16.317: Microprocessor Systems Design I

Spring 2012

Lab 2: Assembling and executing instructions with DEBUG software
Due **Wednesday, 3/28/12**

Introduction

In this lab, you will learn how to:

1. Assemble instructions into the memory of the PC.
2. Execute an instruction to determine the operation it performs.
3. Verify the operation of data transfer, arithmetic, compare, and jump instructions.

Remember that each student must generate an individual report that follows the guidelines listed in the "Lab Report Format Guidelines", and that the last page of this assignment contains a cover page each student must use as the first page of that report.

Part 1: Practice

Recall that for a reference on DOS DEBUG commands, check http://kb.iu.edu/data/afhs.html or p. 106 of the text.

To practice DEBUG commands used to load programs, initialize registers and memory, we recommend completing Example 4.26 from p. 136 of the textbook. A description of this exercise is given at the end of this assignment. **Note that you do not have to be checked off on this part of the assignment.**

Part 2: Testing Instructions

In order to successfully be checked off for this assignment, generate at least one screen capture per instruction category. Each category contains several questions to be answered as well; each student must answer these questions in his or her lab report.

Category 1: Data transfer I

1. Use DEBUG to assemble the following instructions:
    a. `MOV  AX, BX`
    b. `MOV  AX, AAAA`
    c. `MOV  AX, [BX]`
    d. `MOV  AX, [4]`
    e. `MOV  AX, [BX+SI]`
    f. `MOV  AX, [SI+4]`
    g. `MOV  AX, [BX+SI+4]`

2. Initialize the internal registers of the 80386 as follows:
    - `(AX) = 0000H`
    - `(BX) = 0001H`
    - `(CX) = 0002H`
    - `(DX) = 0003H`
    - `(SI) = 0010H`
    - `(DI) = 0020H`
    - `(BP) = 0030H`

    Verify these registers are correctly initialized by displaying the new register contents.

3. Fill all memory locations in the range DS:00 through DS:1F with 00H and initialize the word storage locations listed below:
    - `(DS:0001H) = BBBBH`
    - `(DS:0004H) = CCCCH`
    - `(DS:0011H) = DDDDH`
    - `(DS:0014H) = EEEEH`
    - `(DS:0016H) = FFFFH`

4. Trace the execution of the instructions you assembled in step 1.

***Question 1.1:* Explain the execution of each instruction (a-g), including the addressing mode used, physical address accessed, and the value stored in AX.**

Category 2: Data transfer II

1. Assemble the instruction MOV SI, [0ABC] to memory at address CS:100 and verify loading of the instruction.

Question 2.1: ***How many bytes does this instruction take up?***

2. Initialize the word of memory starting at DS:0ABC with the value FFFFH.
3. Clear the SI register and verify that operation by displaying its content.
4. Trace the execution of this instruction.

Question 2.2: ***Describe the operation of this MOV instruction.***

5. Assemble the instruction MOV WORD PTR [SI], ABCD into memory at address CS:100 and then verify loading of the instruction.

Question 2.3: ***How many bytes does this instruction take up?***

6. Initialize the SI register with the value 0ABCH.
7. Clear the word of memory starting at DS:0ABC.
8. Trace the execution of this instruction.

Question 2.4: ***Describe the operation of this MOV instruction.***

Category 3: Arithmetic instructions

1. Assemble the instruction ADC AX, [0ABC] to memory at address CS:100 and verify loading of the instruction.

Question 3.1: ***How many bytes does the instruction take up?***

2. Initialize the word of memory starting at DS:0ABC with value FFFFH.
3. Initialize AX with the value 0001H. Verify by displaying register contents.
4. Clear the carry flag.
5. Trace the execution of this instruction.

Question 3.2: ***Describe the operation performed by this ADC instruction.***

Question 3.3: ***Does this instruction produce a carry out from the last bit?***

Category 4: Flag control instructions

1. Assemble the following instruction sequence:
    a. `LAHF`
    b. `MOV  BH, A`
    c. `AND  BH, 1F`
    d. `AND  AH, E0`
    e. `MOV  [200], BH`
    f. `SAHF`

Question 4.1: ***How many bytes do these instructions take?***

2. Initialize the byte of memory starting at DS:200 with the value 00H.

3. Clear register AX and BX.

4. Display the current state of the flags and ensure that the flags equal NG, ZR, AC, PE, and CY.

5. Trace the execution of this sequence.

Question 4.2: ***Describe the operation of each instruction. In particular, note what value is initially read out of the flags register, what value is saved in memory, and what value is reloaded into the flags register.***

Category 5: Compare instructions

1. Assemble the following instruction sequence into memory at address CS:100 and then verify loading of the instructions:
    a. `MOV  BX, 1111`
    b. `OR   AX, BBBB`
    c. `CMP  BX, AX`

