CSci370 Computer Architecture: Homework 2 (double-sided)

Due date: On or before Monday, March 30, 2020 in class
Absolutely no copying others' works 6 **Name:** Derek Tsom

- There are four algorithms discussed for multiplication and division. Make sure you are using the correct ones for the two questions below.
- The purpose of homeworks is for students to practice for the exams without others' help, so the penalty of mistakes will be minor.
- Without practicing for the exams properly, students would not be able to do well on the exams.

---

1. **(Refined multiplication: 50%)** Using a table similar to that shown in the Slide 8.6, calculate the product of the octal unsigned 6-bit integers 65_8 (or 110101_2) and 53_8 (or 101011_2) using the hardware and ~~4~~7 algorithm described in the figures of Slide 8.6. You should show the content of each register on each step.

Iteration		Multiplicand	Carry	Product = HILO
0	Initialize	110101		000000 101011
	Add or do nothing		0	110101 101011
1	shift	110101		011010 110101
			1	001111 110101
	Add or nothing			
2	shift	110101		100111 111010
			0	100111 111010
	Add or not			010011 111101
3	shift	110101		
			1	001000 111101
	Add or not			100100 011110
4	shift	110101		
			0	100100 011110
	Add or not			010010 001111
5	shift	110101		
			1	000111 001111
	Add or not			100011 100111
6	shift	110101		

$100011\ 100111_2 = 4347_8 = 2279_{10}$

difference = remender - divisor

2. **(First-version division: 50%)** Using a table similar to that shown in the Slide 8.10, calculate the octal unsigned 6-bit integer 65_8 (or 110101_2) divided by another octal unsigned 6-bit integer 16_8 (or 001110_2) using the hardware and algorithm described in the figures of Slide 8.10. You should show the content of each register on each step.

†Note that you have to actually show the differences in the procedures, not just the signs.

		Remainder	Divisor	Difference	Quotient
0	Initialize	000000 110101	001110 000000		000000
1	Shift and subtract	000000 110101	000111 000000	111001 110101	000000
	Assign & set or not	000000 110101			000000
2	Shift and subtract	000000 110101	000011 100000	111011 010101	000000
	assign & set or not	000000 110101			000000
3	Shift and sub	000000 110101	000001 110000	111001 110101	000000
	assign & set or not	000000 110101			000000
4	Shift and Sub	000000 110101	000000 111000	111111 111101	000000
	assign & set or not	000000 110101			000000
5	Shift and sub	000000 110101	000000 011100	000000 011001	000000
	assign & set or not	000000 011001			000001
6	Shift and Sub	000000 110011	000000 001110	000000 001011	000010
	assign and set or not	000000 001011			000011

53_{10} remainder 11_{10} = 000011_2 remainder 000000001011_2

53
÷ 14