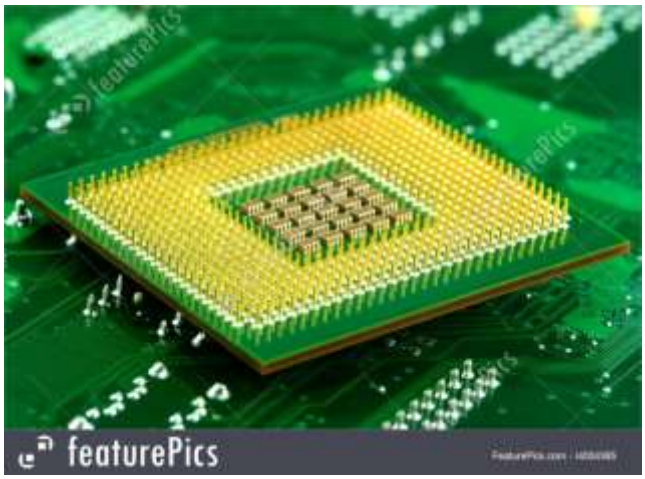
LABORATORY MANUAL
Microprocessor Laboratory



Department of Electrical Engineering
JORHAT ENGINEERING COLLEGE
Assam-785007

EI181513	MICROPROCESSOR LABORATORY	Semester V	L-T-P 0-0-2	1 CREDIT

COURSE OUTCOMES:
After the successful completion of the course student should be able to:
CO1: To define the basic concept of programmable device and structural arrangement of 8085 and its instructions.
CO2: To define the general idea for interfacing memory devices and I/O devices to ensure unique address for each device
CO3: To define the basic concept of PPIs and their operational aspects.
CO4: To apply the knowledge of PPIs for different engineering applications
CO5: To demonstrate the basic evolutionary process leading to the development of microprocessor based system for a few practical applications

Experiment No.	Title of the Experiment	Objective of the Experiment
1	Addition of two 8-bit numbers	To write a assembly language program for adding 2 bit (8) numbers by using-8085 micro-processor kit.
2	Subtraction of two 8 bit numbers.	To write a assembly language program for subtracting 2 bit (8) numbers by using-8085 micro-processor kit.
3	Addition of two 8 bit decimal numbers.	To write a assembly language program to add two 8 bit decimal numbersby using-8085 micro-processor kit.
4	To find the 2's complement of an 8-bit number.	To write a assembly language program to find the 2's compliment of an 8 bit decimal numbers by using-8085 micro-processor kit.
5	To find the larger of the two numbers.	To write a assembly language program to find the larger of the two numbers (04H and 08H) by using-8085 micro-processor kit.
6	To arrange three numbers in descending order.	To write a assembly language program to arrange 3 numbers in descending order by using-8085 micro-processor kit.
7	To find the summation of series of four 8-bit numbers.	To write a assembly language program tofind the summation of series of four 8-bit numbers by using-8085 micro-processor kit.
8	To multiply two 8-bit numbers.	To write a assembly language program tomultiply two8-bit numbers by using-8085 micro-processor kit.
9	To divide 16 bit number by 8 bit number.	To write a assembly language program to divide 16 bit number by8-bit numbers by using-8085 micro-processor kit.

Text book:

- **"**Microprocessor Architecture, Programming and Applications with 8085" by R S Gaonkar,
- "Fundamentals of Microprocessors and Microcontrollers" by B Ram

Student Profile	
Name	
Roll Number	
Department	
Year	

Student Performance		
Sl. No.	**Title of the Experiment**	**Remarks**
1	Addition of two 8-bit numbers	
2	Subtraction of two 8 bit numbers	
3	Addition of two 8 bit decimal numbers.	
4	To find the 2's complement of an 8-bit number	
5	To find the larger of the two numbers.	
6	To arrange three numbers in descending order.	
7	To find the summation of series of four 8-bit numbers	
8	To multiply two 8-bit numbers.	
9	To divide 16 bit number by 8 bit number	

Office Use	
Checked and found …… Grade/ Marks …… Signature ……	

Microprocessor 8085

Fig1: Kit 8085

Fig2: Keys used in writing Assembly Language Programming

Some commonly used Command keys:

S= Substitute()	Examine or write data in memory/IO/Register etc.
R= Serial	Serial monitor mode
G= Go To	Execute
M= Move	Move block to another memory
D= 7 Segment	Set 8 digit 7 segment or LCD display as Console Output
CR= Enter	Used as Increment key to increment the address of the location
Ctrl U	Used as Decrement key to decrement the address location
Esc= Escape	

Introduction to Microprocessor 8085.

