COE 205 Computer Organization & Assembly Language – Fall 2004

Assignment 1 Solution

Q1. Describe the functionality of the Program Counter (PC), the Instruction Register (IR), the Memory Address Register (MAR), and the Memory Data Register (MDR).

A1. PC: A register that holds the address of the next instruction to be fetched.

IR: Holds the instruction that was fetched.

MAR: Holds the address of the variable to be read or written to memory

MDR: Holds the value of the variable that is read or written to memory

Q2. List and briefly describe the tools used by the assembly language programmer.

A2. Assembler: Is a program that translates programs written in assembly language to object files in machine language.

Linker: Is a program that links together object files in machine language with functions defined in libraries. The output of a linker is an executable file that can be loaded into the memory of the computer and can be executed.

Debugger: Is used to monitor the execution of a program, by executing it either step by step, or by using breakpoints. The content of memory and registers can be examined and modified for testing purposes.

Q3. a) Define the term *bus*.

b) Why are buses used in computers?

A3. a) Bus: Is a group of wires used to interconnect the different components of a computer.

b) Buses are used to transfer data between the processor and memory, between the processor and the I/O devices, and between the I/O devices and memory.

Q4. Represent the following numbers in binary, octal, and hexadecimal.

a) 2345.5

b) 149.625

A4. a) $2345.5 = (100100101001.1)_2 = (4451.4)_8 = (929.8)_{16}$

b) $149.625 = (10010101.101)_2 = (225.5)_8 = (95.A)_{16}$

Q5. Perform the following arithmetic operations using the designated bases and verify your result by converting the numbers and performing the operation in decimal:

a) $(10110110)_2 + (10111)_2$

b) $(713)_8 + (167)_8$

c) $(C0E)_{16} + (13B)_{16}$

A5. **a)** **Carry** 11 11

10110110_2 = 182

\+ 10111_2 = 23

11001101_2 = 205

b) **Carry** 111

713_8 = 459

\+ 167_8 = 119

1102_8 = 578

c) **Carry** 1

$C0E_{16}$ = 3086

\+ $13B_{16}$ = 315

$D49_{16}$ = 3401

Q6. Express the following numbers in sign-magnitude, 1's complement, and 2's complement notations, using 8-bit representation:

a) – 101

b) – 45

A6. **a) – 101 = (11100101)$_{\text{sign-magnitude}}$ = (10011010)$_{\text{1's complement}}$ = (10011011)$_{\text{2's complement}}$**

b) – 45 = (10101101)$_{\text{sign-magnitude}}$ = (11010010)$_{\text{1's complement}}$ = (11010011)$_{\text{2's complement}}$

Q7. Perform the following operations twice, once for a sign-magnitude notation and once for 2's complement notation, assuming 4-bit representation of numbers. Indicate in your answer when an overflow occurs:

a) 0101 + 1111

b) 1011 – 0111

A7. **In sign-magnitude:**

a) 0101+1111 = 0101–0111 = –(0111–0101) because 0111 is larger = –0010 = 1010

b) 1011 – 0111 = – 0011 – 0111 = – (0011+0111) = – 1010 = 0010 (Overflow)

In 2's complement:

a) 0101+1111 = 0100 (There is a carry out, BUT NO Overflow)

b) 1011 – 0111 = 1011 + 1001 = 0100 (There is a carry out AND Overflow)

Q8. Using 8-bit registers, give the following in both binary and decimal:

a) The maximum unsigned number that can be stored.

b) The smallest negative number and the largest positive number that can be stored using sign-magnitude notation.

c) The smallest negative number and the largest positive number that can be stored using 2's complement notation.

A8. a) maximum unsigned = $(11111111)_2$ = 255

b) smallest negative using sign-magnitude = $(11111111)_2$ = –127

largest positive using sign-magnitude = $(01111111)_2$ = +127

c) smallest negative using 2's complement = $(10000000)_2$ = –128

largest positive using 2's complement = $(01111111)_2$ = +127

Q9. If you type the string "I Love COE 205" on your keyboard, what byte sequence is sent to the computer using ASCII codes, with the 8th bit being an even parity bit?

A9. I = 49h, Space = 20h, L = 4Ch, o = 6Fh, v = 76h, e = 65h, C = 43h, O = 4Fh,

E = 45h , 2 = 32h, 0 = 30h, 5 = 35h

With the 8th bit being an even parity bit the byte sequence becomes:

C9h, A0h, CCh, 6Fh, F6h, 65h, A0h, C3h, CFh, C5h, A0h, B2h, 30h, 35h.