MARRI LAXMAN REDDY
INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(AN AUTONOMOUS INSTITUTION)
(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)
Accredited by NBA and NAAC with 'A' Grade & Recognized Under Section2(f) & 12(B)of the UGC act,1956

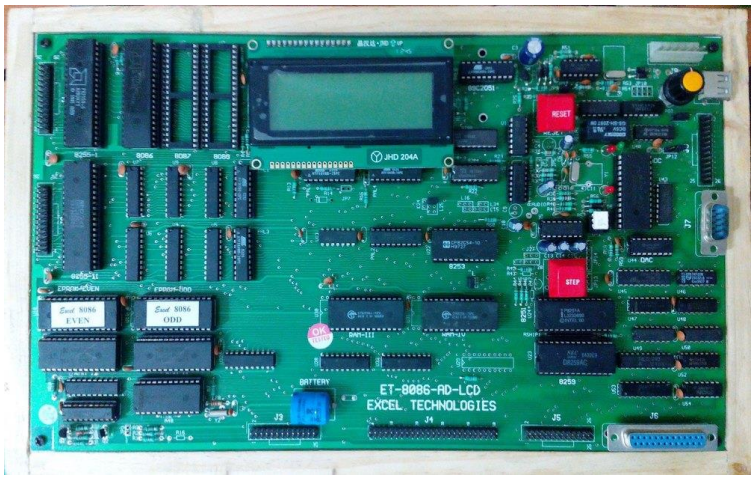
Department of Electronics & Communication Engineering

MICROPROCESSORS AND MICROCONTROLLERS LAB MANUAL

COURSE CODE(2450475)

A.Y: 2026-27

III B. TECH I Semester-ECE&EEE (R24)



Prepared by

Mr.V.KOTESWARA RAO, Asst Professor

Dr.R.RAJA KISHORE, Asst Professor

Mr.M.KRANTHI KUMAR, Asst Professor

VISION AND MISSION OF THE INSTITUTE

INSTITUTE VISION:

To establish as an ideal academic institution in the service of the nation, the world and the humanity by graduating talented engineers to be ethically strong, globally competent by conducting high quality research, developing breakthrough technologies, and disseminating and preserving technical knowledge.

INSTITUTE MISSION:

To fulfill the promised vision through the following strategic characteristics and aspirations:

1. An atmosphere that facilitates personal commitment to the educational success of students in an environment that values diversity and community.
2. Prudent and accountable resource management;
3. Undergraduate programs that integrate global awareness, communication skills and team building across the curriculum;
4. Leadership and service to meet society's needs;
5. Education and research partnerships with colleges, universities, and industries to graduate education and training that prepares students for interdisciplinary engineering research and advanced problem solving;
6. Highly successful alumni who contribute to the profession in the global society.

VISION AND MISSION OF THE DEPARTMENT

DEPARTMENT VISION:

Imparting quality technical education through research, innovation and team work for a lasting technology development in the area of Electronics and Communication Engineering.

MISSION:

To develop a strong centre of excellence for education and research with excellent infrastructure and well qualified faculties to instill in them a passion for knowledge.

To achieve the Mission the department will

1. Establish a unique learning environment to enable the students to face the challenges of the Electronics and Communication Engineering field.
2. Promote the establishment of centre of excellence in niche technology areas to nurture the spirit of innovation and creativity among faculty and students.
3. Provide ethical and value-based education by promoting activities addressing the societal needs.
4. Enable students to develop skills to solve complex technological problems of current times and also provide a framework for promoting collaborative and multidisciplinary activities.

PROGRAMME EDUCATIONAL OBJECTIVES

1. **PEO 1:** have successful **careers in Industry.**
2. **PEO 2:** show excellence in **higher studies/ Research.**
3. **PEO 3:** Show good competency towards **Entrepreneurship.**

PROGRAM SPECIFIC OUTCOMES

PSO1:

1.Analyze response of a circuit diagrams
2.Design of a circuit or system for a given specification

PSO2:

1. Implement opearational block diagrams.
2. Applications of a circuit or system.

Course Outcomes:

After successful completion of the course, Students will be able to:

CO No	Course Outcomes	Knowledge Level (Bloom's Taxonomy)
CO 1	Implement and Debug Complex Operations in Assembly Language for 8086 Microprocessor	**L4 (Analyze)**
CO 2	Apply Interfacing Techniques for External Devices with the 8051 Microcontroller	**L3(Apply)**
CO 3	Analyze and Optimize the Performance of Triangular Wave Generation Using 8051 and DAC.	**L4(Analyze)**
CO 4	Write a program for establishing Serial Communication Using 8051	**L3(Apply)**
CO 5	Create sequence generation using serial communication in 8051	**L2(Create)**

PROGRAM OUTCOMES

a	An ability to apply knowledge of Science, Mathematics, Engineering & Computing fundamentals for the solutions of Complex Engineering problems
b	An ability to identify, formulates, research literature and analyze complex engineering problems using first principles of mathematics and engineering sciences.
c	An ability to design solutions to complex process or program to meet desired needs.
d	Ability to use research-based knowledge and research methods including design of experiments to provide valid conclusions.
e	An ability to use appropriate techniques, skills and tools necessary for computing practice.
f	Ability to apply reasoning informed by the contextual knowledge to assess social issues, consequences & responsibilities relevant to the professional engineering practice.
g	Ability to understand the impact of engineering solutions in a global, economic, environmental, and societal context with sustainability.
h	An understanding of professional, ethical, Social issues and responsibilities.
i	An ability to function as an individual, and as a member or leader in diverse teams and in multidisciplinary settings.
j	An ability to communicate effectively on complex engineering activities within the engineering community.
k	Ability to demonstrate and understanding of the engineering and management principles as a member and leader in a team.
l	Ability to engage in independent and lifelong learning in the context of technological change.

Justifications for CO - PO Mapping:

Course Outcomes (COs)	POs / PSOs	Justification for mapping (Students will be able to)	No. of key competencies
CO 1	PO1	1. Scientific principles and methodology 2. Own and / or other engineering disciplines to integrate / support study of their own engineering discipline.	2
	PO2	1. Problem or opportunity identification 2. Problem statement and system definition 3. Problem formulation and abstraction 4. Information and data collection 5. Solution development or experimentation /Implementation 6. Interpretation of results	6
	PO3	1. Understand customer and user needs and the importance of considerations such as aesthetics. 2. Use creativity to establish innovative solutions; 3. Manage the design process and evaluate outcomes. 4. Knowledge and understanding of commercial and economic context of engineering processes. 5. Knowledge of management techniques which may be used to achieve engineering objectives within that context. 6. Understanding of the requirement for engineering activities to promote sustainable development.	6
	PSO2	1. Implement Operational block diagrams 2. Applications of a circuit or system	2
CO 2	PO2	1. Problem or opportunity identification 2. Problem statement and system definition 3. Problem formulation and abstraction 4. Information and data collection 5.Solutiondevelopmentorexperimentation/Implementation 6. Interpretation of results	6
	PO3	1. Problem statement and system definition 2. Problem formulation and abstraction 3. Information and data collection 4. Model translation 5. Experimental design 6. Solution development or experimentation /Implementation	6
	PO9	1. Self-direction (take a vaguely defined problem and systematically work to resolution). 2. Teams are used during the class room periods, in the hands – on labs, and in the design projects. 3. Instruction on effective teamwork and project management is provided along with an appropriate textbook for reference. 4. Teamwork is important not only for helping the students know their classmates but also in completing assignments. 5. Students also are responsible for evaluating each other'	8

		performance, which is the reflected in the final grade. 6. Ability to work with all levels of people in an organization. 7. Ability to get along with others. 8. Demonstrated ability to work well with a team.	
	PSO1	1. Analyze response of a circuit or system 2. Design of a circuit or system for a given specifications	**2**
	PS02	1. Implement Operational block diagrams 2. Applications of a circuit or system	**2**
CO 3	**PO2**	1. Problem statement and system definition 2. Problem formulation and abstraction 3. Information and data collection 4. Solution development or experimentation /Implementation	**4**
	PO3	1. Investigate and define a problem and identify constraints including environmental and sustainability limitations, health and safety and risk assessment issues; 2. Use creativity to establish innovative solutions; 3. Ensure fitness for purpose for all aspects of the problem including production, operation, maintenance and disposal; 4. Manage the design process and evaluate outcomes. 5. Knowledge of management techniques which may be used to achieve engineering objectives within that context; 6. Understanding of the requirement for engineering activities to promote sustainable development;	**6**
	PO9	1. Self-direction (take a vaguely defined problem and systematically work to resolution). 2. Teams are used during the class room periods, in the hands – on labs, and in the design projects. 3. Instruction on effective teamwork and project management is provided along with an appropriate textbook for reference. 4. Teamwork is important not only for helping the students know their classmates but also in completing assignments. 5. Students also are responsible for evaluating each other' performance, which is the reflected in the final grade. 6. Ability to work with all levels of people in an organization. 7. Ability to get along with others. 8. Demonstrated ability to work well with a team.	**8**
	PSO1	1. Analyze response of a circuit or system 2. Design of a circuit or system for a given specifications	**2**
	PSO2	1. Implement Operational block diagrams 2. Applications of a circuit or system	**2**
	PO2	1. Problem statement and system definition 2. Problem formulation and abstraction	**2**
	PO3	1. Investigate and define a problem and identify constraints including environmental and sustainability limitations, health and safety and risk assessment issues; 2. Knowledge of management techniques which may be used to achieve engineering objectives within that context; 3. Understanding of the requirement for engineering	**4**

		activities to promote sustainable development; 4. Manage the design process and evaluate outcomes.	
CO4	PO4	1. Understanding of engineering principles and the ability to apply them to analyze key engineering processes; 2. Ability to identify, classify and describe the performance of systems and components through the use of analytical methods and modeling techniques; 3. Ability to apply quantitative methods and computer software relevant to their engineering discipline, in order to solve engineering problems; 4. Understanding of and ability to apply a systems approach to engineering problems. 5. Understanding of engineering principles and the ability to apply them to analyze key engineering processes	**5**
	PO9	1. Self-direction (take a vaguely defined problem and systematically work to resolution). 2. Teams are used during the class room periods, in the hands – on labs, and in the design projects. 3. Instruction on effective teamwork and project management is provided along with an appropriate textbook for reference. 4. Teamwork is important not only for helping the students know their classmates but also in completing assignments. 5. Students also are responsible for evaluating each other' performance, which is the reflected in the final grade. 6. Ability to work with all levels of people in an organization. 7. Ability to get along with others. 8. Demonstrated ability to work well with a team.	**8**
	PSO1	1. Analyze response of a circuit or system 2. Design of a circuit or system for a given specifications	**2**
	PSO2	1. Implement Operational block diagrams 2. Applications of a circuit or system	**2**
CO 5	PO1	1.Scientific principles and methodology 2.Mathematical principles	**2**
	PO4	1.Computersoftware/simulationpackages/diagnosticequipment/technicallibraryresources/ literature search tools	**1**
	PO9	1. Self-direction (take a vaguely defined problem and systematically work to resolution). 2. Teams are used during the class room periods, in the hands – on labs, and in the design projects. 3. Instruction on effective teamwork and project management is provided along with an appropriate textbook for reference. 4. Teamwork is important not only for helping the students know their classmates but also in completing assignments. 5. Students also are responsible for evaluating each other' performance, which is the reflected in the final grade. 6. Ability to work with all levels of people in an organization. 7. Ability to get along with others. 8. Demonstrated ability to work well with a team.	**8**
	PSO1	1. Analyze response of a circuit or system 2. Design of a circuit or system for a given specifications	**2**

	PSO2	1. Implement Operational block diagrams 2. Applications of a circuit or system	**2**

Percentage of Key Competencies for CO – PO Mapping:

Course Outcomes(COs)	Program Outcomes (POs) / Number of Vital Features												Program Specific Outcomes (PSOs) / Number of Vital Features	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	3	**10**	**10**	**11**	**1**	**5**	**3**	**3**	**12**	**5**	**12**	**8**	**2**	**2**
CO 1	66	60	60											100
CO 2		60	60						66				100	100
CO 3		40	60						66				100	100
CO 4			40	45					66				100	100
CO 5	66			73					66				100	100

Course Articulation Matrix (CO - PO / PSO Mapping):

COs and POs and COs and PSOs on the scale of 0 to 3, **0** being **no correlation**, **1** being the **low correlation**, **2** being **medium correlation** and **3** being **high correlation**.

0 – **0**≤ C ≤ 5% –No correlation; **2** − 40 % < C < 60% –Moderate.

1 – 5 < C ≤ 40% – Low/ Slight; **3** – 60% ≤ C < 100-% – Substantial /High

Course Outcomes(COs)	Program Outcomes (POs)												Program Specific Outcomes (PSOs)	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO 1	3	3	3											3
CO 2		3	3						3				3	3
CO 3		2	3						3				3	3
CO 4			2	2					3				3	3
CO 5	3			3					3				3	3
TOTAL	**6**	**8**	**11**	**7**					15				**15**	**18**
AVERAGE	**3**	**2.6**	**2.7**	**3.5**					**3.7**				**3.7**	**3.6**

MARRI LAXMAN REDDY INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(AUTONOMOUS)

2450475: MICROPROCESSORS & MICROCONTROLLERS LABORATORY

III Year B.Tech. ECE I – Sem.

L	T	P	C
0	0	2	1

Pre-requisites: Basic concepts of microprocessors and microcontrollers

Course Objectives:

The students will try to learn

- Arithmetic and string operations on 16 bit and 32-bit data
- Sorting and searching operation an array for 8086
- Bit level logical operations, rotate, shift, swap and branch operations
- The interfacing of 8051
- Communication between 8051 to interfacing devices

Course Outcomes:

After successful completion of the course, students shall be able to

- Implement and Debug Complex Operations in Assembly Language for 8086 Microprocessor
- Apply Interfacing Techniques for External Devices with the 8051 Microcontroller
- Analyze and Optimize the Performance of Triangular Wave Generation Using 8051 and DAC.
- Write a program for establishing Serial Communication Using 8051
- Create Sequence Generation Using Serial Communication in 8051

List of Experiments:

The following experiments are performed using 8086 Processor Kits and/or Assembler

1. Write a program for 16-bit arithmetic operations for 8086 (using Various Addressing Modes).
2. Write a program for sorting an array for 8086.
3. Write a program for searching for a number or character in a string for 8086.
4. Write a program for string manipulations for 8086.
5. Write a program for rotate, shift and branch instruction for 8086.
6. Parallel communication between two microprocessors.

