIT DIAGRAM / BLOCK DIAGRAM:-

```
                        ( START )
                            │
                            ▼
              ┌─────────────────────────┐
              │  Clear LCD              │
              └─────────────────────────┘
                            │
                            ▼
              ┌─────────────────────────┐
          ┌──▶│  A ← 00 and sent to outport │
          │   └─────────────────────────┘
          │                 │
          │                 ▼
          │   ┌─────────────────────────┐
          │   │  Delay called           │
          │   └─────────────────────────┘
          │                 │
          │                 ▼
          │   ┌─────────────────────────┐
          │   │  A ← FF and sent to outport │
          │   └─────────────────────────┘
          │                 │
          │                 ▼
          │   ┌─────────────────────────┐
          │   │  Dealy Called           │
          │   └─────────────────────────┘
          │                 │
          └─────────────────┘
```

PROCEDURE:-

ANSHUMAN	SCIENTECH
	Reset
S	Exmem
Enter Enter	Starting Address
Program Address	Next
Write Program	Write Program
Execution Steps	**Execution Steps**
Esc	Reset
G	GO
Enter-enter	Starting Address
Prog. Address	Fill
Enter	Reset
S	Exmem
Enter	Result Address
Any key-2	
Enter-2	
Register Name	

RESULT: Waveform observed on the CRO from Pin No. 2 of connector 9.

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.

Question & Answer:

1.   What is ALE?
Ans   The ALE (Address latch enable) is a signal used to demultiplex the address and data lines using an external latch. It is used to enable the external latch.

2.   Where is the READY signal used?
Ans   READY is an input signal to the processor, used by the memory or input/output devices to get extra time for data transfer or to introduce wait states in the bus cycles.

3.   Give some examples of port devices used in 8085 microprocessor based system?
Ans   The various port devices used in 8085 are 8212,8155,8156,8255,8355,8755.

4.   What is the need for timing diagram?
Ans   The timing diagram provides information regarding the status of various signals, when a machine cycle is executed. The knowledge of timing diagram is essential for system designer to select matched peripheral devices like memories, latches, ports etc from a microprocessor system.

5.   What operation is performed during first T-state of every machine cycle in 8085?
Ans   In 8085, during the first T-state of every machine cycle the low byte address is latched into an external latch using ALE signal.

6.   What is interrupt acknowledge cycle?
Ans   The interrupt acknowledge cycle is a machine cycle executed by 8085 processor to get the address of the interrupt service routine in order to service the interrupt device.

7.   What is vectored and non-vectored interrupt?
Ans   When an interrupt is accepted, if the processor control branches to a specific address defined by the manufacturer then the interrupt is called vectored interrupt. In Non-vectored interrupt there is no specific address for storing the interrupt service routine. Hence the interrupted device should give the address of the interrupt service routine.

8.   List the software and hardware interrupts of 8085?
Ans   Software interrupts : RST 0,RST 1,RST 2,RST 3,RST 4,RST 5,RST 6,RST 7
         Hardware interrupts : TRAP,RST 7.5,RST 6.5,RST 5.5, INTR.

9.   What is TRAP?
Ans   The TRAP is a non-maskable interrupt of 8085. It is not disabled by processor reset or after recognition of interrupt.

10.   How clock signals are generated in 8085 and what is the frequency of the internal clock?
Ans   The 8085 has the clock generation circuit on the chip but an external quartz crystal or LC circuit or RC circuit should be connected at the pins X1 andX2. The maximum internal clock frequency of 8085 is 3.03MHz.

EXPERIMENT No. 10

AIM: - To study the stepper motor and to execute microprocessor computer   based control of the same by changing number of steps, the direction of rotation and speed.

APPARATUS USED:- Stepper Motor Kit, µP Kit, Interface Cord and Connecting Leads.

THEORY:- The stepper motor is a special type of motor which is designed to rotate through a specific angle called step for each electrical pulse received from its control unit. It is used in digitally controlled position control system in open loop mode. The input command is in form of a train of pulses to turn the shaft through a specified angle. the main unit is designed to interface with µP 8085 kit. The stepper motor controller card remains active while the pulse sequence generator disabled as given plug is connected with µp interface socket . Following programme enables the stepper motor to run with µp 8085 kit. For two phase four winding stepper motor only four LSB signals are required.

