Київський національний університет імені Тараса Шевченка факультет радіофізики, електроніки та комп'ютерних систем

Лабораторна робота № 2 з курсу: «комп'ютерні системи»

з теми: «Арифметичні операції над двійковими числами»

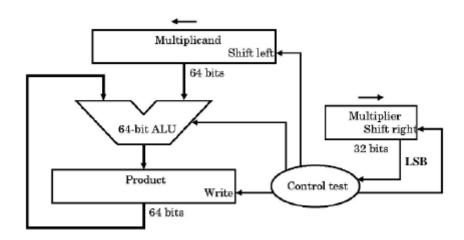
Роботу виконав: студент 3 курсу КІ групи СА Бузанов Дмитро Дмитрович https://github.com/DmytroBuzanov/CS labs/tr ee/master/lab 2

Теоретичні відомості

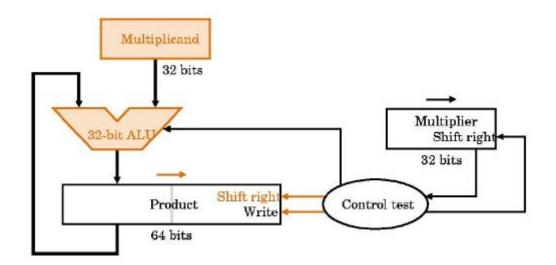
(згідно отриманого варіанту)

1. Множення двійкових чисел

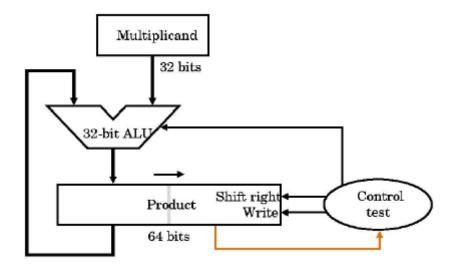
а) Множення як є



b) Зсув результату вправо

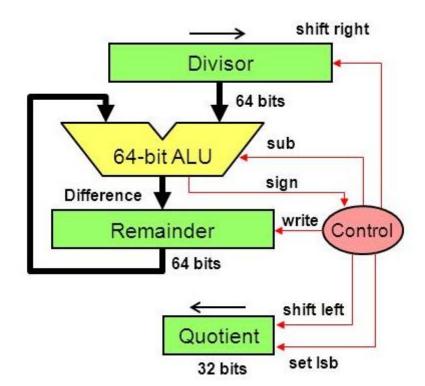


с) Множник в правій частині регістру

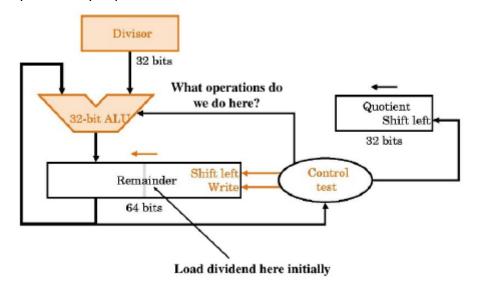


2. Ділення двійкових чисел

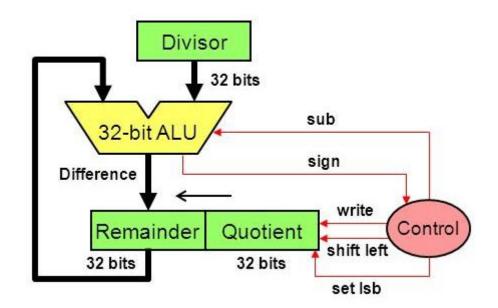
а) Ділення як є



b) Зсув залишку вправо



с) Частка та залишок в одному регістрі



Хід роботи

Варіант: а c b

1. Множення двійкових чисел

Випадковим чином обираємо числа 41 та 19

```
------ RESTART: D:\Lab2 CS\test.py ------
Multiplication check
al = 41
bl = 19
Multiplier: 000000000000000000000000000011
Multiplier LSB=1: adding multiplicand to product
Shifting multiplicand to left: 0000000000000000000000001010010
Multiplier LSB=1: adding multiplicand to product
Shifting multiplicand to left: 0000000000000000000000010100100
Shifting multiplicand to left: 00000000000000000000000101001000
Shifting multiplicand to left: 0000000000000000000001010010000
Multiplier LSB=1: adding multiplicand to product
Shifting multiplicand to left: 0000000000000000000010100100000
Shifting multiplicand to left: 000000000000000000101001000000
Shifting multiplicand to left: 0000000000000000001010010000000
Shifting multiplicand to left: 000000000000000010100100000000
```

```
Shifting multiplicand to left: 000000000000000101001000000000
Shifting multiplicand to left: 00000000000000101001000000000
Shifting multiplicand to left: 00000000000001010010000000000
Shifting multiplicand to left: 00000000000010100100000000000
Shifting multiplicand to left: 00000000000101001000000000000
```

Result: 779

Expected result: 779

2. Ділення двійкових чисел

Випадковим чином обираємо числа 55 та 18

```
Division check
a2 = 55
b2 = 18
Dividend: 0000000000000000000000000110111
Divisor is greater then dividend
The last quotient bit is set to 0
Divisor is greater then dividend
The last quotient bit is set to 0
Divisor is greater then dividend
The last quotient bit is set to 0
Divisor is greater then dividend
The last quotient bit is set to 0
Divisor is greater then dividend
The last quotient bit is set to 0
Divisor is greater then dividend
The last quotient bit is set to 0
```

Divisor is greater then dividend The last quotient bit is set to 0

Divisor is greater then dividend The last quotient bit is set to $\mathbf{0}$

Divisor is greater then dividend The last quotient bit is set to 0

Divisor is greater then dividend The last quotient bit is set to $\mathbf{0}$

Divisor is less then dividend Last bit in quotient is 1 Shifting register to left

Divisor is less then dividend Last bit in quotient is 1 Shifting register to left

Result: 3

Expected result: 3

3. Робота з IEEE 754 Floating Point (Представити лише ключові кроки при виконанні операцій)

```
Enter first number:
Enter second number:
Mantissa multiplication:
            00100000000000000000000000
Mantissa1
Mantissa2 00000000000000000000000000000
Normalization is not needed:
01000000000000000000000
Sign:
\theta XOR \theta = \theta
Exponent:
            10000001 ( 129 )
exp1
exp2
            10000010 ( 130 )
            - 127 + 0
           = 10000100 ( 132 )
Result: 0100001000100000000000000000000000 ( 40 )
```

Висновки: під час виконання лабораторної роботи було досліджено алгоритми, що використовуються в мікропроцесорах для множення та ділення цілих чисел та підходи до роботи з дійсними числам, а саме розглянуто випадки множення/ділення як є, зсуву результату вправо, множника/частки та залишку в правій частині та в одному регістрі відповідно та підкріплено практично шляхом написання відповідної програми