The following experiments are performed using 8051 Processor Kits and interfacing Kits

7. Write a program using arithmetic, logical and bit manipulation instructions of 8051.
8. Perform interfacing ADC to 8051.
9. Generate Triangular wave through DAC interfacing with 8051.
10. Program and verify interrupt handling in 8051.
11. Perform Time delay Generation Using Timers of 8051.
12. Perform interfacing to 8086 and programming to control stepper motor.
13. Perform interfacing matrix/keyboard to 8051.

NOTE: Minimum of 12 experiments to be conducted

2450475: MICROPROCESSORS & MICROCONTROLLERS LABORATORY

III Year B.Tech EEE I-Sem	L	T	P	C
	0	0	2	1

Pre-requisites: Basic concepts of microprocessors and microcontrollers

Course Objectives:

The students will try to learn

- Arithmetic and string operations on 16 bit and 32-bit data
- Sorting and searching operation an array for 8086
- Bit level logical operations, rotate, shift, swap and branch operations
- The interfacing of 8051
- Communication between 8051 to interfacing devices

Course Outcomes:

After successful completion of the course, students shall be able to

- Implement and Debug Complex Operations in Assembly Language for 8086 Microprocessor
- Apply Interfacing Techniques for External Devices with the 8051 Microcontroller
- Analyze and Optimize the Performance of Triangular Wave Generation Using 8051 and DAC.
- Write a program for establishing Serial Communication Using 8051
- Create Sequence Generation Using Serial Communication in 8051

List of Experiments:

The following experiments are performed using 8086 Processor Kits and/or Assembler

1.Write a program for 16-bit arithmetic operations for 8086 (using Various Addressing Modes).

2.Write a program for sorting an array for 8086.

3.Write a program for searching for a number or character in a string for 8086.

4.Write a program for string manipulations for 8086.

5.Write a program for rotate, shift and branch instruction for 8086.

6.Parallel communication between two microprocessors.

The following experiments are performed using 8051 Processor Kits and interfacing Kits

7.Write a program using arithmetic, logical and bit manipulation instructions of 8051.

8.Perform interfacing ADC to 8051.

9.Generate Triangular wave through DAC interfacing with 8051.

10.Program and verify interrupt handling in 8051.

11.Perform Time delay Generation Using Timers of 8051.

12.Perform interfacing to 8086 and programming to control stepper motor.

13.Perform interfacing matrix/keyboard to 8051.

NOTE: Minimum of 12 experiments to be conducted

Index

Cycle -01

Cycle -02

History of Microprocessor:

A **microprocessor** is a computer processor which incorporates the functions of a computer's central processing unit (CPU) on a single integrated circuit (IC), or at most a few integrated circuits. The microprocessor is a multipurpose, clock driven, register based, programmable electronic device which accepts digital or binary data as input, processes it according to instructions stored in its memory, and provides results as output. Microprocessors contain both combinational logic and sequential digital logic. Microprocessors operate on numbers and symbols represented in the binary numeral system.

The integration of a whole CPU onto a single chip or on a few chips greatly reduced the cost of processing power. Integrated circuit processors are produced in large numbers by highly automated processes resulting in a low per unit cost. Single-chip processors increase reliability as there are many fewer electrical connections to fail. As microprocessor designs get faster, the cost of manufacturing a chip (with smaller components built on a semiconductor chip the same size) generally stays the same.

Before microprocessors, small computers had been built using racks of circuit boards with many medium- and small-scale integrated circuits. Microprocessors combined this into one or a few large-scale ICs. Continued increases in microprocessor capacity have since rendered other forms of computers almost completely obsolete (see history of computing hardware), with one or more microprocessors used in everything from the smallest embedded systems and handheld devices to the largest mainframes and supercomputers.

- 8-bit Microprocessor:
    1. 8008, 8080,8085
- 16-bit Microprocessor:
    1. 8086,8088,80186, 80188,80286
- 32-bit Microprocessor:
    1. 80386DX, 80386SX, 80376, 80386SL,80386EX
    2. 80486DX,80486SX,80486DX2,80486SL, 80486DX4
- 64-bit Microprocessor:
    1. Pentium Pro, PentiumII, Celeron (Pentium II-based), PentiumIII, Pentium II and III Xeon, Celeron M, Intel Core, Dual-Core Xeon LV
    2. Intel Core 2,Intel Pentium Dual-Core, Celeron, Intel Pentium, Core i3,Core i5, Core i7

History of Microcontroller:

A **microcontroller** (or **MCU**, short for microcontroller unit) is a small computer (SoC) on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. Program memory in the form of Ferroelectric RAM, NOR flash or OTP ROM is also often included on chip, as well as a typically small amount of RAM. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications consisting of various discrete chips.

Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools, toys and other embedded systems. By reducing the size and cost compared to a design that uses a separate microprocessor, memory, and input/output devices, microcontrollers make it economical to digitally control even more devices and processes. Mixed signal microcontrollers are common, integrating analog components needed to control non-digital electronic systems.

Some microcontrollers may use four-bit words and operate at frequencies as low as 4 kHz, for low power consumption (single-digit milliwatts or microwatts). They will generally have the ability to retain functionality while waiting for an event such as a button press or other interrupt; power consumption while sleeping (CPU clock and most peripherals off) may be just nanowatts, making many of them well suited for long lasting battery applications. Other microcontrollers may serve performance-critical roles, where they may need to act more like a digital signal processor (DSP), with higher clock speeds and power consumption.

- 8-bit Microcontroller:
  - Freescale S08
  - 68HC05 (CPU05)
  - 68HC08 (CPU08)
  - 68HC11 (CPU11)
- 16-bit Microcontroller:
  - Freescale S12
  - 68HC12 (CPU12)
  - 68HC16 (CPU16)
  - Freescale DSP56800 (DSPcontroller)
- 32-bit Microcontroller:
  - Freescale Kinetis (ARM architecture)
  - M·CORE
  - MPC500
  - MPC 860 (PowerQUICC)
  - MPC 8240/8250 (PowerQUICC II)
  - MPC 8540/8555/8560 (PowerQUICC III)

INTERNAL ARCHITECTURE OF 8086

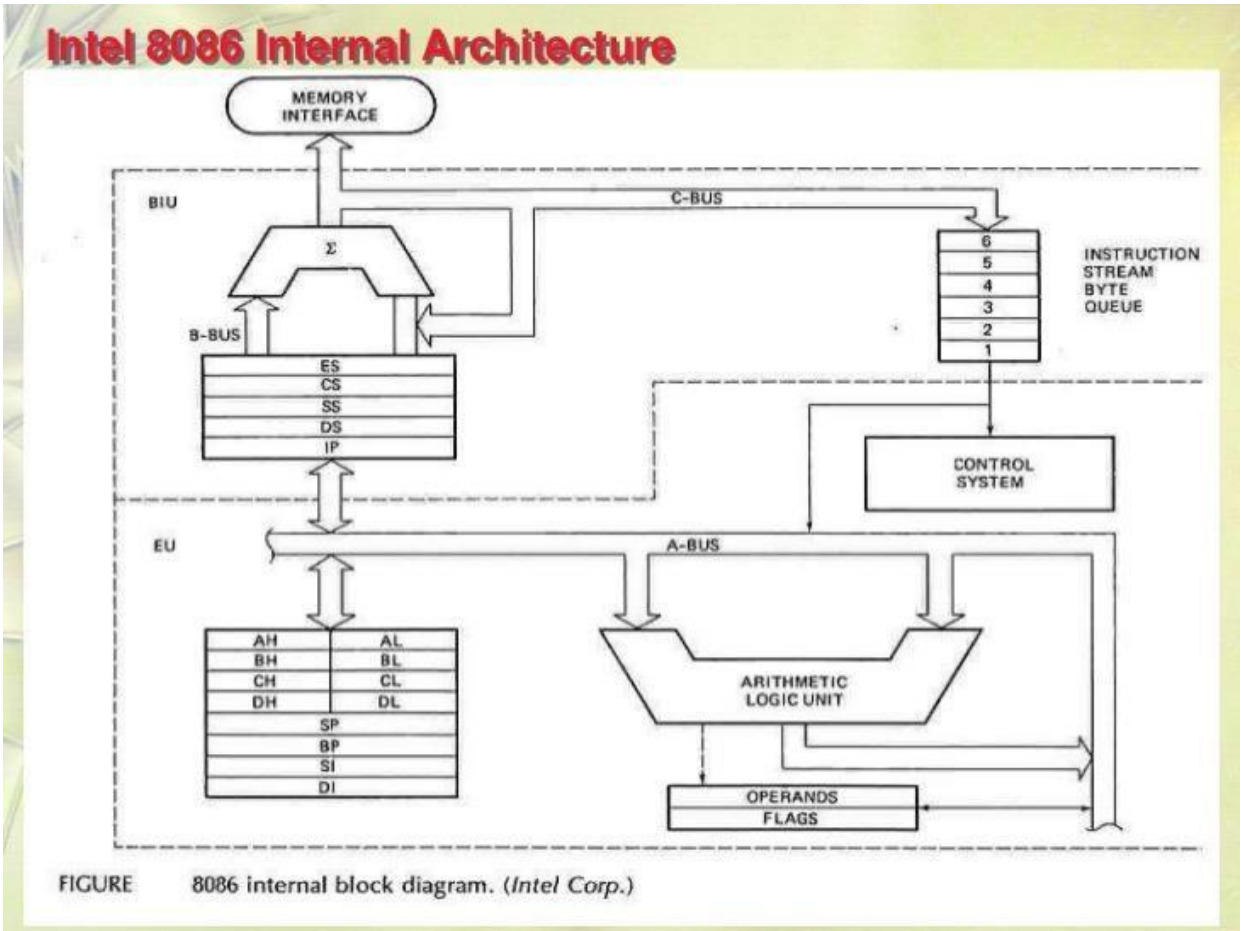
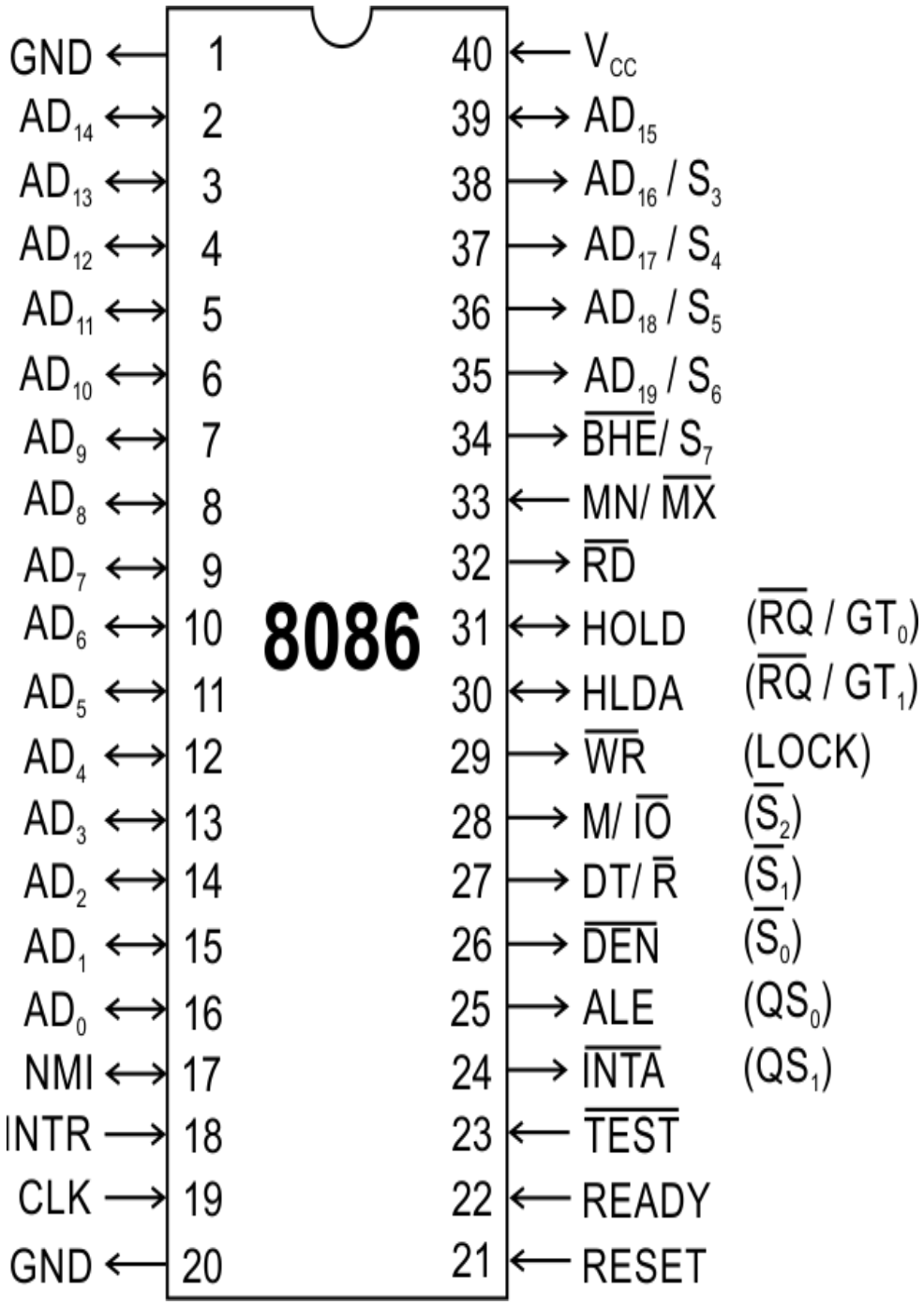


FIGURE 8086 internal block diagram. (*Intel Corp.*)

PIN DIAGRAM OF 8086:



INTRODUCTION TO MASM /TASM

MASM: (Microsoft assembler)

To Create Source File: An editor is a program which allows you to create a file containing the assembly language statements for your program. This file is called a **source file.** Command to create a source file

C:\MASM\BIN> Edit filename. asm

The next step is to process the source file with an assembler. When you run the assembler, it reads the source file of your program. On the first pass through the source program, the assembler determines the displacement of named data items, the offset labels, etc. and puts this information in a symbol table. On the second pass through the source program the assembler produces the binary code for each instruction and inserts the offsets, etc. that it calculated during first pass.

C:\MASM\BIN >Masm filename. asm X, Z

With this command assembler generates three files.

1. The first file (X) called the object file, is given the extension .OBJ

The object file contains the binary codes for the instructions and information about the addresses of the instructions.

2. The third file (Z) generated by this assembler is called the cross-reference file and is given the extension.

3. CRF. The cross-reference file lists all labels and pertinent information required for cross – referencing.

NOTE: The Assembler only finds syntax errors

It will not tell you whether program does what it is supposed to do. To determine whether your program works, you have to run the program and test it. Next step is to process the object file with linker.

