MGM’S

Jawaharlal Nehru Engineering College

Laboratory Manual

MICROCONTROLLER APPLICATIONS

For

TE (EEP)

Prof.J.R.Rana

Author

JNEC Aurangabad.

PREFACE

It is my great pleasure to present this laboratory manual for third year EEP students for the subject of Microcontroller Application (MCA).

Microcontroller is a general/Special purpose programmable logic device. A thorough understanding of 8051 Microcontroller concepts, demands Assembly Language programming with in depth knowledge of instructions and clarity in analysis of the task.

Most of the students find it difficult to start with the programming assignment. Therefore, a structured approach to learn programming through steps like problem statement, analysis, logic & flowchart, actual program (Mnemonics) and comments is illustrated with sample programs.

This lab manual introduces students to the elementary programming techniques, interfacing and designing simple applications using 8051 Microcontroller.

Students are advised to thoroughly go through this manual rather than only topics mentioned in the syllabus, as practical aspects are the key to understanding and conceptual visualization of theoretical aspects covered in the books.

Good Luck for your Enjoyable Laboratory Sessions.

Author

SUBJECT INDEX

I Do’s and Don’ts.

II Lab exercises.

PART A

Revision of Digital Electronics Concepts.

Introduction to the 8051 Simulator.

PART B

* 1. Assembly Language programs Addition.
		1. Arithmetic operation on two 8 Bit numbers.
		2. Arithmetic operation on two 16 Bit numbers.

2. Assembly Language programs Multiplication and division of 8 bit numbers.

 3. Assembly language program of finding the average of five numbers.

 4. Assembly language program of finding the Largest/Smallest number.

 5. Generation of Time delay.

 6. LED Blinking Program.

 7. Rolling numbers on seven segment display.

 8. Interfacing of stepper motor.

 9. Generation of square wave using microcontroller.

 PART C

 Application of Microcontroller.

III. Quiz on the subject.

IV. Conduction Viva-Voce Examination.

V. Evaluation and Marking Systems.

3

I .DO’s and DON’T’s in Laboratory:

1. Do not handle any kit before reading the instructions/Instruction manuals.
2. Use correct power supply with the proper kit.
3. Do not forcefully place connectors to avoid the damage.
4. Strictly observe the instructions given by the teacher/Lab Instructor.

Instruction for Laboratory Teachers:

1. Lab work completed during prior session, should be corrected during the next lab session.
2. Students should be guided and helped whenever they face difficulties.
3. The promptness of submission should be encouraged by way of marking and evaluation patterns that will benefit the sincere students.

4

**II. Lab Exercises: Part A**

[Purpose of these exercises is to introduce the students to concepts of digital electronics used in microcontroller programming followed by introduction to the simulator.

**Exercises 1: Use of Digital Electronics in programming of 8051 Microcontroller.**

**THEORY:**

A. Revision of concept of Register (8 bit) , Memory devices, Counter, Timer.

B. Revision of Data processing techniques.

**Details:**

1. Definition of Flip-flops

2. Applications of flip-flop.

3. Types of Register.

4. Types of Memory Devices.

5. Use of counter and Timer.

6. Data transfer techniques.

**1. Register:**



**2. Memory Devices:**



**3. Counter/Timer:**



**Data processing techniques details:**

1. Synchronous Data and Asynchronous data

2. FIFO, LILO, FILO, LIFO.

3. Serial Data communication and Parallel Data Communication.

**Conclusion:**

An introduction to the flip flop and its applications as well as data processing techniques from digital electronics is necessary to understand the working of microcontroller.

**II. Lab Exercises: Part A**

**Exercise No.2: Introduction to the simulator of 8051 Microcontroller.**

Get introduced with MCU8051 simulator.

**II. Lab Exercises: Part B**

**Exercise No.1:** Assembly Language programs Addition.

**Aim** : To write assembly language program for addition of two 8/16 bit numbers.

**Statement:** Add two 8/16 bit numbers and store result in register A.

**Algorithm:**

1. Load register A with given value.

1. Load register with given value.
	1. Add contents of register & register A.
	2. Store result.

**Program : 8 Bit**

MOV A, #24

MOV R0, # 23

ADD A, R0

RET

**Program : 16 Bit**

MOV R1, #13H

MOV R2, # 25H

MOV R3, #33H

MOV R4, # 45H

MOV A, R2

ADD A, R4

MOV 20H,A

MOV A, R1

ADD A, R3

MOV 21H,A

END

**Data:** Select 8 bit or 16 bit data.

**Result:**

**Conclusion:** Thus, the program executed for addition of two 8/16 bit numbers is successfully executed. The result of arithmetic operation can be stored in internal as well as external memory location.