Aim

To study the microprocessor 8085

Architecture of 8085 Microprocessor

a) General purpose register

It is an 8 bit register i.e. B,C,D,E,H,L. The combination of 8 bit register is known as register pair, which can hold 16 bit data. The HL pair is used to act as memory pointer is accessible to program.

b) Accumulator

It is an 8 bit register which hold one of the data to be processed by ALU and stored the result of the operation.

c) Program counter (PC)

It is a 16 bit pointer which maintain the address of a byte entered to line stack.

d) Stack pointer (Sp)

It is a 16 bit special purpose register which is used to hold line memory address for line next instruction to be executed.

e) Arithmetic and logical unit

It carries out arithmetic and logical operation by 8 bit address it uses the accumulator content as input the ALU result is stored back into accumulator.

f) Temporary register

It is an 8 bit register associated with ALU hold data, entering an operation, used by the microprocessor and not accessible to programs.

g) Flags

Flag register is a group of fire, individual flip flops line content of line flag register will change after execution of arithmetic and logic operation. The line states flags are

i) Carry flag (C)

ii) Parity flag (P)

iii) Zero flag (Z)

iv) Auxiliary carry flag (AC)

v) Sign flag (S)

h) Timing and control unit

Synchronous all microprocessor, operation with the clock and generator and control signal from it necessary to communicate between controller and peripherals.

i) Instruction register and decoder

Instruction is fetched from line memory and stored in line instruction register decoder the stored information.

j) Register Array

These are used to store 8 bit data during execution of some instruction.

PIN Description

Fig: Pin diagram of

Address Bus

1. The pins A_o – A_{15} denote the address bus.
2. They are used for most significant bit

Address / Data Bus

1. AD_0 – AD_7 constitutes the address / Data bus
2. These pins are used for least significant bit

ALE : (Address Latch Enable)

1. The signal goes high during the first clock cycle and enables the lower order address bits.

IO / M

1. This distinguishes whether the address is for memory or input.
2. When this pins go high, the address is for an I/O device.

S_0 – S_1

S_0 and S_1 are status signal which provides different status and functions.

RD

1. This is an active low signal
2. This signal is used to control READ operation of the microprocessor.

WR

1. WR is also an active low signal
2. Controls the write operation of the microprocessor.

HOLD

1. This indicates if any other device is requesting the use of address and data bus.

HLDA

1. HLDA is the acknowledgement signal for HOLD
2. It indicates whether the hold signal is received or not.

INTR

1. INTE is an interrupt request signal
2. IT can be enabled or disabled by using software

INTA

1. Whenever the microprocessor receives interrupt signal
2. It has to be acknowledged.

RST 5.5, 6.5, 7.5

1. These are nothing but the restart interrupts
2. They insert an internal restart junction automatically.

TRAP

1. Trap is the only non-maskable interrupt
2. It cannot be enabled (or) disabled using program.

RESET IN

1. This pin resets the program counter to 0 to 1 and results interrupt enable and HLDA flip flops.

X_1, X_2

These are the terminals which are connected to external oscillator to produce the necessary and suitable clock operation.

SID

This pin provides serial input data

SOD

This pin provides serial output data

V_{CC} and V_{SS}

1. V_{CC} is +5V supply pin
2. V_{SS} is ground pin

Specifications

1. Processors

Intel 8085 at E144 MHz clock

2. Memory

Monitor RAM: 0000 – IFFF

EPROM Expansion:	2000 –3FFF's
	0000 – FFF
System RAM:	4000 –5FFF
Monitor data area	4100 – 5FFF
RAM Expansion	6000 –BFFF

3. Input / Output

Parallel: A8 TTL input timer with 2 number of 32-55 only input timer available in μ-85 EBI.