C:\MASM\BIN>LINK filename. obj

Run File [Filename1.exe]: "filename1.exe"

List file[nul.map]: NUL

Libraries [.lib]: library name

Definitions File [nul.def]:

Creation of Library: Refer Modular Programming Section

A Linker is a program used to join several object files into one layer object file.

NOTE: On IBM PC – type Computers, you must run the LINK program on your .OBJ file even if it contains only one assembly module. The linker produces a link file with the .EXE extension (an execution file) Next Run **C:\MASM\BIN> filename**

Features of the ALS-SDA-86 8086 MEL Microprocessor Trainer

- 8086 CPU operating at 5 MHz MAX mode.
- Provision for on-board 8087 coprocessor.
- Provision for 256 KB of EPROM & 256 KB of RAM onboard
- Battery backup facility for RAM.
- 48 programmable I/O lines using two 8255's
- Three 16 bit timers using 8253A
- Priority Interrupt Controller (PIC) for eight input using 8259A
- Computer compatible Keyboard.
- Display is 16 x 2 line LCD.
- Designed & engineered to integrate user's application specific interface conveniently at a minimum cost.
- Powerful & user-friendly keyboard / serial monitor, support in development of application programs.
- Software support for development of programs on Computer, the RS-232C interface cable connecting to computer from the kit facilitates transfer of files between the trainer kit & computer for development & debugging purposes.
- High quality reliable PCB with solder mask on both sides & clear legend prints with maximum details provided for the user.

SPECIFICATIONS:

- **CPU** : Intel 8086 operating at 5Mhz in MAX mode.
- **MEMORY** : Total 256KB of memory is in the Kit provided.
  - EPROM : 2 JEDEC compatible sockets for EPROM.
  - RAM : 2 JEDEC compatible sockets for RAM.
- **PARALLEL I/O** : 48 I/O lines using two 8255's.
- **SERIAL I/O** : One RS-232C compatible interface Using USART 8251A**.**
- **TIMER** : Three 16 bit counter / timers 8253A Counter 1 is used for serial I/O Baud rate generation.
- **PIC** : Programmable Interrupt controller Using 8253A provides interrupt Vectors for 8 jumpers selectable Internal /External sources.
- **KEYBOARD / DISPLAY**
  - Keyboard : Computer keyboard can be hocked on to the trainer.
  - Display : LCD 2x16 display.
- **INTERRUPTS**
  - NIM : Provision for connecting NMI to a key switch
  - INTR : Programmable Interrupt controller using 8259A provides Interrupt Vectors for 8 jumper selectable Internal / External Sources.
- **INTERFACE BUS SIGNALS**
  - CPU BUS : All address, data & control lines are TTL compatible & are terminated in berg strip header.
  - PARALLEL I/O: All signals are TTL compatible & Terminated in berg strip header For PPI expansion.
  - SERIAL I/O: Serial port signals are terminated in Standard 9-pin 'D' type connector.
- **MONITOR SOFTWARE**

128KB of serial / Keyboard monitor with Powerful commands to enter verify and Debug user programs, including onboard Assemble and disassemble commands.

- **COMPUTER INTERFACE**

This can be interfaced to host computer System through the main serial port, also

Facilitates uploading, downloading of Intel Hex files between computer and the trainer.

- **POWER REQUIREMENTS**

+5V DC with 2.5 Amps current rating (Max).

OPERATING CONFIGURATION

Two different modes of operation trainer are possible. They are

(i) Serial operation

(ii) Keyboard operation

The first configuration requires a computer system with an RS-232C port, can be used as the controlling device. When a computer system is interfaced to trainer, the driver program must be resident in the computer system. The second mode of operation is achieved through Onboard KEYBOARD / DISPLAY. In this mode, the trainer kit interacts with the user through a computer keyboard and 16x2 LCD Display. This configuration eliminates the need for a computer and offers a convenient way for using the trainer as a stand – alone system.

LIST OF EQUIPMENTS

1. ALS-SDA-86 8086 MEL Microprocessor Trainer.
2. ALS-SDA-51 8051 MEL Microcontroller Trainer.
3. ALS-NIFC-01 Single Stepper Motor Interface.
4. ALS-NIFC-12 LCD Interface.
5. ALS-NIFC-09 Keyboard Display Interface.
6. ALS-NIFC-53 DC Motor Controller Interface.
7. ALS-NIFC-07 ADC Interface.
8. ALS-NIFC-06 DAC Interface.
9. 8086 Communication Package.
10. 8051 Communication Package.

ACCESSORIES

1. Power Supply for Stepper Motor.
2. 26 Core Cables.
3. 50 Core Cables.
4. UP Power Supplies for 8086 &8051.
5. Keyboards for 8086 & 8051.
6. Motor.
7. 4 Way Power Cables.
8. 4 Way Relimate Cables.

EXECUTION PROCEDURE FOR 8086 (for registers)

i) Writing a alp program into processor:

Switch On Power Supply

Press A 2times

SG 0 press enter

DA starting address

Press enter

N

Then display shows @ here you have to write 1st mnemonic

Press enter

N (display displays address opcode mnemonic)

N

Then display shows @ here you have to write 2nd mnemonic

Press enter

N

N

-

-

Then display shows @ up to last mnemonic

Press enter

N

N

! Press enter

EX press enter

ii) Execution of program (for registers):

G enter starting address

Press enter

iii) Verify the result (for registers): press R

Press E.

EXECUTION PROCEDURE FOR 8086 (for memory locations)

i) Writing a alp program into processor:

Switch On Power Supply

Press A

SG 0

Press enter

DA starting address

Press enter

N

Then display shows @ here you have to write 1st mnemonic

Press enter

N (display displays address opcode mnemonic)

N

Then display shows @ here you have to write 2nd mnemonic

Press enter

N

N

-

-

-

Then display shows @ up to last mnemonic

Press enter

N

N

! Press enter

EX press enter

E (exam byte)

Here you have to type SI address, give 1st data, 2nd data, ---- , nth data,

Press enter

ii) Execution of program (for memory locations):

G enter starting address

Press enter

iii) Verify the result (for memory locations): E

Then give DI address press, then display shows the result of 1st 8 bit data

For 2nd 8 bit data again press,------- , nth data

Cycle 1
Using 8086 Kit/Assembler

Exp No.1:

Programs for 16 -bit arithmetic operations for 8086 (using Various Addressing Modes).

1. Addition:

a. AIM: - To write an assembly language program for Addition of two 16-bit numbers (register mode).

APPARATUS: 1.8086 microprocessor kit/MASM ------ 1

2. RPS (+5V)---------- 1

PROGRAM:

i) By using MASM:

```
        Assume cs: code
        Code segment
Start:  MOV AX, 4343H
        MOV BX, 1111H
        ADD AX, BX
        INT 3
        Code ends
        End start
```

ii) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC

4000			MOV AX,4343
			MOV BX,1111
			ADD AX, BX
			INT 3

OUTPUT:

Input		output	
Register	**Data**	**Register**	**Data**
AX	4343	**AX**	**5454**
BX	1111		

b. Aim: 16-Bit Addition in Location mode using 8086 Microprocessor Kit (location mode).

Apparatus:

1. 8086 Microprocessor Trainer Kit.
2. Compatible Keyboard.
3. µP Power Supply.

Program:

i) By using MASM:

```
        Assume cs: code
        CODE segment
Start:  MOV SI, 2000h
        MOV AX, [SI]
        ADD SI, 02
        MOV BX, [SI]
        ADD AX, BX
        MOVDI,3000h
        MOV [DI], AX
        INT 03
        Code ends

        End start
```

ii) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC

4000			MOV SI,2000
			MOV AX,[SI]
			ADD SI,02
			MOV BX,[SI]
			ADD AX,BX
			MOV DI,3000
			MOV [DI],AX
			INT 03

OUTPUT:

Output			
MEMORY LOCATION	**Data**	**MEMORY LOCATION**	**Data**
2000	12	3000	68
2001	34	3001	AC
2002	56		
2003	78		

2. Subtraction:

a. AIM: - To write an assembly language program for subtraction of two 16-bit numbers (register mode).

APPARATUS: 1.8086 microprocessor kit/MASM-------1

2. RPS (+5V) ----------1

PROGRAM:

i) By using MASM:

```
        Assume cs: code
        Code segment
Start:  MOV AX, 4343
        MOV BX, 1111
        SUB AX, BX
        INT 3
        Code ends
        End start
```

ii) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	INSTRUCTION

4000			MOV AX,4343
			MOV BX,1111
			SUB AX,BX
			INT 3

OUTPUT:

Input		output	
Register	**Data**	**Register**	**Data**
AX	4343	AX	3232
BX	1111		

b. Aim: 16-Bit Addition in Location mode using 8086 Microprocessor Kit (location mode).

Apparatus:

1. 8086 Microprocessor Trainer Kit.
2. Compatible Keyboard.
3. µP Power Supply.

PROGRAM:

i) By using MASM:

```
Assume cs: code
Code segment
Start: MOV SI, 2000H
MOV AX, [SI]
ADD SI, 02
MOV BX, [SI]
SUB AX, BX
MOV DI, 3000H
MOV [DI], AX
INT 03
Code ends
End start
```

ii) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC

4000			MOV SI,2000
			MOV AX,[SI]
			ADD SI,02
			MOV BX,[SI]
			SUB AX,BX
			MOV DI,3000
			MOV [DI],AX
			INT 03

OUTPUT:

output			
MEMORY LOCATION	**Data**	**MEMORY LOCATION**	**Data**
2000	78	3000	44
2001	56	3001	44
2002	34		
2003	12		

3. Multiplication:

a. AIM: - To write an assembly language program for multiplication of two 16-bit numbers (register mode).

APPARATUS: 1.8086 microprocessor kit/MASM-------1

2. RPS (+5V) ----------1

PROGRAM:

i) By using MASM:

```
        Assume cs: code
        Code segment
Start:  MOV AX, 0040
        MOV BX, 0002
        MUL BX
        INT 3
        Code ends
        End start
```

ii) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC

4000			MOV AX,0040
			MOV BX,0020
			MUL BX
			INT 3

OUTPUT:

Input		Output	
Register	**Data**	**Register**	**Data**
AX	0040	AX	0080
BX	0002	DX	0000

b. AIM: - To write an assembly language program for multiplication of two 16-bit signed numbers (location mode).

APPARATUS: 1.8086 microprocessor kit/MASM ----1

2. RPS (+5V) ----1

PROGRAM:

i) By using MASM:

```
Assume cs: code
Code segment
Start: MOV SI, 2000
MOV AX, [SI]
ADD SI, 02
MOV BX, [SI]
MUL BX
  MOV
  DI,3000H
  MOV [DI], AX
  INT 03
  Code ends
  End start
```

ii) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC

400			MOV SI, 2000
			MOV AX, [SI]
			ADD SI, 02
			MOV BX, [SI]
			MUL BX
			MOV DI, 3000
			MOV [DI], AX
			INT 03

OUTPUT:

Input		output	
MEMORY LOCATION	**Data**	**MEMORY LOCATION**	**Data**
2000	40	3000	80
2001	00	3001	00
2002	02		
2003	00		

4. Division:

a. AIM: - To write an assembly language program for multiplication of two 16-bit numbers (register mode).

APPARATUS: 1.8086 microprocessor kit/MASM-------1

2. RPS (+5V) ----------1

PROGRAM:

A) By using MASM:

```
        Assume cs: code
        Code segment
Start:  MOV AX, 0040
        MOV BX, 0002
        DIV BX
        INT 3
        Code ends
        End start
```

B) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC

4000			MOV AX,0040
			MOV BX,0002
			DIV BX
			INT 3

OUTPUT:

Input		output	
Register	**Data**	**Register**	**Data**
AX	0040	AX	0020
BX	0002		

b. AIM: -To write an assembly language program for multiplication of two 16-bit numbers (location mode).

APPARATUS: 1.8086 microprocessor kit/MASM ----1

2. RPS (+5V) ----1

PROGRAM:

A) By using MASM:

```
Assume cs: code
Code segment
Start: MOV SI, 2000
MOV AX, [SI]
ADD SI, 02
MOV BX, [SI]
 DIV BX
  MOV DI, 3000
  MOV [DI], AX
  INT 03
   Code ends
   End start
```

B) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC

400			MOV SI, 2000
			MOV AX, [SI]
			ADD SI, 02
			MOV BX, [SI]
			DIV BX
			MOV DI, 3000
			MOV [DI], AX
			INT 03

OUTPUT:

Input		output	
MEMORY LOCATION	**Data**	**MEMORY LOCATION**	**Data**
2000	40	3000	80
2001	00	3001	00
2002	02		
2003	00		

RESULT: 16 bit arithmetical operations are performed by using different addressing modes.

Viva Questions

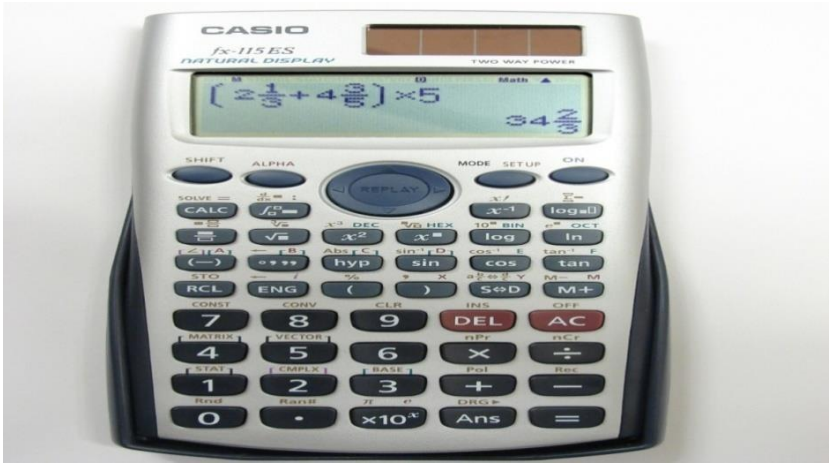
1) How many bit 8086 microprocessor is?
2) What is the size of data bus of 8086?
3) What is the size of address bus of 8086?
4) What is the max memory addressing capacity of 8086?
5) Which are the basic parts of 8086?
6) Difference between 8085 and 8086?
7) Define Minimum mode operation in 8086?
8) Define Maximum mode operation in 8086?
9) What is the purpose of AX register?
10) What is the purpose of CX register?
11) What are the features of Intel 8086 ?
12) What are the flags in 8086?
13) What is 1st / 2nd / 3rd / 4th generation processor?
14) How many bit combinations are there in a byte?
15) What are the different functional units in 8086?
16) What are the various segment registers in 8086?
17) Which Stack is used in 8086?
18) What is SIM and RIM instructions?
19) What is meant by Interrupt?
20) What is an Instruction?
21) What is Microprocessor?
22) Define Compiler
23) Define Interpreter
24) Define Assembler
25) What is Assembly leve l language?
26) Abbreviate MASM & TASM
27) Differentiate micro & macro
28) Define two pass assembler
29) What are the advantages of modular programming?
30) What is linking and relocation?

