Name: Id#

COE 301/ICS 233, Term 171

Computer Architecture & Assembly Language

Quiz# 3 Solution

 Date: Tuesday, Nov. 7, 2017

## **Q1. [4 Points]** Given that **Multiplicand=1001** and **Multiplier=1011** are signed 2’s complement numbers, show the **signed** multiplication of **Multiplicand** by **Multiplier**. The result of the multiplication should be an 8-bit **signed** number in HI and LO registers. Show the steps of your work.

|  |  |  |  |
| --- | --- | --- | --- |
|  **Iteration** | **Multiplicand** | **Sign** | **Product = HI,LO** |
| 0 | Initialize | **1001** |  |  **0000 1011** |
| 1 | LO[0] = 1 => ADD |  | **1** |  **1001 1011** |
| Shift Product = (HI, LO) right 1 bit | **1001** |  |  **1100 1101** |
| 2 | LO[0] = 1 => ADD |  | **1** |  **0101 1101** |
| Shift Product = (HI, LO) right 1 bit | **1001** |  |  **1010 1110** |
| 3 | LO[0] = 0 => Do nothing |  | **1** |  **1010 1110** |
| Shift Product = (HI, LO) right 1 bit | **1001** |  |  **1101 0111** |
| 4 | LO[0] = 1 => SUB (ADD 2's compl) | **0111** | **0** |  **0100 0111** |
| Shift Product = (HI, LO) right 1 bit |  |  |  **0010 0011** |

## **Q2. [6 Points]** Given that **Dividend=0111** and **Divisor=1101** are signed 2’s complement numbers, show the **signed** division of **Dividend** by **Divisor**. The result of division should be stored in the Remainder and Quotient registers. Show the steps of your work, and show the final result.

## Since the Divisor is negative, we take its 2’s complement ⇒ Divisor = 0011

 Sign of Quotient = negative, Sign of Remainder = positive

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Remainder (HI)** | **Quotient (LO)** | **Divisor** | **Difference** |
| 0 | Initialize | **0000** | **0111** | **0011** |  |
| 1 | 1: SLL, Difference | **0000** | **1110** | **0011** | **1101** |
| 2: Diff < 0 => Do Nothing | **0000** | **1110** | **0011** |  |
| 2 | 1: SLL, Difference | **0001** | **1100** | **0011** | **1110** |
| 2: Diff < 0 => Do Nothing | **0001** | **1100** | **0011** |  |
| 3 | 1: SLL, Difference | **0011** | **1000** | **0011** | **0000** |
| 2: Rem = Diff, set lsb Quotient | **0000** | **1001** | **0011** |  |
| 4 | 1: SLL, Difference | **0001** | **0010** | **0011** | **1110** |
| 2: Diff < 0 => Do Nothing | **0001** | **0010** | **0011** |  |

**Quotient =** **1110 Remainder =** **0001**