**Exercise No.2:** Assembly Language programs Multiplication and division of 8 bit numbers.

**Aim:** To write assembly language program for multiplication/division of two 8 bit numbers.

**Statement:** Multiply/Divide two 8 bit numbers and store result.

**Algorithm:**

1. Load register A with given value.

 2. Load any other register with given value.

 3. Multiply/Divide contents of register & Accumulator.

 4. Store result.

**Program : Multiplication.**

MOV A, #51

MOV B, #05

MUL AB

MOV R0,A

END

**Program : Division.**

MOV A, #51

MOV B, #05

DIV AB

MOV R0,A

END

**Data:** Select 8 bit data for multiplication and division operation.

**Result:**

**Conclusion:** Thus, the program executed for Multiplication and Division of two 8 bit numbers is successfully executed. The separate instruction of multiplication and division is available for 8051 microcontroller which is not available for 8085 microprocessor.

**Exercise No.3:** Assembly language program of finding the average of five numbers.

**Aim:** To write assembly language program for the average of five numbers**.**

**Statement:** Five 8 bit numbers are stored at different register and their average is to be obtained.

**Algorithm:**

1. Load registers five numbers with given value.

 2. Perform addition of these five numbers.

 3. Divide contents with 5 and obtain average.

 4. Store result.

**Program:**

**MOV 60H,#06**

MOV 61H, #35

MOV 62H, #15

MOV 63H, #55

MOV 64H, #05

MOV R0, #60H

MOV R1, #05

MOV A, @R0

DEC R1

UP:INC R0

ADD A,@R0

DJNZ R1,UP

MOV B, #05

DIV AB

MOV 50H, A

END

**Data:** Select five 8 bit numbers data for addition and division operation.

**Result:**

**Conclusion:** Thus, the program executed for finding the average of five numbers. This program demonstrates that multiple arithmetic operations can be executed in a single program to perform complex arithmetic operation.

**Exercise No.4:** Assembly language program of finding the Largest/Smallest number.

**Aim:** To write assembly language program for the finding the Largest/Smallest number.

**Statement:** Compare two number and obtain largest/smallest number.

**Algorithm:**

1. Load two numbers with given value.

 2. Perform comparison of these numbers.

 3. Obtain largest/smallest number.

 4. Store result.

**Program:**

MOV 51H,#03

MOV 52H,#05

MOV R0, #51H

MOV A, @R0

INC R0

CLR C

SUBB A, @R0

JNC NEXT

SJMP LARGE

NEXT: DEC R0

LARGE: MOV 53H,@R0

END

**Program:**

MOV 51H,#03

MOV 52H,#05

MOV R0, #51H

MOV A, @R0

INC R0

CLR C

SUBB A, @R0

JC NEXT

SJMP SMALL

NEXT: DEC R0

LARGE: MOV 53H,@R0

END

**Data:** Select any two 8 bit numbers.

**Result:**

**Conclusion:** Thus, the program executed for finding the largest/smallest number. This program demonstrates that specific data parameter can be obtained by using conditional instructions.

**Exercise No.5:** Generation of time delay.

**Aim:** To write program for generation of time delay.

**Statement:** Generation of time delay.

**Algorithm:**

1. Load hex code of constant.

 2. Store hex code in a register and decrement it.

 3. Repeat the loop.

 4. Return to main program.

**Program:**

DELAY:

MOV R2,#25

UP:DEC R2

DJNZ R2,UP

RET

**Data:** Select decimal number and obtain its hex code.

**Result:**

**Conclusion:** Thus, the program executed for delay subroutine. This program demonstrates that by using register in combination large amount of time delay can be generated for interfacing operation.

**Exercise No.6:** LED Interfacing.

**Aim:** To write program for generating pattern using LED interfacing.

**Statement:** Interface LED to microcontroller and generate blinking pattern.

**Algorithm:**

1. Load two numbers with given value.

 2. Provide output to the LED.

 3. Obtain blinking pattern.

 4. Conclude program.

**Program:**

REP:

MOV A, #00

MOV P0, A

CALL SUB

MOV A, #255

MOV P0, A

CALL SUB

SJMP REP

SUB:

MOV R2, #02

UP: DEC R2

DJNZ R2, UP

RET

**Data:** Select number 00 and 255 for generating all LED ON/OFF pattern.

**Result:**

**Conclusion:** Thus, the program executed for LED blinking. This program demonstrates that by using numeric values and interfacing of LED different patterns of lighting can be generated.

**Exercise No7:** Seven Segment Display Interfacing.

**Aim:** To write program for representing number 0 to 9 on seven segment display.

**Statement:** Interface seven segment display to microcontroller and represent number.

**Algorithm:**

1. Load hex code of decimal number 0 to 9.

 2. Provide hex code to display unit and provide delay.

 3. Obtain numbers on display.

 4. Conclude program.

**Program:**

REP:

MOV A,#255

MOV P0,A

CALL SUB

MOV A,#2

MOV P0,A

CALL SUB

MOV A,#00

MOV P0,A

CALL SUB

MOV A,#72

MOV P0,A

CALL SUB

MOV A,#8

MOV P0,A

CALL SUB

SJMP REP

SUB:

MOV R2,#2

UP:DEC R2

DJNZ R2,UP

RET

**Data:** Select number 0 and 9 and obtain its hex code.

**Result:**

**Conclusion:** Thus, the program executed for seven segment interfacing. This program demonstrates that by using numeric values and its hex code alphanumeric data can be generated on seven segment display.

