МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

«Вятский государственный университет»

(ФГБОУ ВПО «ВятГУ»)

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Отчет

Лабораторная работа №1

«Моделирование операционного автомата в САПР QUARTUS»

по дисциплине «Теория автоматов»

Выполнил студент группы ВМ-22 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Куваев Алексей.С./

Проверил преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Агалаков Е.В./

Киров 2012

*1. Постановка задачи*

Разработать ОА, реализующий операцию умножения чисел с плавающей запятой в дополнительном коде 2 способом с характеристикой простой коррекцией. Собрать схему разработанного автомата в САПР Quartus и убедиться в работоспособности автомата.

*2.Численный пример*

А = -99= 0,10111 00110101112

В = -2,3125= 1,00001 10011101112

1) 1  0 = 1

2) Сложим характеристики:

1,0111

0,0001

10,1000

ПРС и ПМР не возникло.

3) Перемножим мантиссы:

|  |  |  |  |
| --- | --- | --- | --- |
| Множитель | Множимое | Сумма ЧП | Пояснения |
| *0011010111* | 0000000000 1001110111 | 0000000000 0000000000  0000000000 1001110111  0000000000 1001110111 | Сложение, сдвиги |
| 0*001101011* | 0000000001 0011101110 | 0000000000 1001110111  0000000001 0011101110  0000000001 1101100101 | Сложение, сдвиги |
| 00*00110101* | 0000000010 0111011100 | 0000000001 1101100101  0000000010 0111011100  0000000100 0101000001 | Сложение, сдвиги |
| 000*0011010* | 0000000100 1110111000 | 0000000100 0101000001 | Сдвиги |
| 0000*001101* | 0000001001 1101110000 | 0000000100 0101000001  0000001001 1101110000  0000001110 0010110001 | Сложение, сдвиги |
| 00000*00110* | 0000010011 1011100000 | 0000001110 0010110001 | Сдвиги |
| 000000*0011* | 0000100111 0111000000 | 0000001110 0010110001  0000100111 0111000000  0000110101 1001110001 | Сложение, сдвиги |
| 0000000*001* | 0001001110 1110000000 | 0000110101 1001110001  0001001110 1110000000  0010000100 0111110001 | Сложение, сдвиги |
| 00000000*00* | 0010011101 1100000000 | 0010000100 0111110001 | Сдвиги |
| 000000000*0* | 0100111011 1000000000 | 0010000100 0111110001 | Сдвиги |
| 0000000000 | 1001110111 0000000000 | 0010000100 0111110001 |  |

Получено псевдо произведение 0010000011 1111110001, так как множитель отрицательный, то необходимо провести коррекцию дополнительным кодом множимого:

0010000100 0111110001

0110001001 0000000000

1000001101 0111110001

Необходима нормализация на 1 разряд, выполним сдвиг мантиссы в лево и вычтем из характеристики 1:

Результат произведения : 0000011010 1111100011 Характеристика: 0,1000

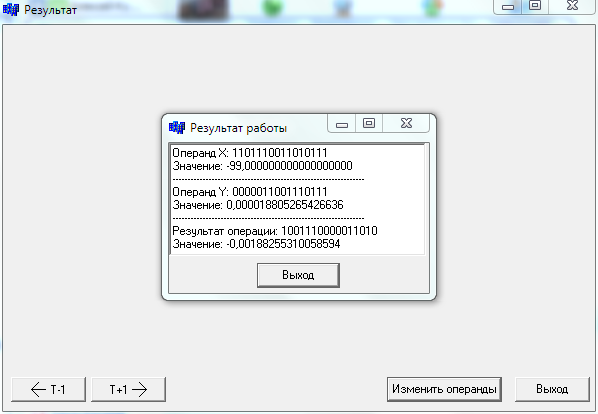
0,1111

1,0111

Ответ: 1,00111 00000110102 = -0,0018825531005859410

Проверка: -99\* 0,000018805265426636=-0,001861721277236964

Результат симуляции в САПР Quartus:



*4.Описание схемы:*

*1)Управляющие сигналы:*

у[0] – запись в RG1и RG2 , T1(inst1), обнуление CT2,запись 001001 в CT1;

у[1] – запись в RG2, Т2(inst7), RG4, обнуление RG3, установка T3(inst14) в положение «0»;

у[2]– сдвиг RG1 вправо, сдвиг RG2 влево, СТ1:=СТ1+1;

у[3]– подача сигнала на crp SM1(cin SM1), управление инвертором;

у[4]– запись в СТ1 значения с выхода SM2s(SM2 result);

у[5]– CT2:=CT2+1, сдвиг RG3 в лево;

у[6]– запись в RG3 значения с выхода SM1s(SM1 result);

у[7]– управление мультиплексором (MS1);

у[8]– установка Т1(inst1 ), Т2(inst7) в положение «0»;

у[10]– установка флага ПРС;

y[11] – сдвиг RG3 в лево на 1 разряд.