Serial: Only one number RS 232-C, Compatible, crucial interface using 8281A

Timer: 3 channel -16 bit programmable units, using 8253 channel '0' used for no band late.

Clock generator. Channel '1' is used for single stopping used program.

Display: 6 digit – 7 segment LED display with filter 4 digit for adder display and 2 digit for data display.

Key board: 21 keys, soft keyboard including common keys and hexa decimal keys.

RES: Reset keys allow to terminateany present activity and retain to μ - 85 its on initialize state.

INT: Maskable interrupt connect to CPU's RST 7.5 interrupt

DEC: Decrement the adder by 1

EXEC: Execute line particular value after selecting address through go command.

NEXT: Increment the address by 1 and then display its content.

In Entering Program into Anshuman Trainer Kitfor mp 8085

1. R ⟶ S ⟶ CR ⟶ CR
2. Enter the Starting address followed by CR t
3. The hexcode of the opcode is entered followed by operand and CR
4. This is continued until the end of the program is reached followed by Halt.
5. Data can be checked by pressing Cntrl-U

How to executive program

Esc ⟶ G(Go To) ⟶ CR ⟶ CRStarting address

How to check contents of registers

SCR ⟶ Any key ⟶ (except CR and Esc) ⟶ CR ⟶ CR ⟶ Register

How to check contents of any address

S ⟶ CR ⟶ Any key (except CR and Esc) ⟶ CR ⟶ CR ⟶ Address

Result:

Thus 8085 microprocessor was studied successfully.

EXPERIMENT 1: ADDITION OF TWO 8-BIT NUMBERS

Aim: To write a assembly language program for adding 2 bit (8) numbers by using-8085 micro-processor kit.

Apparatus required:

8085 microprocessor kit

(0-5V) DC battery

Algorithm:

Step1	:	Start the microprocessor
Step2	:	Intialize the carry as 'Zero'
Step3	:	Load the first 8 bit data into the accumulator
Step4	:	Copy the contents of accumulator to Register B
Step5	:	Load the second 8 bit data into the accumulator.
Step6	:	Add the 2 - 8 bit datas and check for carry.
Step7	:	Jump on if no carry
Step8	:	Increment carry if there is
Step9	:	Store the added request in accumulator
Step10	:	More the carry value to accumulator
Step11	:	Store the carry value in accumulator
Step12	:	Stop the program execution.

Flowchart:

Table:

Address	Label	Mnemonics	Hex Code	Comments

Results:

Input	Calculation	Output

Conclusion:

The assembly language program for 8 bit addition of two numbers was executed successfully by using 8085 micro processing kit.

EXPERIMENT 2: SUBTRACTION OF TWO 8 BIT NUMBERS

Aim: To write a assembly language program for subtracting 2 bit (8) numbers by using-8085 micro-processor kit.

Apparatus required:

8085 microprocessor kit;

(0-5V) DC battery

Algorithm:

Step1	:	Start the microprocessor
Step2	:	Intialize the carry as 'Zero'
Step3	:	Load the first 8 bit data into the accumulator
Step4	:	Copy the contents of contents into the register 'B'
Step5	:	Load the second 8 bit data into the accumulator.
Step6	:	Subtract the 2 8 bit datas and check for borrow.
Step7	:	Jump on if no borrow
Step8	:	Increment borrow if there is
Step9	:	2's compliment of accumulator is found out
Step10	:	Store the result in the accumulator
Step11	:	More the borrow value from 'c' to accumulator
Step12	:	Store the borrow value in the accumulator
Step13	:	Stop program execution

Flowchart:

Table:

Address	Label	Mnemonics	Hex Code	Comments

Results:

Input	Calculation	Output

Conclusion:

The assembly language program for 8 bit addition of two numbers was executed successfully by using 8085 micro processing kit.

