NAME / ID#

**Quiz 2 Fall 2019**

**Odd**

**Question 1 (20 points)** 7-bit registers are used in this question.

(a) **Convert the following two signed numbers to binary, employing the 2's complement representation:**

**A = (- 31)$_{10}$**

A's magnitude |A| = 31$_{10}$ = 16 + 8 + 4 + 2 + 1, represented on 7 bits =

=> A = -31 = 2's complement (001 1111) = 110 0001

| sgn | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
|---|---|---|---|---|---|---|
| **0** | 0 | 1 | 1 | 1 | 1 | 1 |

**B = (+ 63)$_{10}$** = 011 1111

**OR:**

| 31 | 2 |
|---|---|
| 15 | 1 |
| 7 | 1 |
| 3 | 1 |
| 1 | 1 |
| 0 | 1 |
| 0 | 0 |

| | 63 | 2 |
|---|---|---|
| $2^0$ | 31 | 1 |
| $2^1$ | 15 | 1 |
| $2^2$ | 7 | 1 |
| $2^3$ | 3 | 1 |
| $2^4$ | 1 | 1 |
| $2^5$ | 0 | 1 |

(b) **Find the 2's complement of the signed binary number B, and** verify your calculations by converting your result to decimal**.**

*2's complement of B* (i.e., -B) = *2's complement of* (011 1111) = 100 0001, which is a negative number, say **-x**.

To convert **–x** to decimal, we have to find the magnitude *x* = |**-x**| of this number (**-x**) by complementing it:

*x* = - (**-x**) = *2's complement of* (**-x**) = *2's complement of* (100 0001) = 011 1111 = +63

⇨ *2's complement of B* in decimal = **-x** = -63, => the result 100 0001 is **correct**, because its decimal value -63 = -(-63)=-*B*; which makes sense, since by complementing *B*=+63, we negate its value to -63;

(c) **Perform the following arithmetic operations in signed 2's complement representation, considering that a 7-bit adder is used to perform these operations; show the operations and the results (including intermediary steps), both in binary and, for verification, in decimal.**

**S=A+B** = (- 31)$_{10}$ + (63)$_{10}$

| Operations for verification in decimal, but not converted from registers binary | |
|---|---|
| - 31 | |
| +63 | |
| +32 | |

| Carry: | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|
| A → | **1** | **1** | 0 | 0 | 0 | 0 | 1 |
| B → | **0** | 1 | 1 | 1 | 1 | 1 | 1 |
| S → | **0** | 1 | 0 | 0 | 0 | 0 | 0 |

The result S = **010 0100 converted to decimal** (S = +32$_{10}$) coincides with the result of operation performed in decimal (+32), and, which verifies S.

**Comment if 7 bits would be enough to correctly carry out that operation? Justify your answer**

All 3 numbers A, B and S ⊂ [-64, +63], which is the range of signed numbers with 7 bits, => no overflow ➔ 7 bits are sufficient to perform S=A+B with these numbers.

**D = A - B** = - ( 31)$_{10}$ - (63)$_{10}$ = (-31)$_{10}$ + (-63)$_{10}$

| Operations for verification in decimal, but not converted from registers binary | |
|---|---|
| - 63+ | |
| - 31 | |
| -94 | |

| Carry: | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|
| - B → | **1** | 0 | 0 | 0 | 0 | 0 | 1 |
| A → | **1** | 1 | 0 | 0 | 0 | 0 | 1 |
| D → | **0** | 1 | 0 | 0 | 0 | 1 | 0 |

**Comment at each operation if 7 bits would be enough to correctly carry out that operation?**

The result D=**0**10 0010 is a positive number $D_{10} = 2^5 + 2^1 = 34$, which, of course, is wrong ≠ -94.

-94 is out of range [-64, +63], so 7 bits are not sufficient to perform D=A-B with these numbers.

**Question 2 (5 points)**

An 8-bit register contains the binary value 00011101. What is the BCD representation of this number?

| | | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ | |
|---|---|---|---|---|---|---|---|---|---|
| Binary: | **0** | 0 | 0 | 1 | 1 | 1 | 0 | 1 | Decimal: 16+8+4+1= (2 9) $_{10}$ |
| | | | | | | | | | BCD: 0010 1001 |