CIRCUIT DIAGRAM:-
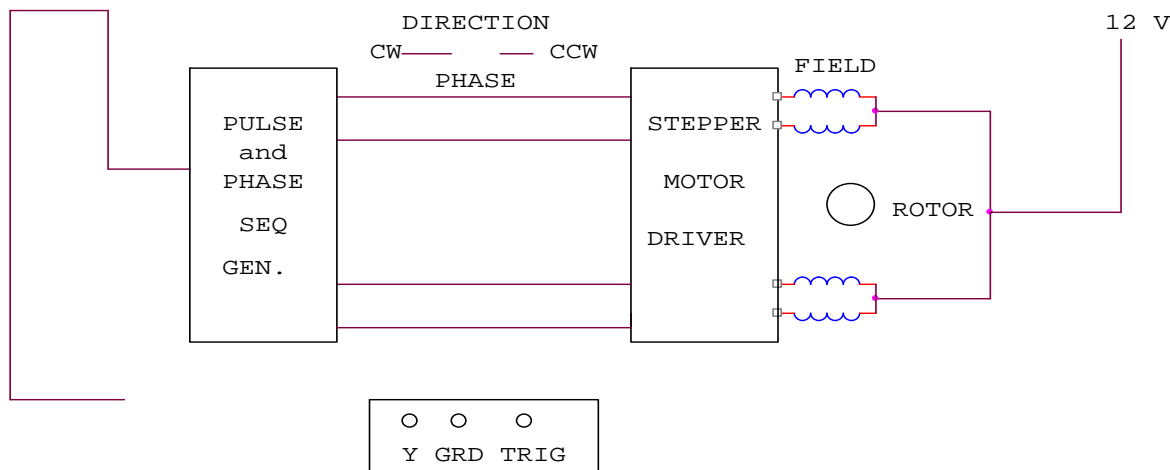
PROCEDURE:-
Connect the stepper motor with µp 8085 kit as shown in fig. press EXMEM key to enter the address as given then press NEXT to enter   data .

ADDRESS   DATA
ADDRESS	DATA			
2000	3E 80	MVI	A,80	Initialize port A as output port.
2002	D3 03	OUT	03	OB
2004	3E F9	Start MVI	AFA	
2006	D3 00	OUT	00	Output code for step o.
2008	CD 3020	call delay		delay between two steps.
200B	3E F5	MVI	A, F6	Location reserve for current Delay
200D	D3 OO	OUT	OO	Output code for step 1.

200F	CD 3020 Call delay	delay between two steps.
2012	3E F6 MVI A, F5	
2014	D3 OO OUT OO	Output code for step 2.
2016	CD 3020 calls delay	between two steps.
2019	3E FA MVI A, F9.	
201B	D3 OO OUT OO	Output code for step 3.
201D	CD 3020 call delay	delay between two steps.
2020	C3 04 20 JMP START	Start.

Press FILL key to store data in memory area. This will complete the pulse sequence generation. To delay programme route, first press EXMEM to start, a dot sign will appear in address field then enter the start address. Press NEXT to enter data.

ADDRESS   DATA

2030	11 00 00 LXI D 00 00	Generates a delay.
2033	CD BC 03 CALL DELAY	
2036	11 00 00 LXI D 00 00	Generates a delay.
2039	CD BC 03 CALL DELAY	
203C	C9 RET	

Press FILL to save data.to execute the programme press the key GO .The above programme is to rotate the motor at a particular as defined by the given address. Changing the following contents will change the motor speed.

ADDRESS   DATA

2030	11 00 20	AND 2036 TO SIMILAR	11 00 20
CHANGE	11 00 10	TO	11 00 10
CHANGE	11 00 05	TO	11 00 05
CHANGE	11 00 03	TO	11 00 03.

The motor direction depends upon codes FA, F6 ,F5 AND F9.Change in following codes will change the motor direction.