EXERCISE:

1. Write an alp program for multi byte addition?
2. Write an alp program for multi byte subtraction?
3. Write an alp program for one word addition?
4. Write an alp program for one byte subtraction?
5. Write an alp program for one word addition?
6. Write an alp program for one word subtraction?
7. Write an alp program for one byte multiplication?
8. Write an alp program for one byte multiplication?
9. Write an alp program for one byte division?
10. Write an alp program for one byte division?
11. Write alp program for 8 bit signed multiplication?
12. Write alp program for 8 bit signed division?
13. Write alp program for 8 bit unsigned addition?
14. Write alp program for 8 bit unsigned subtraction?
15. Write an ALP program to displaying the system clock time shows 01:25:30?
16. Write alp program for 16 bit signed addition?
17. Write alp program for 16 bit signed subtraction?
18. Write alp program for 16 bit signed multiplication?
19. Write alp program for 16 bit signed division?
20. Write an alp program for addition and subtraction of two 16bit numbers?

Industrial applications

Calculators,Vending-machines.



A calculator is a device that performs arithmetic operations on numbers. The simplest calculators can do only addition, subtraction, multiplication, and division. More sophisticated calculators can handle exponent ial operations, roots, logarithm s, trigonometric functions, and hyperbolic functions. Internally, some calculators actually perform all of these functions by repeatedprocessesofaddition.

EXP NO.2: Program for sorting an array for 8086.

a) ASCENDING ORDER

AIM:-Program to sort the given numbers in ascending order

APPARATUS: 1.8086 microprocessor kit/MASM-------1

2. RPS (+5V) ----------1

PROGRAM:

A) By using MASM:

```
                ASSUME CS: CODE
                CODE SEGMENT
START:          MOV AX, 0000H
                MOV CH, 0004H
                DEC CH
UP1 :           MOV CL, CH
        MOV SI, 2000
UP:             MOV AL, [SI]
                INC SI
                CMP AL, [SI]
                JC DOWN
                XCHG AL, [SI]
                DEC SI
                MOV [SI], AL
                INC SI
DOWN:           DEC CL
                JNZ UP
                DEC CH
                JNZ UP1
                INT 3
CODE ENDS
        END START
```

B) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC
4000			MOV AX, 0000H
			MOV CH, 0004H
			DEC CH
		UP1:	MOV CL, CH
			MOV SI,2000
		UP:	MOV AL,[SI]
			INC SI
			CMP AL,[SI]
			JC DOWN
			XCHG AL,[SI]
			DEC SI
			MOV [SI],AL
			INC SI
			DEC CL
		DOWN:	JNZ UP
			DEC CH
			JNZ UP1
			INT 3

OUTPUT:

Input		**output**	
MEMORY LOCATION	**Data**	**MEMORY LOCATION**	**Data**
2000	03	2000	03
2001	06	2001	04
2002	07	2002	06
2003	04	2003	07

b) DESCENDING ORDER

AIM:-Program to sort the given numbers in descending order

APPARATUS: 1.8086 microprocessor kit/MASM-------1

2. RPS (+5V) ----------1

PROGRAM:

A) By using MASM:

```
                ASSUME CS: CODE
                CODE SEGMENT
START:          MOV AX, 0000H
                MOV CH, 0004H
                DEC CH
UP1 :           MOV CL, CH
        MOV SI, 2000
UP:             MOV AL, [SI]
                INC SI
                CMP AL, [SI]
                JNC DOWN
                XCHG AL, [SI]
                DEC SI
                MOV [SI], AL
                INC SI
DOWN:           DEC CL
                JNZ UP
                DEC CH
                JNZ UP1
                INT 3
CODE ENDS
        END START
```

B) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC

4000			MOV AX, 0000H
			MOV CH, 0004H
			DEC CH
			MOV CL, CH
			MOV SI,2000
			MOV AL,[SI]
			INC SI
			CMP AL,[SI]
			JNC DOWN
			XCHG AL,[SI]
			DEC SI
			MOV [SI],AL
			INC SI
			DEC CL
			JNZ UP
			DEC CH
			JNZ UP1
			INT 3

		UP1: UP: DOWN:	

OUTPUT:

Input		output	
MEMORY LOCATION	**Data**	**MEMORY LOCATION**	**Data**
2000	03	2000	07
2001	06	2001	06
2002	07	2002	04
2003	04	2003	03

RESULT: Program for sorting an array performed by using masm software and trainer kit.

Viva:

1) What are the functions of BIU?

2) What are the functions of EU?

3) How many pin IC 8086 is?

4) What IC8086 is?

5) What is the size of instruction queue in 8086?

6) What are the functions of BX register?

7) What are the functions of DX register?

8) How many pin IC 8085 is?

9) What IC8085 is?

10) What is the size of instruction queue in 8085?

11) What does EU do?

12) Difference between JMP and JNC?

13) What are the basic units of a microprocessor ?

14) What is the data and address size in 8086?

15) What are the modes in which 8086 can operate?

16) What are the interrupts of 8086?

17) What are the functional units available in 8086 architecture?

18) When the 8085 processor checks for an interrupt?

19) What is USART?

20) Define stack

21) What is Tri-state logic?

22) What is Program counter?

23) How many bit combinations are there in a byte?

24) What is meant by Maskable interrupts?

25) What is Non-Maskable interrupts?

26) What are the various segment registers in 8086?

27) What does EU do?

28) What are Flag registers?

29)W hat does the 8086 Architecture contain?

30) What are Data Copy/Transfer Instructions?

EXERCISE:

1. Write an alp program for multi byte multiplication in location mode?
2. Write an alp program for multi byte division in location mode?
3. Write an alp program for one byte multiplication in location mode?
4. Write an alp program for one byte division in location mode?
5. Write an alp program for one word multiplication in location mode?
6. Write an alp program for one word division in location mode?
7. Write an alp program for multi byte addition in location mode?
8. Write an alp program for multi byte subtraction in location mode?
9. Write an alp program for one word addition in location mode?
10. Write an alp program for one byte subtraction in location mode?
11. Write an alp program for addition and subtraction of two 16bit numbers?
    1) A278
    2) B634
12. Write an alp program for multiplication and division of two 16bit numbers?
    1)0012
    2)0006
13. Write an alp program for to sort the given number in descending order?
    1)14 2)A2 3)85 4)54
14. Write an alp program to sort the given numbers in ascending order?
    1) 1E2)2A 3) 56 4)98
15. Write an alp program tosearch a number 05 from a given array?
    1)06 2)05 3)08 4)02
16. Write an alp program to search a number 45 from a given array?
    1) 09 2)45 3)22 4)A2
17. Write an alp for insertor delete abyte in a given string with SI memory location is 4000and DI location is 6000?
18. Write an alp for moving or reversing the given stringwith the length of the string is 12?
19. Write an ALP program todisplaying the system clock time shows 10:15:05?
20. Write an alp program for one byte division in location mode?

Applications: Database management.

A **database-management system** (**DBMS**)is a computer-software application that interacts with end-users, other applications, and the database itself to capture and analyze data. A general-purpose DBMS allows the definition, creation, querying, update, and administration of databases. Well known DBMS include MySQL, PostgreSQL, Enterprise DB, MongoDB, MariaDB, Microsoft SQL Server, Oracle, Sybase, SAP HANA, Mem SQL, SQLite and IBM DB2.

EXP NO: 3

Program for searching for a number or character in a string for 8086.

AIM: Write an alp program for to search a number or character from a string.

APPARATUS: 1.8086 microprocessor kit/MASM-------1

2. RPS (+5V)----------1

PROGRAM:

A) SEARCHING A NUMBER:

```
ASSUME CS: CODE
CODE SEGMENT
START:
        MOV SI,2000H
        MOV BX,2500H
        MOV CH,05H
        MOV DX,0000H
BACK:MOV AL,[SI]
        CMP AL,[SI]
        JZ DOWN
        INC SI
        DEC CH
        JNZ BACK
        JMP DOWN1
DOWN: INC DX
DOWN1:INT 03H
CODE ENDS
END START
```

B) SEARCHING A CHARACTER:

```
Assume Cs:Code,Ds:Data
Data Segment
Str DB “MLRITM”
Count  Equ 06h
Search Db “R”
L1  “given character is present$”
L2 “given character is absent$”
Data ends
Code segment
Start:   mov ax,data
         Mov ds,ax
          Mov si ,offset str
          Mov bx,offset search
         Mov cx,count
Back:mov al,[si]
        Cmp al,[bx]
         Jz down
         Inc si
         Dec cx
        Jnz back
         Mov dx,offset l2
         Jmp down1
Down: mov dx,offset l1
Down1: mov ah,09h
           Int 21h
            Int 03h
```

Code ends

End start

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC
4000			MOV AX, 15H
			MOV SI, 2000
			MOV DI, 3000
			MOV CX, 0005
			MOV BX,6000
		BACK:	MOV AL,[SI]
			CMP AL,[BX]
			JZ DOWN
			REP
			MOVSB
			JNZ BACK
			JMP DOWN
			MOV [DI],AL
		DOWN:	JMP DOWN2
			MOV AX, 00
		DOWN1:	MOV [DI], AL
			INT 3H

OUTPUT:

Input		**output**	
MEMORY LOCATION	**Data**	**MEMORY LOCATION**	**Data**
2000	53	2500	15
2001	15	DX	01
2002	19		
2003	02		
6000	15		

RESULT: Program for search a number or character from a string performed by using masm software and trainer kit.

Viva:

1) What is the size of instruction queue in 8086?
2) Which are the registers present in 8086?
3) What do you mean by pipelining in 8086?
4) How many 16 bit registers are available in 8086?
5) Specify addressing modes for any instruction?
6) Define segmentation?
7) Describe General purpose registers?
8) Describe special purpose registers?
9) Describe the segment registers?
10) Define ALU?
11) What is Bandwidth ?
12) What is Clock Speed ?
13) What are the features of Intel 8086 ?
14) What is Logical Address:?
15) What is the size of instruction queue in 8086?
16) Which are the registers present in 8086?
17) What do you mean by pipelining in 8086?
18) How many 16 bit registers are available in 8086?
19) Specify addressing modes for any instruction?
20) What are Machine Control Instructions?
21) What are Flag Manipulation Instructions?
22) What are String Instructions?
23) What are different parts for 8086 architecture?
24) What is an Interrupts
25) What is an Opcode?
26) What is an Operand?
27) Explain the difference between a JMP and CALL instruction?
28) What is meant by Interrupt?
29) What is an Instruction?
30) What is Microcontroller and Microcomputer?

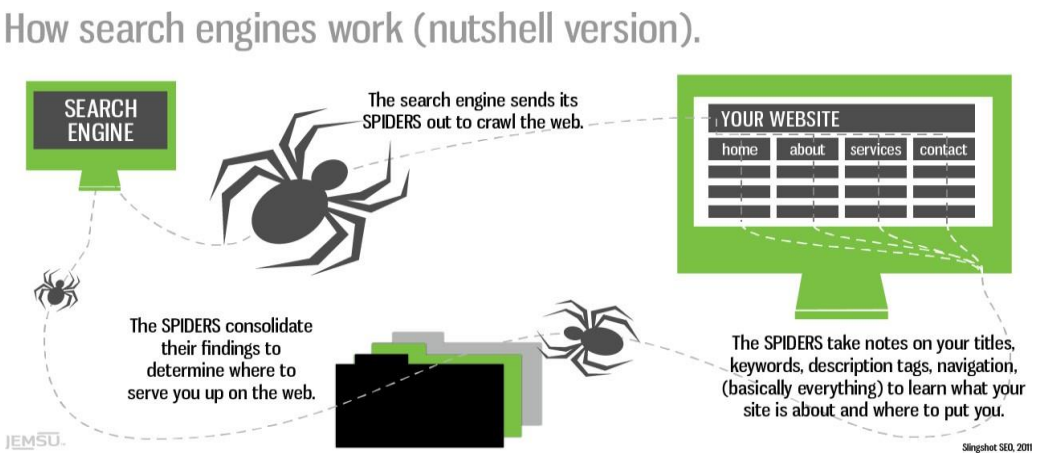
EXERCISE:

Write an alp program for median of an array?

Write an alp program for next number of median in an array?

1. Write an alp program to find out how many multiples of given number in a given array?
2. Write an alp program to find how many divisible numbers by given number in array?
3. Write alp program for 8 bit signed addition?
4. Write alp program for 8 bit signed subtraction?
5. Write alp program for 8 bit signed multiplication?
6. Write alp program for 8 bit signed division?
7. Write alp program for 8 bit unsigned addition?
8. Write alp program for 8 bit unsigned subtraction?
9. Write alp program for 8 bit unsigned addition?
10. Write alp program for 8 bit unsigned subtraction?
11. Write an ALP program to displaying the system clock time shows 01:25:30?
12. Write an ALP program to displaying the system clock time shows 10:15:05?
13. 16Bit Addition in Location mode using 8086 Microprocessor Kit.
14. 16Bit subtraction in Location mode using 8086 Microprocessor Kit
15. Write an alp program to find the smallest number in an array using masm software
16. Write an alp program to find the largest number in an array using masm software.
17. Write an alp program to find the ASCII number to the given BCD number 56
18. using 8086 triner kit
19. Write an alp program to find the unpacked BCD to the given BCD number 56
20. using 8086 triner ki

Applications: Search engines.



A **web search engine** is a software system that is designed to search for information on the World Wide Web. The search results are generally presented in a line of results often referred to as search engine results pages (SERPs). The information may be a mix of web pages, images, and other types of files. Some search engines also mine data available in databases or open directories. Unlike web directories, which are maintained only by human editors, search engines also maintain real-time information by running an algorithm on a web crawler.

EXP NO.4:

Program for string manipulations for 8086.

1) Moving string from one location to another location

AIM: Write an alp for moving a string from one location to another location.