EXPERIMENT 3: ADDITION OF TWO 8 BIT DECIMAL NUMBERS.

Aim: To write a assembly language program to add two 8 bit decimal numbersby using-8085 micro-processor kit.

Apparatus required:

8085 microprocessor kit;

(0-5V) DC battery

Algorithm:

Step 1: Start the microprocessor

Step2: Initialize Carry as Zero

Step3: Get the 1st number

Step4: Get the 2nd number

Step5: Add the two numbers

Step6: Perform Decimal adjustment

Step7: Check if carry is generated

Step8: Jump on if no carry

Step9: If carry is generated, increment the carry

Step 10: Store the added request in accumulator

Step 11: Move the carry value to accumulator

Step12: Store the carry value in accumulator

Step 13: Stop the program execution

Flowchart:

Table:

Address	Label	Mnemonics	Hex Code	Comments

Results:

Input	Calculation	Output

Conclusion:

The assembly language program to add two 8-bit decimal numbers was executed successfully by using 8085 micro processing kit.

EXPERIMENT 4: TO FIND THE 2'S COMPLEMENT OF AN 8-BIT NUMBER.

Aim: To write a assembly language program to find the 2's compliment of an 8 bit decimal numbers by using-8085 micro-processor kit.

Apparatus required:

8085 microprocessor kit;

(0-5V) DC battery

Algorithm:

Step 1 : Move 81H to the accumulator

Step 2 : Complement the accumulator content

Step 3 : Increment the content of accumulator by 1

Step 4 : Store the result at memory location C050H

Step 5 : Halt the program

Flowchart:

Table:

Address	Label	Mnemonics	Hex Code	Comments

Results:

Input	Calculation	Output

Conclusion:

The assembly language program to find the 2's compliment of an 8 -bit decimal numbers was executed successfully by using 8085 micro processing kit.

EXPERIMENT 5: TO FIND THE LARGER OF THE TWO NUMBERS.

Aim: To write a assembly language program to find the larger of the two numbers (04H and 08H) by using-8085 micro-processor kit.

Apparatus required:

8085 microprocessor kit;

(0-5V) DC battery

Algorithm:

Step1	:	Move 04H to the accumulator
Step2	:	Load H,L pair with address C000H
Step3	:	Move the content of accumulator to memory
Step4	:	Move 08H to the accumulator
Step5	:	Increment the H, L pair.
Step6	:	Move the constant of accumulator to memory
Step7	:	Load H,L pair with address C000H
Step8	:	Move 1st operand from memory to accumulator
Step9	:	Increment the H,L pair
Step10	:	Move 2nd operand from memory to register B
Step11	:	Compare the value in register B with the value in accumulator
Step12	:	Jump to address C115H if there is no carry
Step13	:	Move largest from register B to accumulator
Step14	:	Store the result in C070H
Step15	:	Halt

Flowchart:

Table:

Address	Label	Mnemonics	Hex Code	Comments

Results:

Input	Calculation	Output

Conclusion:

The assembly language program to find the larger of the two numbers was executed successfully by using 8085 micro processing kit.

EXPERIMENT 6: TO ARRANGE 3 NUMBERS IN DESCENDING ORDER.

Aim: To write a assembly language program to arrange 3 numbers in descending order by using-8085 micro-processor kit.

Apparatus required:

8085 microprocessor kit;

(0-5V) DC battery

Algorithm:

Step1	:	Start the microprocessor
Step2	:	Load the number of values into accumulator and save the number of values in register 'B'
Step3	:	Decrement register 'B' for (N-1) Repetitions
Step4	:	Set 'HL' register pair as data array address pointer and load the data of array in accumulator
Step5	:	Set 'C' register as counter for (N-1) repetitions
Step6	:	Increment 'HL' pair (data address pointer)
Step7	:	Compare the data pointed by 'HL' with accumulator
Step8	:	If the value of accumulator is larger than memory, then jump to step 10, otherwise next step.
Step9	:	Exchange the contents of memory pointed by 'HL' and accumulator
Step10	:	Decrement 'C' register, if the of 'C' is not zero go to step 6, otherwise next step.
Step11	:	Decrement 'B' register, if 'B' is not zero, go step 3, otherwise next step.
Step12	:	Stop the program execution

Flowchart:

Table:

Address	Label	Mnemonics	Hex Code	Comments

Results:

Input	Calculation	Output

Conclusion:

The assembly language program to arrange 3 numbers in descending order was executed successfully by using 8085 micro processing kit.