2)Осведомительные сигналы которые

P[1] – 1 если необходима коррекция множимым (знак множителя, Q триггера inst1);

Р[2] – 1 если необходима коррекция множителем (знак множимого,Q триггера inst14);

Р[3] – 1 если возникло ПМР;

Р[4]- младший анализируемый разряд множителя;

Р[5]– 0 если множимое или множитель равен 0 ;

Р[6]– 1 если операция умножения окончена(5 разряд СТ1);

Р[7]– 1 если возникло ПРС;

Р[8]– 1 если возникло временное ПРС;

Р[9] - 0 если необходима нормализация(результат сложения по модулю 2, знака и 20 разряда RG3);

P[10] – 1 если возникло временное прс в регистре ЧП(Q триггера inst31)

**Микропрограмма:**

module UA

(

// {{ALTERA\_ARGS\_BEGIN}} DO NOT REMOVE THIS LINE!

p, clk, clkout, y, Z

// {{ALTERA\_ARGS\_END}} DO NOT REMOVE THIS LINE!

);

// Port Declaration

// {{ALTERA\_IO\_BEGIN}} DO NOT REMOVE THIS LINE!

input [10:0] p;

input clk;

output clkout;

output [11:0] y;

output Z;

// {{ALTERA\_IO\_END}} DO NOT REMOVE THIS LINE!

integer pc=1;

reg [11:0] y;

wire clkout;

reg Z=0;

assign clkout=!clk;

always @(posedge clk)

begin

case(pc)

1:begin

y=12'b000000000011; *//блок 2*

pc=pc+1;

end

2:begin

if(p[5]==1) begin *//блок 3*

y=12'b000000010000; *//блок 4*

pc=pc+1;end

else begin

pc=16;

end;

end

3:begin

y=12'b000000000010; *//блок 6*

pc=pc+1;

end

4:begin

if(p[5]==1) begin *//блок 7*

y=12'b000000010000; *//блок 8*

pc=pc+1;end

else begin

pc=16;

end;

end

5:begin

y=12'b000000000000;

if (p[7]==0)begin *//блок 9*

pc=pc+1;

end else begin

pc=17;

end

end

6:begin

if(p[3]==0)begin *//блок 10*

pc=pc+1;

end

else

begin

pc=16;

end

end

7:begin

if(p[4]==1)begin *//блок 11*

y=12'b000001000000; *//блок 12*

pc=pc+1;

end

else begin

pc=pc+1;

end;

end

8:begin

y=12'b000000000100; *//блок 13*

pc=pc+1;

end

9:begin

y=12'b000000000000;

if(p[6]==0)begin *//блок 14*

pc=7;

end

else begin

pc=pc+1;

end

end

10:begin

y=12'b000000000000;

if(p[1]==1)begin *//блок 15*

y=12'b000001001000; *//блок 16*

end;

pc=pc+1;

end

11:begin

y=12'b000000000000;

if(p[2]==1)begin *//блок 17*

y=12'b000011001000; *//блок 18*

end

pc=pc+1;

end

12:begin

y=12'b000000000000;

if(p[9]==0)begin *//блок 19*

y=12'b000000100000; *//блок 20*

end

pc=pc+1;

end

13:begin

y=12'b000000000000;

if(p[8]==1)begin *//блок 22*

pc=17;end

else

begin

pc=pc+1;

end

end

14:begin

y=12'b000000000000;

if(p[3]==1)begin *//блок 21*

pc=16;

end

else

begin

pc=pc+1;

end

end

15:begin

y=12'b000000000000;

Z=1; *//блок 24*

end

16:begin

y=12'b000100000001; *//блок 23*

pc=15;

end

17:begin

y=12'b010000000000; *//блок 25*

pc=15;

end

endcase;

end

endmodule

**Вывод:** результат тестирования схемы ОА в САПР Quartus показал, что разработанный ОА работает правильно.