APPARATUS: 1. 8086 microprocessor kit/MASM ----- 1

2. RPS (+5V)---------- 1

PROGRAM:

A) Moving string:

```
        Assume cs:code,ds:data,es:extra
Data segment
Srcdata db “mpmc lab$”
Count equ 08h
Data ends
Extra segment
Dstdata db 12 dup(0)
Extra ends
Code segment
Start: mov ax,data
       Mov ds,ax
       Mov ax,extra
       Mov es,ax
       Mov cx,count
       Mov si,offset srcdata
       Mov di,offset dstdata
       Cld
       Rep movsb
       Mov ah,09h
       Int 21h
       Int 03h
Code ends
End start
```

B) Reverse of a string

```
                Assume cs:code,ds:data,es:extra
Data segment
String  db “mpmc lab$”
Count equ $-string
Data ends
Extra segment
Reverse  db 0000
Extra ends
Code segment
Start: mov ax,data
        Mov ds,ax
        Mov ax,extra
        Mov es,ax
        Mov cx,count
        Mov si,offset string
        Mov di,offset reverse
        Add di,count-1
        Cld
Back: mov al,[si]
        Mov ES:[di],al
        inc si
       dec di
        jnz back
        Mov ah,09h
        Int 21h
        Int 03h
Code ends
End start
```

C) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC
4000			MOV SI,2000
			MOV DI,3000
			MOV BX,5000
			MOV CX,0005
			CLD
			MOV AL,[SI]
		L1	CMP AL,[BX]
			JZ L2
			MOVSB
			JMP L3
			MOVSB
		L2	MOV BX,7000
			MOV AL,[BX]
			MOV [DI],AL
			DEC CX
			INC DI
			REP
			MOVSB
			INT 3
		L3	

OUTPUT:

Input		**output**	
MEMORY LOCATION	**Data**	**MEMORY LOCATION**	**Data**
2000	02	3000	02
2001	04	3001	04
2002	43	3002	43
2003	76	3003	08
2004	01	3004	76
5000	43	3005	01
7000	08		

D) By using 8086 kit:

MEMORY LOCATION	OP-CODE	LABEL	MNEMONIC
4000			MOV SI,2000
			MOV DI,3000
			MOV BX,5000
			MOV CX,0005
			CLD
			MOV AL,[SI]
		L1	CMP AL,[BX]
			JZ L2
			MOVSB
			LOOP L1
			JMP L3
			INC SI
		L2	DEC CX
			REP
			MOVSB
			INT 3
		L3	

OUTPUT:

Input		**output**	
MEMORY LOCATION	**Data**	**MEMORY LOCATION**	**Data**
2000	01	3000	01
2001	02	3001	02
2002	03	3002	03
2003	04	3003	05
2004	05		
5000	04		

RESULT: Program for string manipulation performed by using masm software and trainer kit.

Viva:

1) What do you mean by assembler directives?

2) What .model small stands for?

3) What is the supply requirement of 8086?

4) What is the relation between 8086 processor frequency & crystal Frequency?

5) What are the functions of Accumulator or AX register?

6) What are the functions of BX register?

7) What are the functions of CX register?

8) What are the functions of DX register?

9) What are the functions of CS register?

10) What are the functions of DS register?

11) What do you mean by assembler directives?

12) What .model small stands for?

13) What is the supply requirement of 8086?

14) What is the relation between 8086 processor frequency & crystal Frequency?

15) Functions of Accumulator or AX register?

16) What is meant by Maskable interrupts?

17) What is Non-Maskable interrupts?

18) What are the various segment registers in 8086?

19) What are the flags in 8086?

20) What are the different types of Addressing Modes?

21) What is Assembler?

22) Define Variable?

23) Explain Dup?

24) Why address bus is unidirectional?

25) What is macro?

26) What is the difference between Macro and Procedure?

27) How 8086 is faster than 8085?

28) What does microprocessor speed depend on?

29) What is the size of data bus and address bus in 8086?

30) What is the maximum memory addressing capability of 8086

EXERCISE:

1. Write an alp for moving a string from one location to another location?
2. Write an alp for reversing the given string with the length of the string is 12?
3. Write an alp for arranging given a string in alphabetical order?
4. Write alp program for 16-bit signed addition?
5. Write alp program for 16-bit signed subtraction?
6. Write alp program for 16-bit signed multiplication?
7. Write alp program for 16-bit signed division?
8. Write alp program for 16-bit unsigned addition?
9. Write alp program for 16-bit unsigned subtraction?
10. Write alp program for 16-bit unsigned multiplication?
11. Write an alp for insertor delete a byte in a given string with SI memory location is 4000 & DI location is 6000?
12. Write an alp for moving or reversing the given string with the length of the string is 12?
13. Write an alp program to perform OR operation using 8051microcontroller trainer Kit?
14. Write an alp program to perform addition and subtraction operation using 8051

    A) 56

    B) 12

15. Write an ALP program to study timer1 gated mode
16. Write an alp program to find the length of the given array using masm software.
17. Write an alp program to find the sum of,, numbers using masm software.
18. Write an alp program to perform an operation to find the sum of squares of a given array using masm software.
19. Write an alp program to perform an operation to find the cubes of squares of a given array using masm software
20. Write alp program for 16-bit signed addition?

Applications: Voice-assistants.

Voice assistant are the next big thing. Some say they're the next mobile, though I don't even know if that's accurate or an understatement. All the major platform companies have one, and startups building them appear ever faster, making it hard to even keep track of everything. The point is, they are going to be everywhere and are going to dominate the way we interact with our computers. Yet I hear many questioning if these assistants are even viable from a business perspective. The argument goes that by moving people away from screens, assistants may be diminishing traditional screen-based revenue streams. How is Google going to sell ads along their search results if the user gets taken directly to the information, they desire without ever looking at a list of results?

Content providers may indeed have harder time turning their work into paychecks. If you're running a blog or publication, your main business is placing ads next to your reporting. When more people move away from screens and have their news read to them by an AI instead, less people will see your ads. Though if that is something people are actually going to do in significant quantities remains to be seen. For the companies operating the voice assistants, however, they will become a gold mine. Even better, their value proposition for the customer is precisely what makes them valuable for the operating businesses.

EXP.NO.5:

Program for Rotate,Shift and Branch instructions using 8086.

AIM: To write an ALP program for Rotate,Shift and Branch instructions using 8086.

APPARATUS:1. MASM Software

2.PC

1.Exchange of two numbers:

PROGRAM:

```
        ASSUME CS: CODE
        CODE SEGMENT
START:  MOV AX,1234H
        MOV BX,5678H
        XCHG AX,BX
        INT 03H
        CODE ENDS
        END START
```

2.Average of two numbers:

```
        PROGRAM:
        ASSUMECS: CODE
        CODE SEGMENT
START:  MOV AX,0000H
        MOV AL,11H
        MOV BL,33H
        ADD AL,BL
        SHR AL,1
        INT 03H
        CODE ENDS
        END START
```

3.To double the given byte:

PROGRAM:

```
        ASSUME CS: CODE
        CODE SEGMENT
START:  MOV AX,0000H
        MOV AL,11H
        MOV BL,33H
        ADD AL,BL
        SHL AL,1
        INT 03H
        CODE ENDS
        END START
```

RESULT: Program for rotate Shift and Branch instructions were performed using masm software.

Viva:

1) Functions of BX register?
2) Functions of CX register?
3) Functions of DX register?
4) How Physical address is generated?
5) Which are pointers present in this 8086?
6) What are the functions of BIU?
7) What are the functions of EU?
8) How many pin IC 8086 is?
9) What IC8086 is?
10) What is the size of instruction queue in 8086?
11) What are the different types of ADC?
12) What is an Interrupt?
13) Define opcode and operand?
14) What is DMA?
15) Define machine cycle?
16) What are the interrupts of 8086?
17) What is the data and address size in 8086?
18) Define bit, byte and word?
19) What is assembly language?
20) Difference between JMP and JNC?
21) Stack is used in 8086?
22) What is macro?
23) What is a compiler?
24) What is the disadvantage of microprocessor?
25) Which Stack is used in 8086?
26) What is the difference between 8086 and 8088?
27) What are the functional units in 8086?
28) What is a Microprocessor?
29) What is meant by Maskable Interrupts?
30) Give example for Non-Maskable Interrupts?

EXERCISE:

1. Write an ALP program to displaying the system clock time shows 01:25:30?
2. Write an ALP program to displaying the system clock time shows 10:15:05?
3. Write an ALP to perform average of 1 to 10 natural numbers
4. Write an ALP to perform the Armstrong number.
5. Write an ALP to perform the reverse operation of a given number.
6. Write an ALP to perform the decimal to binary conversion for the given number.
7. Write an ALP to perform the binary to decimal conversion for the given number.
8. Write an ALP to perform the decimal to hexadecimal conversion for the given number
9. Write an ALP to perform average of 1 to 20 natural numbers
10. Write an ALP to perform the percentage of 1 to 20 natural numbers
11. Write an alp program to perform an operation to find the squares of a given number using masm software.
12. Write an alp program to perform an operation to find the squares of a given number using MP trainer kit
13. Write an alp program to perform an operation to find the cubes of a given number using masm software
14. Write an alp program to perform an operation to find the cubes of a given numbers using MP trainer kit
15. Write an alp program for addition of multi byte numbers.
16. Write an alp program to divide 32 bit by the 16 bit.
17. Write an alp program for median of an array.
18. Write an ALP to perform average of 1 to 20 natural numbers
19. Write an ALP to perform the percentage of 1 to 20 natural numbers
20. Write an ALP program to displaying the system clock time shows 10:15:05?

EXP.NO.6:
Parallel communication between two microprocessors using 8255.

AIM: To write an alp for parallel communication between two microprocessors by using 8255.

APPARATUS: 8086 Trainer kit-2, 8255, Power Supply and connectors.

PROCEDURE: -

1. Connect the 26 core FRC connector to the 8086 trainers at connector no CN4 and the interface module.
2. Connect the power mate connector to the interface module and the other side of the connector to the power supply. The connections to the power supply are given below.

Connections: (power supply)

Black & Red: Gnd.

Blue & Green: +5V

3. 5- Way power mate is wired to the motor. This power mate is to be inserted into the male socket provided on the interface. Care should be taken such that, below given code for the particular-colored wire coincides with the code on the interface.

A- GREEN

C- RED & WHITE

B- GREEN & WHITE

D- RED

V_{DD}- BLACK & WHITE.

4. After the completion of the program and connections enter the program as given in the listing below.

G0< STARTING ADDRESS< ENTER (on the key board of trainer).

PROGRAM:

```
TRANSMITTER:
            Mov al,80h
            Mov dx,0FFC6
            Out dx,al
            Mov al,55h
            Mov dx,0FFC0
            Out dx,al

RECIEVER:   Mov al,90h
            Mov dx,0FFC6
            Out dx,al
            Mov dx,0FFC0
            IN AL,DX
```

OUT PUT:
AX=0055H

PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MNEMONICS

4000		**LOOP1**	MOV AL,90
			MOV DX,3006
			OUT DX
			MOV DX,3000
			IN AL,DX
			NOT AL
			MOV DX,3002
			OUT DX
			MOV AL,02
			MOV DX,3006
			OUT DX
			CALL **DELAY**
			MOV AL,03
			MOV DX,3006
			OUT DX
			CALL **DELAY**
			MOV AL,0A
			MOV DX,3006
			OUT DX
			CALL **DELAY**
			MOV AL,0B
			MOV DX,3006
			OUT DX
			CALL **DELAY**
			MOV AL,0E
			MOV DX,3006
			OUT DX
			CALL DELAY
			MOV AL,0F
			MOV DX,3006
			OUT DX
			CALL DELAY
			JMP **LOOP1**

DELAY PROGRAM

MEMORY LOCATION	OPCODE	LABEL	MNEMONICS
4500		**NEXT**	MOV CX,7FFF LOOP **NEXT** RET

RESULT: Program for parallel communication between two microprocessors by using 8255performed.

Viva:

31) Functions of BX register?
32) Functions of CX register?
33) Functions of DX register?
34) How Physical address is generated?
35) Which are pointers present in this 8086?
36) What are the functions of BIU?
37) What are the functions of EU?
38) How many pin IC 8086 is?
39) What IC8086 is?
40) What is the size of instruction queue in 8086?
41) What are the different types of ADC?
42) What is an Interrupt?
43) Define opcode and operand?
44) What is DMA?
45) Define machine cycle?
46) What are the interrupts of 8086?
47) What is the data and address size in 8086?
48) Define bit, byte and word?
49) What is assembly language?
50) Difference between JMP and JNC?
51) Stack is used in 8086?
52) What is macro?
53) What is a compiler?
54) What is the disadvantage of microprocessor?
55) Which Stack is used in 8086?
56) What is the difference between 8086 and 8088?
57) What are the functional units in 8086?
58) What is a Microprocessor?
59) What is meant by Maskable Interrupts?
60) Give example for Non-Maskable Interrupts?

EXERCISE:

1. Write an ALP program to displaying the system clock time shows 01:25:30?
2. Write an ALP program to displaying the system clock time shows 10:15:05?
3. Write an ALP to perform average of 1 to 10 natural numbers
4. Write an ALP to perform the Armstrong number.
5. Write an ALP to perform the reverse operation of a given number.
6. Write an ALP to perform the decimal to binary conversion for the given number.
7. Write an ALP to perform the binary to decimal conversion for the given number.
8. Write an ALP to perform the decimal to hexadecimal conversion for the given number
9. Write an ALP to perform average of 1 to 20 natural numbers
10. Write an ALP to perform the percentage of 1 to 20 natural numbers
11. Write an alp program to perform an operation to find the squares of a given number using masm software.
12. Write an alp program to perform an operation to find the squares of a given number using MP trainer kit
13. Write an alp program to perform an operation to find the cubes of a given number using masm software
14. Write an alp program to perform an operation to find the cubes of a given numbers using MP trainer kit
15. Write an alp program for addition of multi byte numbers.
16. Write an alp program to divide 32 bit by the 16 bit.
17. Write an alp program for median of an array.
18. Write an ALP to perform average of 1 to 20 natural numbers
19. Write an ALP to perform the percentage of 1 to 20 natural numbers
20. Write an ALP program to displaying the system clock time shows 10:15:05?

Cycle 2
Using
8051 Microcontroller Kit

Exp.No.07

Programming using arithmetic, logical and bit manipulation instructions of 8051.

I) Arithmetical operations:

i) 8-bit addition

AIM: To perform 8-bit addition by using 8051.

APPARATUS: 8051 with keyboard

PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MNEMONIC

8000			MOV A, #02
			MOV B, #02
			ADD A,B
			LCALL 03

OUTPUT:

Input		output	
REGISTER	**Data**	**REGISTER**	**Data**
A	02	A	04
B	02		

ii) 8-bit subtraction

AIM: To perform 8-bit subtraction by using 8051.

APPARATUS: 8051 with keyboard

PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MNEMONIC

8000			MOV A,#04
			MOV B,#02
			SUBB A,B
			LCALL 03

OUTPUT:

Input		output	
REGISTER	**Data**	**REGISTER**	**Data**
A	04	A	02
B	02		

iii) 8-bit multiplication:

AIM: To perform 8-bit multiplication by using 8051.