ADDRESS	DATA		
2005	3E F9	TO	3E FA
200C	3E F5	TO	3E F6
2012	3E F6	TO	3E F5
2019	3E FA	TO	3E F9.

RESULT:- The stepper motor runs as per fed programme.

PRECAUTION:-
    1. Make the connection of motor with µp kit properly.
    2. Do not change the motor direction at high speed.

Question & Answer:

1.   Define stack?
Ans   Stack is a sequence of RAM memory locations defined by the programmer.

2.   What is program counter? How it is useful in program execution?
Ans   The program counter keeps track of program execution. To execute a program the starting address of the program is loaded in program counter. The PC sends out an address to fetch a byte of instruction from memory and increments its content automatically.

3.   Define opcode and operand?
Ans   Opcode(operation code) is the part of an instruction that identifies a specific operation. Operand is a part of instruction that represents a value on which the instruction acts.

4.   How the 8085 processor differentiates a memory access and I/O access?
Ans   The memory access and I/O access is differentiated using IO/M signal. The 8085 processor asserts IO/M low for memory operation and high for I/O operations.

5.   When the 8085 processor checks for an interrupt?
Ans   In the second T-state of the last machine cycle of every instruction, the 8085 processor checks whether an interrupt request is made or not.

6.   Why interfacing is needed for I/O devices?
Ans   Generally I/O devices are slow devices. Therefore the speed of I/O devices does not match with the speed of microprocessor. And so an interface is provided between system bus and I/O devices.

7.   What is interrupt I/O?
Ans   If the I/O device initiate the data transfer through interrupt then the I/O is called interrupt driven I/O.

8.   What is a port?
Ans   The port is a buffered I/O, which is used to hold the data transmitted from the microprocessor to I/O devices and vice versa.

9.   What is the need for interrupt controller?
Ans   The interrupt controller is employed to expand the interrupt inputs. It can handle the interrupt request from various devices and allow one by one to the processor.

10.   What is synchronous data transfer scheme?
Ans   For synchronous data transfer scheme, the processor does not check the readiness of the device after commands have been issued for read/write operation. For this scheme the processor will request the device to get ready and then read/write to the device immediately after the request.

EXPERIMENT NO. 11

AIM : WRITE A PROGRAM TO CONTROL THE TRAFFIC LIGHT SYSTEM USING 8085 & 8255 PPI.

APPARATUS : 8085 microprocessor kit, 5V power supply, Keyboard.

DESCRIPTION: This Program controls light of one square. By changing the delay between two signals one can change the speed of traffic. 8255 Port Address**.**
Port A- 00H
Port B -01H
Port C- 02H
Control Word 03H

THEORY(Program)

Memory Address	Label	Machine Code	Mnemonics	Operands	Comments
2000		3E 80	MVI	A,80H	Init PA &PB as output
2002		D3 03	OUT	03H	
2004		3E 11	MVI	A,11H	Stop all four ends
2006		D3 00	OUT	00H	
2008		D3 02	OUT	02H	
200A		CD 50 20	CALL	DELAY1	
200D	LOOP	3E 44	MVI	A,44H	GO STR signal of North & South, STOP signal of East &West
200F			OUT	00H	
2011			CALL	DELAY1	
2014			MVI	A,22H	Alert signal for traffic
2016			OUT	00H	
2018			CALL	DELAY2	
201B			MVI	A,99H	GO LEFT signal of North & South
201D			OUT	00H	
201F			CALL	DELAY1	
2022			MVI	A,22H	Alert signal for traffic
2024			OUT	00H	
2026			CALL	DELAY2	
2029			MVI	A,11H	STOP signal of North & South
202B			OUT	00H	
202D			MVI	A,44H	GO STR signal of East & West
202F			OUT	02H	
2031			CALL	DELAY1	
2034			MVI	A,22H	Alert signal for traffic

Memory Address	Label	Machine Code	Mnemonics	Operands	Comments
2036			OUT	02H	
2038			CALL	DELAY2	

