INTRODUCTION TO MASM

MASM is a short name used for Microsoft " MICRO ASSEMBLER", The MASM.EXE executable file executes all assembly language programs. Another executable file used in execution is LINK.EXE.

Every program execution involves three steps assembling, linking and debugging.

ASSEMBLING:-

In the process of assembling a file with the extension as **.asm** will be translated to object code in the form of machine code. Along with this offset addresses are calculated for the data variables in data segment and instructions in code segment.

Any sort of errors and warnings related to instructions, assembler directives are displayed on the screen after assembling process. If there are no errors in source code. .obj file is created and it is not in executable form.

LINKING:

In the liking process generated object file is converted to .exe file which is an executable file consisting the machine codes.

Another task of a linker is to convert the separately assembled files into one executable module.

DEBUGGING: -

Debug is a DOS program that allows user to view memory contents. User can enter hex codes of a program in memory and trace the program execution. DEBUG is simple to use and universally available in all computers. Debug always considers the data in hexadecimal format. Single step is one of the widely used commands in Debug, which allows user to view the effect of instruction on register and memory contents.

DEBUG COMMANDS: -

All Debug commands are single character commands.

A – Assemble – Convert mnemonics to hex codes

D – Display – Displays the memory contents. The type of memory has given by segment register. (DS, ES, SS, CS).

E – Enter data – Enter data or code in memory from a given specific location.

G – GO – To run an executable file.

N – Name – Assign name to a program.

P – Proceed – execute a set of related instructions.

Q – Quit – Exit from the debugging mode.

R – View register – Displays the contents of one or more registers.

T – Trace – The execution of a program instruction by instruction.

U – Disassemble – Convert hex code to mnemonics.

W – Write to Disk – Write a program to a disk.

Both lower case (or) uppercase of letters can be used as commands.

PROCEDURE FOR AN ASSEMBLY LANGUAGE PROGRAM EXECUTION

1. **Create a folder:** - Create an empty folder with your roll number (ex: ec423). In C – drive. Copy the two executable files MASM.exe and LINK.exe from MASM folder to your folder.
2. **GO TO DOS PROMT:** - Click on start, select RUN enter CMD in box and press enter key. Clear the path in C drives by entering >CD\ enter. Now c:\> will be displayed on the screen. Go to your folder by typing CD folder name and press enter. Then screen shows c: ec423>
3. **Entering a program:** - An assembly language program is entered and can be altered in a text editor. To enter a program use EDIT command along with filename. Ex: - c:\ec423>EDIT ADD8.ASM. A text editor window will be opened for program entry. Type the program and save the file. Exit from the window. A notepad editor can also be used for program entry but the file should be saved with the extension .asm.

4. **Assembling a program:** - MASM command along with filename is used in c prompt to assemble a file. Ex: - c:\ ec423 > MASM ADD.ASM

FILES CREATED IN ASSEMBLING PROCESS:-

1. file.obj:- objective file – Object file consists the object code for a source file. Which is of special machine symbols.
2. file.lst:- listing file – Listing file shows both source and objective files together which means that each instruction with its hex equivalent value displayed on the screen.Lst file will provide complete description of a program. To view this file type EDIT file.lst at c prompt.
3. file.crf – cross reference file this file consists the symbols used by the user for cross checking purpose. To skip the generation of these files place a semicolon in MASM command ex:- c:\ec423 >MASM ADD8.ASM; This step will directly shows the status of errors and warnings of the program and returns to c prompt. Ex:- C:ec423 > Any sort of syntax errors and warnings will be displayed on the screen along with line number and type of error/ warning. To rectify these go back to text editor with edit command and change accordingly.

LINKING A PROGRAM:-

Use link command to link an object file. To an executable file. Enter link filename.obj and press enter at c prompt. Ex:- ec423>LINK ADD8.OBJ.

Files created in link process.

1. .exe file – Executable file. This file consists machine code. which is understood by DEBUG in execution process.
2. .map file – mapping file. Map file provides the relative location and size of each segment used in the program. It also displays the errors related to segment declarations like "No stack segment".
3. .lib file – Library file- this file provides the information of any other associated files in same module.

To skip all these place a (;) is link command. EX: LINK ADD8.obj;

EXECUTING A PROGRAM

To execute a program use DEBUG command along with filename .exe in C prompt.

Ex: - DEBUG ADD8.exe;

DEBUG change the curser control from C prompt to (-) hyphen prompt.

Use GO command to execute a program

Ex: -g

This command executes the program and displays the contents of all internal registers available in 8086 microprocessor.

To verify the memory contents from data segment i.e inputs and outputs together use the command

Dds:0000 n

'n' indicates the number of locations to be displayed.

To quit from Debug mode use -Q

Curser returns to C prompt.