APPARATUS: 8051 with keyboard

PROGRAM:

Memory location	Opcode	Label	Mnemonic

8000			MOV A,#04H
			MOV OF0,#02H
			MUL AB
			LCALL 03

OUTPUT:

Input		output	
MEMORY LOCATION	**Data**	**REGISTER**	**Data**
9000	03	A	06
9001	02		

iv) 8-bit division:

AIM: To perform 8-bit division by using 8051.

APPARATUS: 8051 with keyboard

PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MNEMONIC

8000			MOV A,#04H
			MOV OF0,#02H
			DIV AB
			LCALL 03

OUTPUT:

Input		output	
MEMORY LOCATION	**Data**	**REGISTER**	**Data**
9000	03	A	06
9001	02		

II) logical operations:

i) AND operation

AIM: To perform AND operation by using 8051.

APPARATUS: 8051 with keyboard

PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MEMONIC
8000			MOV R0,#DATA 1 MOV A,#DATA 2 ANL A,R0 MOV R1,A LCALL 03

OUTPUT:

Input		output	
REGISTER	**Data**	**REGISTER**	**Data**
R0	14 (DATA 1)	R1	10
A	12 DATA 2)		

ii) XOR operation

AIM: To perform AND operation by using 8051.

APPARATUS: 8051 with keyboard

PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MEMONIC
8000			MOV R0,#DATA 1 MOV A,#DATA 2 XRL A,R0 MOV R1,A LCALL 03

OUTPUT:

Input		output	
REGISTER	**Data**	**REGISTER**	**Data**
R0	23 (DATA 1)	R1	17
A	34 (DATA 2)		

BIT AND BYTE OPERATIONS BY USING 8051

AIM: To write an assembly language program to perform the BIT and BYTE operations like set, reset and swap by using 8051 microcontrollers.

APPARATUS:

1. 8051 Micro Controller kit.
2. Key Board.
3. Adapter.

PROGRAM:

BIT OPERATIONS:

SET A BIT:

```
        MOV     DPTR, #STARTING ADDRESS
        MOVX    A, @DPTR
        SETB    0E5
        INC     DPTR
        MOVX    @DPTR, A
L4:     SJMP    L4 (OFFSET ADDRESS)
        LCALL   03
```

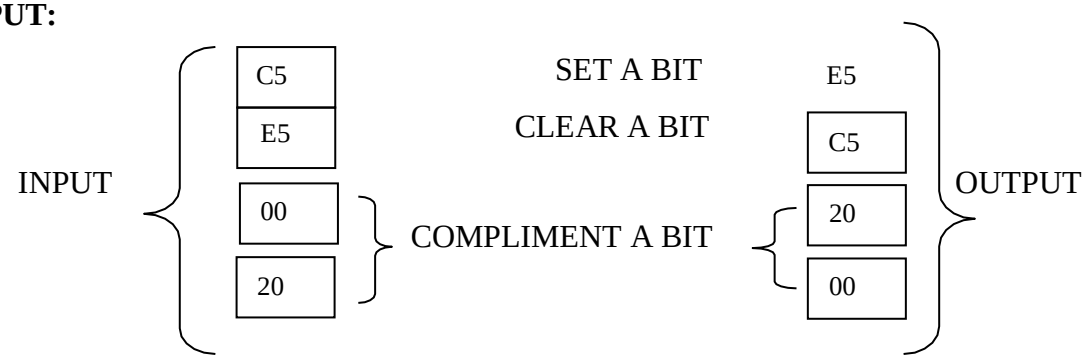
RESET A BIT:

```
        MOV     DPTR, #STARTING ADDRESS
        MOVX    A, @DPTR
        CLR     0E5
        INC     DPTR
        MOVX    @DPTR, A
L4:     SJMP    L4 (OFFSET ADDRESS)
        LCALL   03
```

COMPLIMENT A BIT:

```
        MOV     DPTR, #STARTING ADDRESS
        MOVX    A, @DPTR
        CPL     0E5
        INC     DPTR
        MOVX    @DPTR, A
L4:     SJMP    L4 (OFFSET ADDRESS)
        LCALL   03
```

OUTPUT:



PROGRAM:

BYTE OPERATONS:

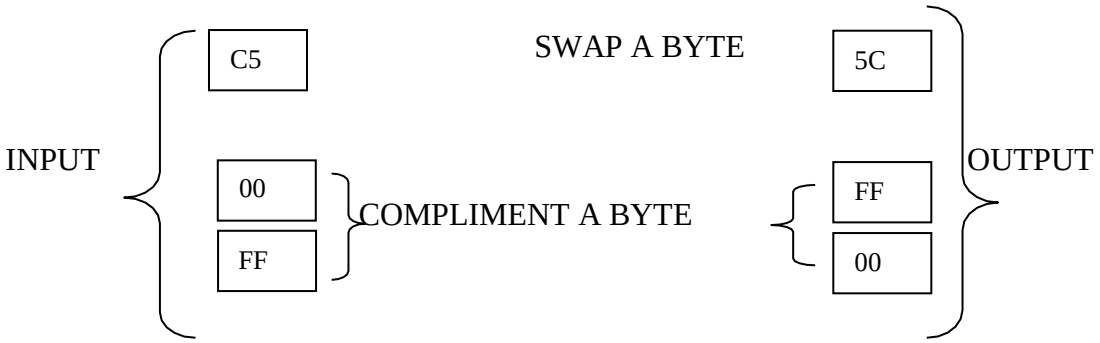
SWAP A BYTE:

```
        MOV     DPTR, #STARTING ADDRESS
        MOVX     A, @DPTR
        SETB    A
        INC     DPTR
        MOVX     @DPTR, A
L4:     SJMP    L4 (OFFSET ADDRESS)
        LCALL   03
```

COMPLIMENT A BYTE:

```
        MOV     DPTR, #STARTING ADDRESS
        MOVX     A, @DPTR
        CPL     A
        INC     DPTR
        MOVX     @DPTR, A
L4:     SJMP    L4 (OFFSET ADDRESS)
        LCALL   03
```

OUTPUT:



RESULT: Programs for arithmetic, logical and bit manipulation instructions of 8051 performed

Viva:

1) What is the function of 01h of Int 21h?
2) What is the function of 02h of Int 21h?
3) What is the function of 09h of Int 21h?
4) What is the function of 0Ah of Int 21h?
5) What is the function of 4ch of Int 21h?
6) What is Microprocessor
7) Explain the function of CPU in Microprocessor
8) Define Compiler
9) Define Interpreter
10) Define Assembler
11) What does u mean by Prefix?
12) What model small means?
13) Difference between small, medium, tiny, huge?
14) What is dd, dw, db?
15) Interrupts in 8086 and there function
16) What is the reset address of 8086?
17) What is the size of flag register in 8086? Explain all.
18) What is the difference between 08H and 01H functions of INT 21H?
19) Which is faster- Reading word size data whose starting address is at even or at odd address of memory in 8086?
20) Which is the default segment base: offset pairs?
21) What is Assembly level language
22) What are Mnemonics
23) Distinguish between Microprocessor & Microcontroller.
24) Define address bus, data bus and control bus
25) What is nibble, byte and word
26) How many bit combinations are there in a byte?
27) Have you studied buses? What types?
28) What is the Maximum clock frequency in 8086?
29) What is meant by Maskable interrupts?
30) What are the different functional units in 8086?

EXERCISE:

1. Write an alp program to find even and odd numbers in a given array?
2. Write an alp program to find sum of even numbers in an array?
3. Write an ALP to **LCM** of two numbers
4. Write an ALP to search a number in given string.
5. Write an ALP to search a character in the given string.
6. Write an ALP to convert binary to gray code.
7. Write an ALP to convert gray to binary code.
8. Write an ALP to find out **square root** of given number. **Num = (225) $_D$**
9. Write an ALP to find **factorial** of given number.(N=6)
10. Write an ALP to find number of times letter 'e' exist in the given string 'exercise'
11. Write an alp program to find the unpacked BCD to the given BCD number 56 using 8086 trainer kit?
12. Write an alp program to find the ASCII number to the given BCD number 56 using 8086 trainer kit?

13) Which is the default segment base: offset pairs?

14) What is Assembly level language

15) What are Mnemonics

16) Distinguish between Microprocessor & Microcontroller.

17) Define address bus, data bus and control bus

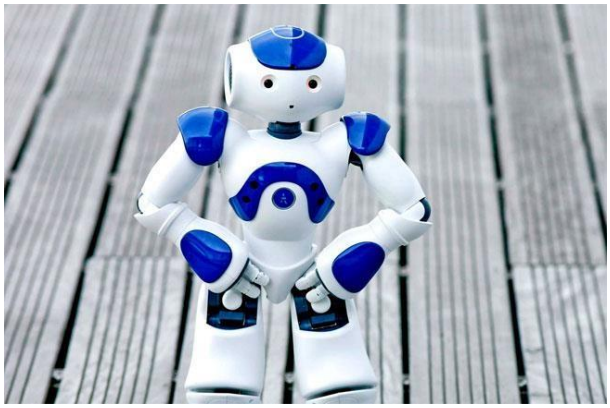
18) What is nibble, byte and word

19) Write an ALP to search a number in given string.

20) Write an ALP to search a character in the given string.

Industrial Applications

Robots.



1. **Industrial robots** – These robots bring into play in an industrialized manufacturing atmosphere. Typically these are articulated arms particularly created for applications like- material handling, painting, welding and others. If we evaluate merely by application then this sort of robots can also consist of some automatically guided automobiles and other robots.

2. **Domestic or household robots** – Robots which are used at home. This sort of robots consists of numerous different gears for example- robotic pool cleaners, robotic sweepers, robotic vacuum cleaners, robotic sewer cleaners and other robots that can perform different household tasks. Also, a number of scrutiny and tele presence robots can also be considered as domestic robots if brought into play in that sort of environment.

3. **Medical robots** – Robots employed in medicine and medicinal institutes. First & foremost surgical treatment robots. Also, a number of robotic directed automobiles and perhaps lifting supporters.

4. **Service robots** – Robots that cannot be classed into any other types by practice. These could be various data collecting robots, robots prepared to exhibit technologies, robots employed for research, etc.

5. **Military robots** – Robots brought into play in military & armed forces. This sort of robots consist of bomb discarding robots, various shipping robots, exploration drones. Often robots at the start produced for military and armed forces purposes can be employed in law enforcement, exploration and salvage and other associated fields.

6. **Entertainment robots** – These types of robots are employed for entertainment. This is an extremely wide-ranging category. It begins with model robots such as robot sapiens or the running photo frames and concludes with real heavy weights like articulated robot arms employed as movement simulators.

7. **Space robots** – I would like to distinct out robots employed in space as a split apart type. This type of robots would consist of the robots employed on Canadair that was brought into play in space Shuttles, the International Space Station, together with Mars explorers and other robots employed in space exploration & other activities.

8. **Hobby and competition robots** – Robots that is created by students. Sumo-bots, Line followers, robots prepared merely for learning, fun and robots prepared for contest

EXP.NO.8

Interfacing ADC to 8051.

AIM:

1. To write a program for conversion of analog data to digital output.

2. To write a program for conversion of digital data to analog output. The analog output will be in the form of triangular wave, saw tooth wave, square wave/rectangular wave.

APPARATUS:

1. 8086 Trainer.
2. Power supply for trainer and interface module.
3. A/D interface module.
4. Power mate connector.
5. FRC connector.
6. Cathode ray oscilloscope

PROCEDURE: -

1. Connect the 26 core FRC connector to the 8086 trainers at connector no CN4 and the interface module.
2. Connect the power mate connector to the interface module and the other side of the connector to the power supply. The connections to the power supply are given below.

   Connections: (power supply)Black & Red: Gnd. Blue & Green: +5V
3. 5- Way power mate is wired to the motor. This power mate is to be inserted into the male socket provided on the interface. Care should be taken such that, below given code for the particular-colored wire coincides with the code on the interface.A- GREE C- RED & WHITEB- GREEN & WHITE D- REDV_{DD}- BLACK & WHITE.
4. After the completion of the program and connections enter the program as given in the listing below.

   G0< STARTING ADDRESS< ENTER (on the key board of trainer).

A/D CONVERTER

PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MNEMONIC

			MOV AL,90
			MOV DX,0FFC6
			OUT DX
			MOV AL,07
			MOV DX,FFC4
			OUT DX
			MOV AL,0F
			MOV DX,0FFC6
			OUT DX
			MOV CX,3FFF
			LOOP **D1**
			MOV AL,0E
			MOV DX,0FFC6
			OUT DX

		D1	MOV AL,0C
			MOV DX,0FFC6
			OUT DX
			MOV DX,0FFC0
			IN DX
			AND AL,80
			CMP AL,80
			JNZ**D2**
			MOV AL,0D
			MOV DX,0FFC6

		D2	OUT DX
			MOV DX,0FFC0
			IN DX
			MOV DX,0FFC2
			OUT DX
			JMP **D3**

OUTPUT:

INPUT:

POWER SUPPLY: 05V

CHANNEL NO:'00 T0

OUTPUT:

DISPLAY SHOWS: FF

RESULT: Program for interfacing ADC to 8086 performed.

Viva:

1. Can we use SP as offset address holder with CS?
2. Which is the base registers in 8086?
3. Which is the index registers in 8086?
4. What do you mean by segment override prefix?
5. Whether micro reduces memory requirements?
6. What do u mean by assembler?
7. What do u mean by linker?
8. What do u mean by loader?
9. What do u mean by compiler?
10. What do you mean by emulator?
11. Stack related instruction?
12. What are Mnemonics
13. Distinguish between Microprocessor & Microcontroller.
14. Define address bus, data bus and control bus
15. What is nibble, byte and word
16. How many bit combinations are there in a byte?
17. Have you studied buses? What types?
18. What is the Maximum clock frequency in 8086?
19. What is meant by Maskable interrupts?
20. What are the different functional units in 8086?
21. What is the purpose of CX register?
22. What are the features of Intel 8086 ?
23. What are the flags in 8086?
24. What is 1st / 2nd / 3rd / 4th generation processor?
25. How many bit combinations are there in a byte?
26. What are the different functional units in 8086?
27. What are the various segment registers in 8086?
28. Which Stack is used in 8086?
29. What is SIM and RIM instructions?
30. What is meant by Interrupt?