Memory Address	Label	Machine Code	Mnemonics	Operands	Comments
203B			MVI	A,99H	GO Left signal of East & West
203D			OUT	02H	
203F			CALL	DELAY1	
2042			MVI	A,22H	Alert signal for traffic
2044			OUT	02H	
2046			CALL	DELAY2	
2049			MVI	A,11H	STOP signal of East &West
204B			OUT	02H	
204D			JMP	LOOP	Jump to loop
2050	DELAY1:		MVI	B,25H	Delay of 10 sec.
2052	LP3:		MVI	C,0FFH	
2054	LP2:		MVI	D, 0FFH	
2056	LP1:		DCR	D	
2057			JNZ	LP1	
205A			DCR	C	
205B			JNZ	LP2	
205E			DCR	B	
205F			JNZ	LP3	
2062			RET		
2063	DELAY2:		MVI	B,05H	Delay of 2 sec
2065	LP6:		MVI	C,0FFH	
2067	LP5:		MVI	D,0FFH	
2069	LP4:		DCR	D	
206A			JNZ	LP4	
206D			DCR	C	
206E			JNZ	LP5	
2071			DCR	B	
2072			JNZ	LP6	
2075			RET		

PROCEDURE:-

	SCIENTECH
ANSHUMAN	Reset
S	Exmem
Enter Enter	Starting Address
Program Address	Next
Write Program	Write Program

Execution Steps	**Execution Steps**
Esc	Reset
G	GO
Enter-enter	Starting Address
Prog. Address	Fill
Enter	Reset
S	Exmem
Enter	Result Address
Any key-2	
Enter-2	
Register Name	

RESULT: Traffic Signal Timing observed for four lane.

PRECAUTIONS:-

Make sure that all the machine codes should be as per specified in the program.

Question & Answer:

1.   What is asynchronous data transfer scheme?
Ans   In asynchronous data transfer scheme, first the processor sends a request to the device for read/write operation. Then the processor keeps on polling the status of the device. Once the device is ready, the processor executes a data transfer instruction to complete the process.

2.   What are the internal devices of 8255?
Ans   The internal devices of 8255 are port-A, port-B, port-C. The ports can be programmed for either input or output function in different operating modes.

3.   What is USART?
Ans   The device which can be programmed to perform Synchronous or Asynchronous serial communication is called USART (Universal Synchronous Asynchronous Receiver Transmitter). Eg: INTEL 8251

4.   What is scanning in keyboard and what is scan time?
Ans   The process of sending a zero to each row of a keyboard matrix and reading the columns for key actuation is called scanning. The scan time is the time taken by the processor to scan all the rows one by one starting from first row and coming back to the first row again.

5.   What is programmable peripheral device?
Ans   If the function performed by the peripheral device can be altered or changed by a program instruction then the peripheral device is called programmable device. It have control register. The device can be programmed by sending control word in the prescribed format to the control register.

6.   What is baud rate?

Ans   The baud rate is the rate at which the serial data are transmitted. Baud rate is defined as (The time
        for a bit cell). In some systems one bit cell has one data bit, then the baud rate and bits/sec are
        same.

7.   What are the tasks involved in keyboard interface?
Ans   The tasks involved in keyboard interfacing are sensing a key actuation, Debouncing the key and
        generating key codes (Decoding the key). These tasks are performed software if the keyboard is
        interfaced through ports and they are performed by hardware if the keyboard is interfaces through
        8279.

8.   How a keyboard matrix is formed in keyboard interface using 8279?
Ans   The return lines, RL0 toRL7 of 8279 are used to form the columns of keyboard matrix. In decoded
        scan lines SL0 t0SL3 of 8279 are used to form the rows of keyboard matrix. In encoded scan
        mode, the output lines of external decoder are used as rows of keyboard matrix.

9.   What is GPIB?
Ans   GPIB is the General Purpose interface Bus. It is used to interface the test instruments to the system
        controller.

10.   Advantages of differential data transfer?
Ans   1. Communication at high data rate in real world environment.
        2. Differential data transmission offers superior performance.
        3. Differential signals can help induced noise signals.