16.317: Microprocessor Systems Design I

Spring 2012

Lab 1: DOS commands and DEBUG software
Due **in class**, **Wednesday, 2/29/12**

Objective

In this lab, you will learn how to:

1. Execute basic DOS commands: Boot up DOS, edit/save/copy/move/delete a file.
2. Use the DEBUG software: Run/quit DEBUG software, show/modify register content, show/modify flags, dump memory contents, assemble and debug programs.

Part 1: DOS commands

This part of the lab will familiarize you with MS-DOS commands. Although DOS is a rather old operating system for PCs, it was widely used in 80's and early 90's. You will use DOS extensively in this course as you learn 80386 real-mode assembly language programming. You can do the lab using the DOS command-line console on Windows XP.

Commonly used DOS commands (reference: http://www.computerhope.com/dostop10.htm)

Command	Function
dir	The dir command allows you to see the available files in the current and/or parent directories **dir /p →** list the directory content and pause for long list
cd	cd (change directory) switches directories in MS-DOS. For example: **cd lab1 →** Change the directory to lab1 (assuming "lab1" is a sub-directory in the current directory). **cd .. →** Go up one directory level. For example, if the current directory is "C:\Users\John_Smith", "cd .." changes directories to "C:\Users".
edit	Edit allows a user to view, create and or modify their own files. Alt-F opens the editor menu; where you can save or open another file.
copy	Allows the user to copy one or more files to an alternate location without deleting the source file. For example, **copy file.asm newfile.asm** copies "file.asm" to "newfile.asm".
del	Deletes files from the computer. Example: **del file.asm**
move	Move files/directories from one location to another. For example, **move file.asm newfile.asm** renames "file.asm" as "newfile.asm". Can use to move file to new location without renaming it. For example, **move file.asm C:\Users\John_Smith\Desktop** would move "file.asm" from the current directory to the one specified.
ren	Used to rename files and directories. For example, **ren file.asm newfile.asm** functions similarly to the move example shown above.
mkdir	Used to create new directories. For example, **mkdir labs** will create a directory called "labs" in the current directory.

Create a directory based on your name (for example, "John_Smith"). Inside this directory, create a subdirectory called **lab1**. Inside **lab1**, you need to create a file named **test.asm**.

Part 2: Use DEBUG program

You will learn how to run DEBUG, a program that will allow you to assemble programs and execute them line by line in later labs. You will gain some basic familiarity with the DEBUG commands in this part of the lab, which will require you to complete some basic examples from Chapter 4 of the textbook.

DOS DEBUG commands: (reference: http://kb.iu.edu/data/afhs.html or p. 106 of the text)

Command	Function
?	Displays a list of debug commands
A	Assembles 8086/8087/8088 mnemonics
C	Compares two portions of memory
D	Displays the contents of a portion of memory
E	Enters data into memory starting at a specified address
F	Fills a range of memory with specified values
G	Runs the executable file that is in memory
H	Performs hexadecimal arithmetic
I (capital i)	Displays one byte value from a specified port
L	Loads the contents of a file or disk sectors into memory
M	Copies the contents of a block of memory
N	Specifies a file for an L or W command, or specifies the parameters for the file you are testing
O	Sends a single byte value to an output port
P	Executes a loop, a repeated string instruction, a software interrupt, or a subroutine
Q	Stops the Debug session
R	Displays or alters the contents of one or more registers
S	Searches a portion of memory for a specified pattern of one or more byte values
T	Executes one instruction and then displays the contents of all registers, the status of all flags, and the decoded form of the instruction that Debug will execute next.
U	Disassembles bytes and displays the corresponding source statements
W	Writes the file being tested to a disk
XA	Allocates expanded memory
XD	Deallocates expanded memory
XM	Maps expanded memory pages
XS	Displays the status of expanded memory

You need to practice DEBUG commands by reproducing examples from the textbook. Specifically, you need to practice the following command categories (see page 4 for details of what commands need to be practiced from each example, if you do not have the textbook):

Register and flag commands
Example 4.9, 4.10, 4.11 (page 109-111)

Memory
Example 4.14, 4.15, 4.16, 4.17, 4.18, 4.19 (page 112-121)

I/O
Example 4.20 (page 122)

Hexadecimal
Example 4.22 (page 124)

Part 2 check off

To check off, demonstrate one of the experiments (4.9, 4.10, 4.11, 4.17) to an instructor. The TA will record your name.

Examples from text

Example 4.9 (p. 109) Verify the initialized state of the 80386 DX by examining the contents of its registers with the register command, "R".

Example 4.10 (p. 109) Issue commands to the debugger on the PC/AT that will cause the value in BX to be modified to $FF00_{16}$, and then verify that this new value is loaded into BX.

Example 4.11 (p. 111) Use the register command to set the parity flag to even parity. Verify that the flag has been changed.

Example 4.14 (p. 114) Issue a dump command that will display the contents of the 32 bytes of memory that are located at offsets 0300_{16} through $031F_{16}$ in the current data segment.

Example 4.15 (p. 115) Use the dump command to examine the 16 bytes of memory just below the top of the stack.

Example 4.16 (p. 117) Start a data entry sequence by examining the contents of address DS:100 and then, without entering new data, depress the – key. What happens?

Example 4.17 (p. 118) Initialize all storage locations in the block of memory from DS:120 through DS:13F with the value 33_{16} and the block of storage locations from DS:140 through DS:15F with the value 44_{16}. Verify that the contents of these ranges of memory are correctly modified.

Example 4.18 (p. 119) Fill each storage location in the block of memory from address DS:100 through DS:11F with the value 11_{16}. Then, copy this block of data to a destination block starting at DS:160. Verify that the block move is correctly done.

Example 4.19 (p. 121) Perform a search of the block of data from addresses DS:100 through DS:17F to determine which memory locations contain 33_{16}.

Example 4.20 (p. 122) Write a command that will display the byte contents of the input port at I/O address $00FE_{16}$.

Example 4.22 (p. 124) If a byte of data is located at physical address $02A34_{16}$ and the data segment register contains 0150_{16}, what value must be loaded into the source index register such that DS:SI points to the byte storage location?

16.317: Microprocessor Systems Design I

Spring 2012

Lab 1: DOS commands and DEBUG software

You must include this sheet as the cover page of your lab report. Fill in your name and your partner's name (if applicable). The table below provides the grading rubric for this assignment, as well as space for an instructor to record your grade for each section.

Student name: _____ **Student ID #** _____

Partner's name: _____

Grading rubric

Item	Description	Max points	Actual points
Check off	Part 1 (demonstrate directory creation) Part 2 (demonstrate at least one of the examples listed below to an instructor)	30	
Lab report organization	Your report contains all required sections and is of sufficient length.	10	
Example 4.9	Include a print-out, screen capture, or list of command(s) practiced to show completion of this example.	15	
Example 4.10	Include a print-out, screen capture, or list of command(s) practiced to show completion of this example.	15	
Example 4.11	Include a print-out, screen capture, or list of command(s) practiced to show completion of this example.	15	
Example 4.17	Include a print-out, screen capture, or list of command(s) practiced to show completion of this example.	15	
	TOTAL	100	