EXERCISE QUESTIONS:

1. Write an alp program to perform an operation to find the cubes of a given number using masm software
2. Write an alp program to perform an operation to find the cubes of a given numbers using MP trainer kit
3. Write an alp program to find out how many multiples of given number in a given array?
4. Write an alp program to find how many divisible numbers by given number in a given array?
5. Write alp program for 8-bit signed addition in 8051 kit?
6. Write alp program for 8-bit signed subtraction in 8051 kit?
7. Write alp program for 8-bit signed multiplication in 8051 kit?
8. Write alp program for 8-bit signed division in 8051 kit?
9. Write alp program for 8-bit unsigned addition in 8051 kit?
10. Write alp program for 8-bit unsigned subtraction in 8051 kit?
11. Write an alp program for addition of multi byte numbers
12. Write an alp program for multiplication of given number in location mode

    A)0060

    B)0002
13. Write an alp program to divide 32 bits by the 1bit.
14. Write an alp program for median of an array

EXP.NO.09:
Interfacing DAC to 8051.

AIM:

2. To write a program for conversion of analog data to digital output.

3. To write a program for conversion of digital data to analog output. The analog output will be in the form of triangular wave, saw tooth wave, square wave/rectangular wave.

APPARATUS:

1. 8086 Trainer.
2. Power supply for trainer and interface module.
3. A/D interface module.
4. Power mate connector.
5. FRC connector.
6. Cathode ray oscilloscope.

PROCEDURE: -

5. Connect the 26 core FRC connector to the 8086 trainers at connector no CN4 and the interface module.
6. Connect the power mate connector to the interface module and the other side of the connector to the power supply. The connections to the power supply are given below.

   Connections: (power supply)Black & Red: Gnd. Blue & Green: +5V
7. 5- Way power mate is wired to the motor. This power mate is to be inserted into the male socket provided on the interface. Care should be taken such that, below given code for the particular-colored wire coincides with the code on the interface.A- GREENC- RED & WHITEB- GREEN & WHITE D- REDV_{DD}- BLACK & WHITE.
8. After the completion of the program and connections enter the program as given in the listing below.

   G0< STARTING ADDRESS< ENTER (on the key board of trainer).

D/A CONVERTER:

PROGRAM TO GENERATE SQUARE WAVE:

MEMORY LOCATION	OPCODE	LABEL	MNEMONIC

			MOV AL,80
			MOV DX,0FFC6
			OUT DX
			MOV DX,0FFC2
			MOV AL,00

		A0	OUT DX CALL DELAY 1 MOV AL,0FF OUT DX CALL DELAY2 JMP A0

DELAY PROGRAM1

ADDRESS	OPCODE	LABEL	MNEMONIC
		A1	MOV CX,0020 LOOP A1 RET

DELAY PROGRAM2

ADDRESS	OPCODE	LABEL	MNEMONIC
		A2	MOV CX,0020 LOOP A2 RET

OUTPUT:

INPUT: OUTPUT:

POWER SUPPLY: 05V DISPLAY SHOWS: FF

CHANNEL NO:'00 T0

RESULT: Program for interfacing DAC to 8086 performed.

EXP.NO.10
Interrupt handling in 8051

AIM: Write ALP in 8051 to allow the external interrupt 1.

APPARATUS:8051 with keyboard interrupt kit module.

PROCEDURE:

1. Make the power supply connections from 4-way power mate connector on the ALS-NIFC-09 board.
    - +5V..... blue wire
    - Ground..... black wire
2. Connect 26-pin flat cable from interface module to P1 of the trainer kit.
3. Enter the program in the RAM location in 9000 and execute the program GO<STARTING ADDRESS><EXEC>

PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MEMONIC
			ORG 0000
		AGAIN	LJMP AGAIN
			ORG 0013
			SETB P1.3
			MOV R3,#255
		BACK	DJNZ R3,BACK
			CLR P1.3
			ORG 30H
		MAIN	MOV IE,#10000100B
		HERE	SJMP HERE
			LCALL 03

OUTPUT:

1. When key is pressed, LED ON.
2. When key is opened, LED OFF

RESULT: program for interrupt handling in 8051 verified.

Viva:

1) What is the reset address of 8086?

2) What is the size of flag register in 8086? Explain all.

3) What is the difference between 08H and 01H functions of INT 21H?

4) Which is faster- Reading word size data whose starting address is at even or at odd address of memory in 8086?

5) Which is the default segment base: offset pairs?

6) What is Assembly level language.

7) What are Mnemonics

8) Distinguish between Microprocessor & Microcontroller.

9) Define address bus, data bus and control bus

10) What is nibble, byte and word

11) Can we use SP as offset address holder with CS?

12) Whether micro reduces memory requirements?

13) What do you mean by segment override prefix?

14) Which is the index registers in 8086?

15) Which is the base registers in 8086?

16) What do you mean by macro?

17) What is diff between macro and procedure?

18) Types of procedure?

19) What TASM is?

20) What TLINK is?

21) Name the processor lines of two major manufacturers?

22) How many bit combinations are there in a byte?

23) Have you studied buses? What types?

24) What is the Maximum clock frequency in 8086?

25) What is meant by Maskable interrupts?

26) What are the different functional units in 8086?

27) What are the various segment registers in 8086?

28) What is SIM and RIM instructions?

29) What are the different types of Addressing Modes?

30) What are the General Data Registers & their uses?

EXERCISE:

1. Write an ALP program to find number of odd numbers in a given array.
2. Write an ALP to **insert** a character in the given string.
3. Write an ALP to **delete** a character from the given string.
4. Write an ALP to find the **median** from the list of numbers
5. Write an ALP to convert given Hexadecimal number into its equivalent ASCII number.
6. Write an ALP to convert given ASCII number into its equivalent Hexadecimal number.
7. Write an ALP to find **Result = 1 + 2^2 + 3^3**.
8. Write an ALP to find **Result = N^2 + N^3** .(N=6)
9. Write an ALP to find **Result = N^N**.(N=5)
10. Write an ALP to find **Result = 3^N+ N^3** +6 (If N=5)
11. Write an alp program to perform OR operation using 8051 microcontroller trainerKit?
12. Write an alp program to perform addition and subtraction operation using 8051microcontroller trainer Kit?

    A)56 B) 12
13. Write an alp program to find the length of the given array using masm software
14. Write an alp program to find the sum ofn‟ numbers using masm software
15. Write an alp program to perform an operation to find the cubes of squares of a given array using masm software
16. Write an alp program to perform an operation to find the sum of squares of a given array using masm software
17. Write an alp program to find the length of the given array using masm software.
18. Write an alp program to find the sum of ‘n’ odd numbers using masm software.
19. Write an ALP to search a number in given string.
20. Write an ALP to search a character in the given string.

Industrial applications:

Digital clocks

A digital clock is a type of clock that displays the time digitally (i.e. in numerals or other symbols), as opposed to an analog clock, where the time is indicated by the positions of rotating hands. Digital clocks are often associated with electronic drives, but the "digital" description refers only to the display, not to the drive mechanism. (Both analog and digital clocks can be driven either mechanically or electronically, but "clockwork" mechanisms with digital displays are rare.) The biggest digital clock is the Lichtzelt Pegel ("Light Time Level") on the television tower Rheinturm Düsseldorf, Germany.

Digital clocks typically use the 50 or 60 hertz oscillation of AC power or a 32,768hertz crystal oscillator as in a quartz clock to keep time. Most digital clocks display the hour of the day in 24-hour format; in the United States and a few other countries, a more commonly used hour sequence option is 12-hour format (with some indication of AM or PM). Some timepieces, such as many digital watches, can be switched between 12-hour and 24-hour modes. Emulations of analog-style faces often use an LCD screen, and these are also sometimes described as "digital".

EXP.NO.11

Time delay generation using 8051

AIM: Interface the 8251 USART to the two 8086 microprocessor kits.

APPARATUS:

1. 8086 Trainer kit-2no's
2. 8251 USART
3. Power Supply
4. Connectors.

PROCEDURE:-

1. Connect the 26 core FRC connector to the 8086 trainer at connector no CN4 and the interface module.
2. Connect the power mate connector to the interface module and the other side of the connector to the power supply. The connections to the power supply are given below.

Connections: (power supply)

Black & Red: Gnd.

Blue & Green: +5V

3. 5- Way power mate is wired to the motor. This power mate is to be inserted into the male socket provided on the interface. Care should be taken such that, below given code for the particular colored wire coincides with the code on the interface.

A- GREEN

C- RED & WHITE

B- GREEN & WHITE

D- RED

V_{DD}- BLACK & WHITE.

4. After the completion of the program and connections enter the program as given in the listing below.

G0< STARTING ADDRESS< ENTER (on the key board of trainer).

PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MNEMONICS
4000			MOV AL,36
			MOV DX,0086H
			OUT DX,AL
			MOV DX,0080H
			MOV AL,0A
			OUT DX,AL
			MOV AL,00
			OUT DX,AL
			MOV SP,3000
			MOV DX,0092
			OUT DX,AL
			OUT DX,AL
			OUT DX,AL
			OUT DX,AL
			CALL DELAY
			MOV AL,40
			OUT DX,AL
			CALL DELAY
			MOV AL,CE
			OUT DX,AL
			CALL DELAY
			MOV AL,27
			OUT DX,AL
			CALL DELAY
			MOV SI,2100
		L1	MOV DX,0092
			IN AL,DX
			CMP AL,1B
			JE L1
			MOV DX,0090
			IN AL,DX
			AND AL,81

			CMP BL,AL
			JE L3
		L2	MOV DX,0092
			IN AL,DX
			AND AL,81
			CMP AL,81
			JNE L2
			MOV AL,BL
			MOV DX,0090
			OUT DX,AL
			OUT DX,AL
			MOV [SI],AL
			INC SI
			JMP L1
			OUT DX,AL
			INC SI
			JMP L2
		L3	INT 03

DELAY PROGRAM:

MEMORY LOCATION	**OPCODE**	**LABEL**	**MNEMONIC**
4500			MOV CX,0002
		A3	LOOP A3
			RET

RESULT: Program for serial communication between two microprocessors by using8251Performed

Viva:

1) What does u mean by Prefix?
2) What .model small means?
3) Difference between small, medium, tiny, huge?
4) What is dd, dw, db?
5) Write Interrupts in 8086 and there function.
6) Expand USART?
7) Where do we prefer the serial communication?
8) What is the function of instruction pointer (IP) register?

9) What is the difference between IN and OUT instructions?

10) What is MODEM?

11)What is the reset address of 8086?

12) What is the size of flag register in 8086? Explain all.

13) What is the difference between 08H and 01H functions of INT 21H?

14) Which is faster- Reading word size data whose starting address is at even or at odd address of memory in 8086?

15) Which is the default segment base: offset pairs?

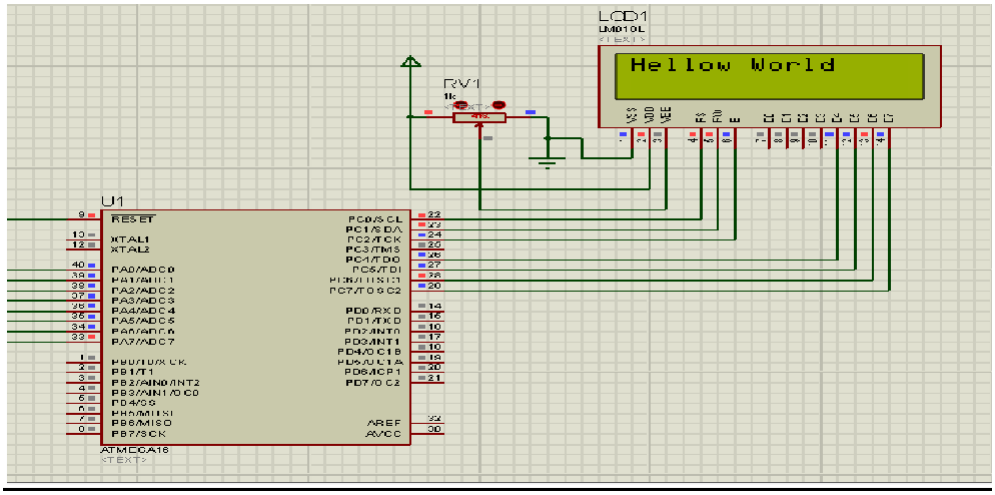
16) What is the difference between instructions DIV & IDIV?

17) What is difference between shifts and rotate instructions?
18) Which are strings related instructions?
19) Which are addressing modes and their examples in 8086?
20) What does u mean by directives?
21) While displaying no. from user why u need to add 30 to that?
22) What are ASCII codes for nos. 0 to F
23) How does U differentiate between positive and negative numbers?
24) What is range for these numbers?
25) Which no. representation system you have used?
26) What do you mean by emulator?
27) Stack related instruction?
28) stack 100 means?
29) What do you mean by 20 dup (0)?
30) Which flags of 8086 are not present in 8085?

EXERCISE:

1. Write an alp program to find even or odd numbers in a given array?
2. Write an alp program to find the sum of 'n' numbers?
3. Write an ALP to count the number of 0's and 1's in the given data. **Num = (25) H**
4. Write an ALP to find **square and cube** of a number.(N=6)
5. Write an ALP to 16-bit signed multiplication.
6. Write an ALP to 8-bit signed division.
7. Write an ALP to **insert** a character in the given string.
8. Write an ALP to **delete** a character from the given string.
9. Write an ALP to find the **median** from the list of numbers
10. Write an ALP to convert given Hexadecimal number into its equivalent ASCII number
11. Write an alp program to find the smallest number in an array using masm software.
12. Write an alp program to find the largest number in an array using masm software
13. Write an ALP to perform the decimal to binary conversion for the given number.
14. Write an ALP to perform the binary to decimal conversion for the given number.
15. Write an ALP to perform the decimal to hexadecimal conversion for the given number.
16. Write an ALP to perform the reverse operation of a given string.
17. Write an ALP to convert binary to gray code.
18. 16-Bit Addition in Location mode using 8086 Microprocessor Kit
19. 16-Bit subtraction in Location mode using 8086 Microprocessor Kit.
20. Write an alp program to find the smallest number in an array using masm software.

INDUSTRIAL APPLICATIONS:



Interfacing is one of the important concepts in microcontroller 8051 because the microcontroller is a CPU that can perform some operation on a data and gives the output. However, to perform the operation we need an input device to enter the data and in turn output device displays the results of the operation. Here we are using keyboard and LCD display as input and output devices along with the microcontroller.

Interfacing is the process of connecting devices together so that they can exchange the information and that proves to be easier to write the programs. There are different type of input and output devices as for our requirement such as LEDs, LCDs, 7segment, keypad, motors and other devices.

EXP.NO.12

Interfacing to 8086 and programming to control stepper motor.

AIM: Write an Assembly Language Program to rotate the Stepper Motor in clockwise as well as anti-clockwise direction.

APPARATUS: 8086 Trainer kit, Stepper,

Motor Interface Card,

Stepper Motor,

Power supply.