EXPERIMENT 7: TO FIND THE SUMMATION OF SERIES OF FOUR 8-BIT NUMBERS.

Aim: To write a assembly language program tofind the summation of series of four 8-bit numbers by using-8085 micro-processor kit.

Apparatus required:

8085 microprocessor kit;

(0-5V) DC battery

Algorithm:

Step 1 : Load H-L pair with address 3000H

Step 2 : Move the counter from memory to register C

Step 3 : Initialize accumulator with 00H

Step 4 : Increment H-L pair

Step 5 : Move next number from memory to register B

Step 6 : Add B with A

Step 7 : Decrement counter

Step 8 : Jump to address 2006H if the counter is not zero

Step 9 : Increment H-L pair

Step 10 : Move the result from accumulator to memory

Step 11 : Halt

Flowchart:

Table:

Address	Label	Mnemonics	Hex Code	Comments

Results:

Input	Calculation	Output

Conclusion:

The assembly language program to find the summation of series of four 8-bit numbers was executed successfully by using 8085 micro processing kit.

EXPERIMENT 8: TO MULTIPLY TWO 8-BIT NUMBERS.

Aim: To write a assembly language program tomultiply two8-bit numbers by using-8085 micro-processor kit.

Apparatus required:

8085 microprocessor kit;

(0-5V) DC battery

Algorithm:

Step1	:	Start the microprocessor
Step2	:	Get the 1^{st} 8 bit numbers
Step3	:	Move the 1^{st} 8it number to register 'B'
Step4	:	Get the 2^{nd} 8 bit number
Step5	:	Move the 2^{nd} 8 bit number to register 'C'
Step6	:	Intialise the accumulator as zero
Step7	:	Intialise the carry as zero
Step8	:	Add both register 'B' value as accumulator
Step9	:	Jump on if no carry
Step10	:	Increment carry by 1 if there is
Step11	:	Decrement the 2^{nd} value and repeat from step 8, till the 2^{nd} value becomes zero.
Step12	:	Store the multiplied value in accumulator
Step13	:	Move the carry value to accumulator
Step14	:	Store the carry value in accumulator

Flowchart:

Table:

Address	Label	Mnemonics	Hex Code	Comments

Results:

Input	Calculation	Output

Conclusion:

The assembly language program to find the multiplication of two 8-bit numbers was executed successfully by using 8085 micro processing kit.

EXPERIMENT 9: TO DIVIDE 16 BIT NUMBER BY 8 BIT NUMBER.

Aim: To write a assembly language program to divide 16 bit number by8-bit numbers by using-8085 micro-processor kit.

Apparatus required:

8085 microprocessor kit;

(0-5V) DC battery

Algorithm:

Step1	:	Start the microprocessor
Step2	:	Intialise the Quotient as zero
Step3	:	Load the 1^{st} 8 bit data
Step4	:	Copy the contents of accumulator into register 'B'
Step5	:	Load the 2^{nd} 8 bit data
Step6	:	Compare both the values
Step7	:	Jump if divisor is greater than dividend
Step8	:	Subtract the dividend value by divisor value
Step9	:	Increment Quotient
Step10	:	Jump to step 7, till the dividend becomes zero
Step11	:	Store the result (Quotient) value in accumulator
Step12	:	Move the remainder value to accumulator
Step13	:	Store the result in accumulator
Step14	:	Stop the program execution

Flowchart:

Table:

Address	Label	Mnemonics	Hex Code	Comments

Results:

Input	Calculation	Output

Conclusion:

The assembly language program to find the division of two 8-bit numbers was executed successfully by using 8085 micro processing kit.