**Exercise No.8:** Stepper motor control using microcontroller.

**Aim:** To write program for controlling the operation of stepper motor.

**Statement:** Stepper motor interfacing.

**Algorithm:**

1. Load hex code of constant.

 2. Store hex code in a register and provide it to port .

 3. Call delay change hex code and repeat.

 4. Obtain stepper motor rotation.

**Program:**

UP:MOV A,#255

MOV P0,A

CALL SUB

RRC

JMP UP:

SUB:

MOV R2,#25

UP:DEC R2

DJNZ R2,UP

RET

**Data:** Select decimal number and obtain its hex code.

**Result:**

**Conclusion:** Thus, the program executed for stepper motor operation control. This program demonstrates that by using microcontroller interfacing rotation, speed as well as normal operation of stepper motor can be executed. Same is applicable to DC motor interfacing.

**Exercise No.9:** Generation of square wave using microcontroller.

**Aim:** To write program for generation of square wave using microcontroller.

**Statement:** Generation of signal.

**Algorithm:**

1. Load 00 in accumulator.

 2. Provide it to output port.

 3. Call delay and change the content of accumulator to 255.

 4. Provide it to output port and repeat from first step.

**Program:**

UP:MOV A,#00

MOV P0,A

CALL SUB

MOV A,#255

MOV P0,A

CALL SUB

JMP UP:

SUB:

MOV R2,#25

UP:DEC R2

DJNZ R2,UP

RET

**Data:** Select number 00 and 255 and use it in loop.

**Result:**

**Conclusion:** Thus, the program executed for GENERATION of square wave signal. By using ADC and DAC IC one can generate different types of analog signal as well as store the data of digital signal.

**II. Lab Exercises: Part C**

**Exercise:** Application of Microcontroller.

**Aim:** To build a simulation/Hardware based small project on application of Microcontroller.

**Statement:** Project Topic:

 1. Object Counter.

 2. LED Lighting.

 3. Speed control of DC Motor.

 4. Temperature sensing using Microcontroller.

5. Water tank cutoff and buzzer.

6. Display message on LCD display device.

7. Simple Automation of electronics equipment.

**Procedure:**

 1. Form a group of five students (4 group per practical batch)

 2. Select a project out of above topics.

 3. Purchase component/Simulate and execute programming.

 4. Obtain output.

**Result:**

**Conclusion:** Thus application of microcontroller studied practically by implementation.

**3. Quiz on the subject:**

Quiz should be conducted on tips in the laboratory, recent trends and subject knowledge of the subject. The quiz questions should be formulated such that questions are normally are from the scope outside of the books. However twisted questions and self formulated questions by the faculty can be asked but correctness of it is necessarily to be thoroughly checked before the conduction of the quiz.

Sample Questions:

Define Microcontroller.

1. Define stack, stack pointer.
2. Define Memory.
3. What is RAM? Is RAM a volatile memory?
4. What is ROM? Is ROM used to store the binary codes for the instructions or lookup table? Why?
5. What is the function of ‘Timing and control unit’ in Microcontroller?
6. Which are the different types of buses used in Microcontroller?
7. Explain fetching, decoding and execution operations of Microcontroller..
8. Explain the difference between PROM, EPROM AND EEPROM.
9. Explain Different Blocks Of Microcontroller..
10. How many data lines, address lines are present in 8051.
11. How many address lines are required to access 2MB of memory?
12. List the internal registers in 8051.Describe the primary function of each register.
13. Give the clock frequency of 8051 operating with each of following frequency crystals:6.25MHZ,6.144MHz,5MHz,4MHz
14. Give the format of Flag Register in 8051.Explain each flag.
15. What is the use of ALE signal?
16. What is the use of ‘clock out’ and ‘reset out’ signals of 8051?
17. Describe function of following pins in 8051:
	1. READY (2) ALE (3) IO/M’ (4) HOLD (5) RESET
18. List the instructions related to DMA operation in 8051.
19. List out different control signals used by 8051.
20. On power on reset, what is the content of PC ?
21. List the instructions related to serial operation in 8051.
22. List the different addressing modes of 8051.
23. Explain following instructions:
	1. PUSH 2)POP 3)CALL 4)RET
24. Explain 8255.

**4. Conduction of Viva-Voce Examinations:**

Teacher should conduct oral exams of the students with full preparation. Normally, the objective questions with guess should be avoided. To make it meaningful, the questions should be such that depth of the students in the subject is tested. Oral examinations are to be conducted in cordial environment amongst the teachers taking the examination. Teachers taking such examinations should not have ill thoughts about each other and courtesies should be offered to each other. Difference of opinion, if any, should be critically suppressed in front of the students.

**5. Evaluation and marking system:**

Basic honesty in the evaluation and marking system is absolutely essential and in the process impartial nature of the evaluator is required in the examination. It is a wrong approach to award the students by way of easy marking to get cheap popularity among the students, which they do not deserve. It is a primary responsibility of the teacher to see that right students who are really putting up lot of hard work with right kind of intelligence are correctly awarded.

The marking patterns should be justifiable to the students without any ambiguity and teacher should see that students are faced with just circumstances.

28