PROCEDURE: -

1. Connect the 26 core FRC connector to the 8086 trainer at connector no CN4 and the interface module.
2. Connect the power mate connector to the interface module and the other side of the connector to the power supply. The connections to the power supply are given below.

Connections: (power supply)

Black & Red: Gnd.

Blue & Green: +5V

3. 5- Way power mate is wired to the motor. This power mate is to be inserted into the male socket provided on the interface. Care should be taken such that, below given code for the particular-colored wire coincides with the code on the interface.

A- GREEN

C- RED & WHITE

B- GREEN & WHITE

D- RED

V_{DD}- BLACK & WHITE.

4. After the completion of the program and connections enter the program as given in the listing below.

G0< STARTING ADDRESS< ENTER (on the key board of trainer).

Program to rotate in clockwise direction

MEMORY LOCATION	OPCODE	LABEL	MNEMONIC
4000			
			MOV AL,80H
			MOV DX,0FFC6
			OUT DX,AL
			MOV CL,0FF
			MOV DX,0FFC4
			MOV AL,0EE
		L1	OUT DX,AL
			CALL DELAY
			ROR AL,1
			DEC CL
			JNZ L1
			INT 03H
		DELAY	MOV BX,8000
		L2	DEC BX
			JNZ L2
			RET

Program to rotate in Anti clockwise direction

MEMORY LOCATION	OPCODE	LABEL	MNEMONIC

4000			MOV AL,80H
			MOV DX,0FFC6
			OUT DX,AL
			MOV CL,0FF
			MOV DX,0FFC4
			MOV AL,0EE

		L1	OUT DX,AL CALL DELAY ROL AL,1 DEC CL JNZ L1 INT 03H

DELAY PROGRAM:

MEMORY LOCATION	OPCODE	LABEL	MNEMONIC
		L2	MOV BX,8000 DEC BX JNZ L2 RET

RESULT: An operation to interface Stepper Motor with8086 Microprocessor performed.

Viva:

28) What does u mean by Prefix?
29) What .model small means?
30) Difference between small, medium, tiny, huge?
31) What is dd, dw, db?
32) Write Interrupts in 8086 and there function.
33) Expand USART?
34) Where do we prefer the serial communication?
35) What is the function of instruction pointer (IP) register?

36) What is the difference between IN and OUT instructions?

37) What is MODEM?

38)What is the reset address of 8086?

39) What is the size of flag register in 8086? Explain all.

40) What is the difference between 08H and 01H functions of INT 21H?

41) Which is faster- Reading word size data whose starting address is at even or at odd address of memory in 8086?

42) Which is the default segment base: offset pairs?

43) What is the difference between instructions DIV & IDIV?

44) What is difference between shifts and rotate instructions?
45) Which are strings related instructions?
46) Which are addressing modes and their examples in 8086?
47) What does u mean by directives?
48) While displaying no. from user why u need to add 30 to that?
49) What are ASCII codes for nos. 0 to F
50) How does U differentiate between positive and negative numbers?
51) What is range for these numbers?
52) Which no. representation system you have used?
53) What do you mean by emulator?
54) Stack related instruction?
28) stack 100 means?
31) What do you mean by 20 dup (0)?
32) Which flags of 8086 are not present in 8085?

EXERCISE:

1. Write an alp program to find even or odd numbers in a given array?
2. Write an alp program to find the sum of 'n' numbers?

21 Write an ALP to count the number of 0's and 1's in the given data. **Num = (25) H**

22 Write an ALP to find **square and cube** of a number.(N=6)

23 Write an ALP to 16-bit signed multiplication.

24 Write an ALP to 8-bit signed division.

25 Write an ALP to **insert** a character in the given string.

26 Write an ALP to **delete** a character from the given string.

27 Write an ALP to find the **median** from the list of numbers

28 Write an ALP to convert given Hexadecimal number into its equivalent ASCII number

29 Write an alp program to find the smallest number in an array using masm software.

30 Write an alp program to find the largest number in an array using masm software

31 Write an ALP to perform the decimal to binary conversion for the given number.

32 Write an ALP to perform the binary to decimal conversion for the given number.

33 Write an ALP to perform the decimal to hexadecimal conversion for the given number.

34 Write an ALP to perform the reverse operation of a given string.

35 Write an ALP to convert binary to gray code.

36 16-Bit Addition in Location mode using 8086 Microprocessor Kit

37 16-Bit subtraction in Location mode using 8086 Microprocessor Kit.

38 Write an alp program to find the smallest number in an array using masm software.

EXP.NO.13
Interfacing Matrix/Keyboard to 8051.

AIM: Interface a Keyboardto8051 microcontroller.

APPARATUS:8051 Trainer kit, keyboard module, FRC cables, &Power Supply.

PROCEDURE:

1. Make the power supply connections from 4-way power mate connector on the ALS-NIFC-09 board.
   - +5V..... blue wire
   - Ground..... black wire
2. Connect 26-pin flat cable from interface module to P1 of the trainer kit.
3. Enter the program in the RAM location in 9000 and execute the program
   GO<STARTING ADDRESS><EXEC>

PROGRAM:

```
CNTRL   EQU  2043H    ; CONTROL PORT ADDRESS OF 8255
PORTA   EQU  2040H    ; PORTA ADDRESS OF  8255
PORTB   EQU  2041H    ; PORTB ADDRESS OF 8255
PORTC   EQU  2042H    ; PORTC ADDRESS OF 8255
```

ADDRESS	OPCODE	LABEL	MNEMONICS
			MOV A,#90H
			MOV DPTR,#CNTRL
			MOVX @DPTR,A
			MOV B,#20H
		BLINK2	MOV DPTR,#PORTB
			MOV A,#FFH
			MOVX@DPTR,A
			MOV DPTR,#PORTC
			MOV A,#00H
			MOVX@DPTR,A
			MOV A,#F0H
			MOVX@DPTR,A
			DJNZ B,BLNK2
		BACK	MOV A,#FEH
			MOV B,#21H
		BLINK1	MOV DPTR,#PORTB

			MOVX@DPTR,A
			MOV DPTR,#PORTC
			MOV A,#00H
			MOVX@DPTR,A
			MOV A,#F0H
			MOVX@DPTR,A
			LCALL DELAY
			RL A
			DJNZ B,BLNK1
			SJMP BACK

		DELAY:
		OLOOP:
		ILOOP:
			MOV R0,#F7H
			MOV R1,#FFH
			DJNZ R1,ILOOP
			DJNZ R0,OLOOP
			RET

RESULT: program for interfacing a keyboard to 8051 microcontroller performed.

Viva:

1) What is the size of flag register?
2) Can you perform 32 bit operation with 8086? How?
3) Whether 8086 is compatible with Pentium processor?
4) What is 8087? How it is different from 8086?
5) While accepting no. from user why u need to subtract 30 from that?
6)Define instruction cycle
7) What is an instruction set?
8) Give the functional categories of 8086 microprocessor instructions of data transfer operations?
9) Define Op-code and operand
10) Define the types of branching operations.
10) Define the types of branching operations?
11) Explain the function of CPU in Microprocessor Define Compiler?
12) Define Intrepreter?
13) What is Assembly level language?
14) Define Assembler?
15) What is Microprocessor?
16) What is Logical Address:?
17) What is The Effective Address:
18) What is Physical Address?
19) What are the flags in 8086?
20) What is Tri-state logic?
21) Why crystal is a preferred clock source?
22) What happens when HLT instruction is executed in processor?
23) What are the different functional units in 8086?
24) What is the position of the Stack Pointer after the PUSH instruction?
25) What is the position of the Stack Pointer after the POP instruction?
26) Bring out the differences between 8086 and 8088?
27) What is meant by ìbootstrap loader?
28) Give practical applications where macro can be used?
29) Briefly describe how ìdirectî and ìindirectî Jumps take place in 8086?
30) Discuss the syntax of macro

EXERCISE:

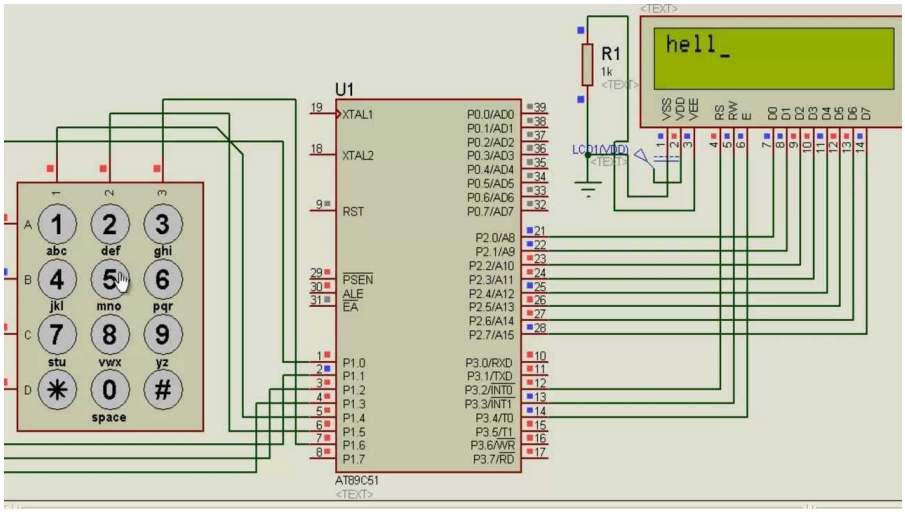
1. Write an alp program for division of 32 bit number by 16 bit number.
2. Write an alp program for multiplication of given number in location mode
3. Write an alp for moving a string from one location to another location?
4. Write an alp for reversing the given string with the length of the string is 12?
5. Write an alp for arranging given a string in alphabetical order?
6. Write alp program for 16 bit signed addition in 8051 kit?
7. Write alp program for 16 bit signed subtraction in 8051 kit?
8. Write alp program for 16 bit signed multiplication in 8051 kit?
9. Write alp program for 16 bit signed division in 8051 kit?
10. Write alp program for 16 bit unsigned addition in 8051 kit?
11. What are the internal devices of 8255 ?
12. 12What are the different functional units in 8086?

13What is the position of the Stack Pointer after the PUSH instruction?

14What is the position of the Stack Pointer after the POP instruction?

13) 15Bring out the differences between 8086 and 8088?
14) 16What is meant by ìbootstrap loader?
15) 17Give practical applications where macro can be used?
16) 18.Briefly describe how ìdirectî and ìindirectî Jumps take place in 8086?
17) Discuss the syntax of macro

Industrial Applications:



Interfacing is one of the important concepts in microcontroller 8051 because the microcontroller is a CPU that can perform some operation on a data and gives the output. However to perform the operation we need an input device to enter the data and in turn output device displays the results of the operation. Here we are using keyboard and LCD display as input and output devices along with the microcontroller.

Interfacing is the process of connecting devices together so that they can exchange the information and that proves to be easier to write the programs. There are different type of input and output devices as for our requirement such as LEDs, LCDs, 7segment, keypad, motors and other devices.

These interview questions test the knowledge of x86 Intel architecture and 8086 microprocessors specifically.

1. What is a Microprocessor? - Microprocessor is a program-controlled device, which fetches the instructions from memory, decodes and executes the instructions. Most Micro Processor are single- chip devices.

2. Give examples for 8 / 16 / 32 bit Microprocessor? - 8-bit Processor - 8085 / Z80 / 6800; 16-bit Processor - 8086 / 68000 / Z8000; 32-bit Processor - 80386 / 80486.

3. Why 8085 processor is called an 8 bit processor? - Because 8085 processor has 8 bit ALU (Arithmetic Logic Review). Similarly 8086 processor has 16 bit ALU.

4. What is 1st / 2nd / 3rd / 4th generation processor? - The processor made of PMOS / NMOS / HMOS / HCMOS technology is called 1st / 2nd / 3rd / 4th generation processor, and it is made up of 4 / 8 / 16 / 32 bits.

5. Define HCMOS? - High-density n- type Complementary Metal Oxide Silicon field effect transistor.

6. What does microprocessor speed depend on? - The processing speed depends on DATA BUS WIDTH.

7. Is the address bus unidirectional? - The address bus is unidirectional because the address information is always given by the Micro Processor to address a memory location of an input / output devices.

8. Is the data bus is Bi-directional? - The data bus is Bi-directional because the same bus is used for transfer of data between Micro Processor and memory or input / output devices in both the direction.

9. What is the disadvantage of microprocessor? - It has limitations on the size of data. Most Microprocessor does not support floating-point operations.

10. What is the difference between microprocessor and microcontroller? - In Microprocessor more op-codes, few bit handling instructions. But in Microcontroller: fewer op-codes, more bit handling Instructions, and also it is defined as a device that includes micro processor, memory, & input / output signal lines on a single chip.

11. What is meant by LATCH? - Latch is a D- type flip-flop used as a temporary storage device controlled by a timing signal, which can store 0 or 1. The primary function of a Latch is data storage. It is used in output devices such as LED, to hold the data for display.

12. Why does microprocessor contain ROM chips? - Microprocessor contain ROM chip because it contain instructions to execute data.

13. What is the difference between primary & secondary storage device? - In primary storage device the storage capacity is limited. It has a volatile memory. In secondary storage device the storage capacity is larger. It is a nonvolatile memory. Primary devices are: RAM / ROM. Secondary devices are: Floppy disc / Hard disk.

14. Difference between static and dynamic RAM? - Static RAM: No refreshing, 6 to 8 MOS transistors are required to form one memory cell, Information stored as voltage level in a flip flop. Dynamic RAM: Refreshed periodically, 3 to 4 transistors are required to form one memory cell; Information is stored as a charge in the gate to substrate capacitance.

15. What is interrupt? - Interrupt is a signal send by external device to the processor so as to request the processor to perform a particular work.

16. What is cache memory? - Cache memory is a small high-speed memory. It is used for temporary storage of data & information between the main memory and the CPU (center processing unit). The cache memory is only in RAM.

17. What is called .Scratch pad of computer? - Cache Memory is scratch pad of computer.

18. Which transistor is used in each cell of EPROM? - Floating .gate Avalanche Injection MOS (FAMOS) transistor is used in each cell of EPROM.

19. Differentiate between RAM and ROM? - RAM: Read / Write memory, High Speed, Volatile Memory. ROM: Read only memory, Low Speed, Non Volatile Memory.

20. What is a compiler? - Compiler is used to translate the high-level language program into machine code at a time. It doesn't require special instruction to store in a memory, it stores automatically. The Execution time is less compared to Interpreter.

21. Which processor structure is pipelined? - All x86 processors have pipelined structure.

22. What is flag? - Flag is a flip-flop used to store the information about the status of a processor and the status of the instruction executed most recently