2. Clear register AX and BX.

3. Display the current state of the flags.

4. Trace the execution of this sequence.

Question 5.1: ***What operation does each instruction perform?***

Question 5.2: ***What are the status flags before and after the compare instruction is executed?***

Category 6: Jump instructions

1. Download the files L5P3.LST and L5P3.EXE from the course web page into a folder.
2. Open the listing file (L5P3.LST).

***Question 6.1:** What are the offsets from CS for the first instruction (PUSH DS) and the last instruction (RET)?*

3. Use the DEBUG program to load the run module L5P3.EXE.
4. Verify loading of the program by disassembling the contents of the current code segment for the offset range found in step 2. (Use the U ("Unassemble") command.)
5. Execute the program according to the instructions that follow:
    a. GO (G) from address CS:00 to CS:5.
    b. Load the number for which the factorial is to be calculated (N=3) into DX.
    c. Clear the memory location DS:0000 for the value of the factorial (FACT).
    d. GO from address CS:5 to CS:10
    e. Execute the JZ instruction using a TRACE (T) command.
    f. GO from address CS:12 to CS:16.
    g. Execute the JMP instruction with a TRACE command.
    h. GO from address CS:E to CS:10.
    i. Execute the JZ instruction with a TRACE command.
    j. GO from address CS:12 to CS:16.
    k. Execute the JMP instruction with a TRACE command.
    l. GO from address CS:E to CS:10.
    m. Execute the JZ instruction with a TRACE command.
    n. GO from address CS:12 to CS:16.
    o. Execute the JMP instruction with a TRACE command.
    p. GO from address CS:E to CS:10.
    q. Execute the JZ instruction with a TRACE command.
    r. GO to CS:1B.
    s. Display the value stored for FACT in memory.

***Question 6.3:** For each jump instruction in the sequence, describe whether or not the jump was taken and, if taken, what the address of the next instruction to be executed is.*

***Question 6.4:** List all values in AL and the instructions that change that register. At what address is the final value in AL stored in memory as FACT?*

Part 2 check off

To check off, get an instructor's signature on each screen print-out you obtained (one from each section). Attach the signed print-outs to your report. If you work in a group, make sure each person has a copy of each print-out.

Appendix: Examples from text (4.24, p. 127; 4.26, p. 136)

Step 1: Complete Example 4.24, which enters a machine code program into memory and saves it on disk.

a. Enter the following all on one line to load the machine code for the program:
```
E CS:100 B8 00 20 8E D8 BE 0 01 BF 20 01 B9 10 0 8A 24 88
25 46 47 49 75 F7 90
```

b. Verify that the code has been correctly loaded using the command: `D CS:100 117`

c. Disassemble the code to see the actual instructions using: `U CS:100 117`

d. Name this file BLK.1, specifying the full directory path where it is to be stored; for example: `N C:\MGEIGER\BLK.1`

- **IMPORTANT NOTE:** Choose your file location wisely, as writing the file into the wrong location could unintentionally damage other files.

e. To ensure that the file is written correctly, the register pair BX/CX must hold the total number of bytes. Since this program runs from CS:100 to CS:117, it holds 18_{16} bytes. Therefore, initialize CX to 18 and BX to 0 using the following commands:

- `R CX` → Enter 18 after the current value of CX is displayed
- `R BX` → Enter 0 after the current value of BX is displayed

f. Write the file to the specified location: `W CS:100`

g. Exit the DEBUG program; then, navigate to the directory holding the file BLK.1 and rename it BLK.EXE using the DOS REN command.

Step 2: Complete example 4.26, which loads the executable program and executes new commands as follows:

a. Start DEBUG and load the executable into address CS:200:

- N <path>\BLK.EXE
- L CS:200

b. Set DS to the value 2000_{16}, memory from DS:100 through DS:10F as FF_{16}, and memory from DS:120 through DS:12F as 00_{16}. Verify memory was loaded correctly.

c. Reload DS with the value 1342_{16}. Display the state of all registers to verify the result of this operation.

d. Display the assembly language version of the program from CS:200 through CS:217.

e. Use a GO (G) command to execute instructions through address CS:20E. What changes have occurred?

f. Now execute through address CS:215. What changes are found in the blocks of data?

g. Now execute through address CS:217. What changes are found in the blocks of data?

16.317: Microprocessor Systems Design I

Spring 2012

Lab 2: Assembling and executing instructions with 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	Generate at least one signed printout for each section of the lab	40	
Data transfer I/II	Include a print-out or screen capture of the section results. Answer questions correctly.	20	
Arithmetic	Include a print-out or screen capture of the section results. Answer questions correctly.	10	
Flag control	Include a print-out or screen capture of the section results. Answer questions correctly.	10	
Compare	Include a print-out or screen capture of the section results. Answer questions correctly.	10	
Jump	Include a print-out or screen capture of the section results. Answer questions correctly.	10	
	TOTAL	100	