16.317: Microprocessor Systems Design I

Fall 2015

Homework 1

Due **2:00 PM**, **Friday, 9/18/15**

Notes:

- While typed solutions are preferred, handwritten solutions are acceptable.
- Any electronic submission must be in a single file. Archive files will not be accepted.
- Electronic submissions should be e-mailed to Dr. Geiger at Michael_Geiger@uml.edu.
- This assignment is worth 100 points.

1. (50 points) Given each of the binary or hexadecimal number below, determine what the decimal value is if the number is (i) an unsigned integer, and (ii) a signed integer. Note that, in some cases, your answers for both will be the same.

a. 01011101_2
b. 10100110_2
c. 9Ch (or 0x9C—recall that, in x86 assembly notation, the "h" at the end of a number signifies that the previous value is in hexadecimal)
d. 4B93h
e. FEEDh

See the next page for Question 2.

2. (50 points) Assume the contents of memory are shown below. All values are in hexadecimal. The table shows four bytes per line; the given address is the starting address of each line.

   Each block in the table contains a single byte, with the low and high bytes per line indicated as shown. Each byte has its own address, so the byte at address 20590h is 09h, address 20591h is 12h, address 20592h is 15h, and address 20593h is 20h.

   You should assume all multi-byte values are stored in little-endian format.

Address	Lo			Hi
20590h	09	12	15	20
20594h	62	AB	DD	EF
20598h	11	3C	77	91
2059Ch	CF	06	48	55
205A0h	78	6D	72	00
205A4h	B3	46	13	17
205A8h	39	30	C7	B9

Furthermore, assume the following initial register values:

- EBX = 000205A0h
- ECX = FFFFFFF4h
- ESI = 00020590h
- EDI = 00000005h

a. (10 points) In class, we discussed how to determine if words or double-words (in an x86 processor) are aligned. Newer Intel processors support quad-words that contain 64 bits of data. Determine the starting addresses of all aligned quad-words shown in the diagram above, and explain why those addresses are aligned.

b. (10 points) What is the result of the instruction MOV AL, [2059Ah]? What is the decimal value of the data transferred in this instruction?

c. (10 points) What is the result of the instruction MOV DX, [ESI+001Ah]? Is this memory access aligned?

d. (10 points) What is the result of the instruction MOV EDX, [EBX+ECX]? (Hint: treat ECX as a signed, two's complement integer and check its decimal value.) Is this memory access aligned?

e. (10 points) What is the result of the instruction MOV [ESI+2*EDI], AX, if EAX = 3399FFAAh? Is this memory access aligned?