Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №4

З предмету: «Комп’ютерна логіка»

Варіант 10

Виконав: студент групи ІО-91

Оксеньчук В. С.

Перевірив: Флеров А.І.

Київ 2010р.

**Завдання:** Виконати операцію множення над операндами з плаваючою комою у в доповняльному коді без відновлення остачі з округлюванням результату. Скласти функціональний мікроалгоритм та таблицю логічного модулювання.

**Виконання:**

А=41210=1010011112

В=63610=10101101002

Приведемо ці числа до потрібної форми:

А=29\*0,101001111;

В=210\*0,1010110100.

Запишемо порядки та мантиси цих чисел:

РА=00,1001 МА=0,101001111.

РВ=00,1010 МВ=0,10101101.

Виконуємо додавання порядків:

РС=РА+РВ=01,0011.

Проводимо нормалізацію: РС=00,10011.

Виконуємо множення мантис молодшими розрядами вперед зі зсувом суматора праворуч:

101001111

10101101

101001111

000000000

101001111

101001111

000000000

101001111

000000000

101001111

1110001001100011

МС=0,01110001001100011.

Мантиса не є нормалізованою, тому зсуваємо її на 1 розряд вправо при цьому коректуємо порядок віднімаючи одиницю.

МС=0,1110001001100011; РС=0,10010.

Після округлення мантиси отримуємо остаточний результат:

С=111000100110001100.

**Функціональний мікроалгоритм:**

Ні

так

Початок

RPA=00,1001

RMA=0,101001111

RPB=00,1010

RMB=0,1010110100

RPC=RPA+RPB;

RMC=RMA\*RMB

RPC(1)=0

RPC=RPC-1;

RMC=L(RMC).0;

Вихід

**Таблиця логічного моделювання:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № | RMA | RPA | RMB | RPB | RMC | RPC | МО |
| ПС | 0.101001111 | 00.1001 | 0.1010110100 | 00.1010 | 0.000000000 | 00.0000 | ПС |
| 1 |  |  |  |  | 0.01110001001100011 | 01.0011  00.10011 | RPC:=RPA+RPB  RMC:=RMA\*RMB |
| 2 |  |  |  |  | 0.1110001001100011 | 00.10010 | RPC:=RPC-1  RMC=L(RMC).0 |

**Лістинг програми:**

program dobutoc;

Type

Tindex=1..10;

Tzn=0..3;

Tarray=array [Tindex] of Tzn;

procedure suma(A,B:Tarray;var c:Tarray);

Var

i:integer;

Begin

For i:=1 to 10 do

begin

c[i]:=c[i]+a[i]+b[i];

if c[i]>1 then

begin

if c[i]mod 2=1 then

begin

c[i+1]:=c[i]div 2;

c[i]:=1;

end

else

begin

c[i+1]:=c[i] div 2;

c[i]:=0;

end;

end;

end;

end;

procedure turn\_right(var r:Tarray);

Var

i:integer;

Begin

For i:=1 to 1 do

r[i]:=r[i+1];

r[10]:=0;

end;

procedure dob(a,b:Tarray;var c:Tarray);

Var

i:integer;

Begin

For i:=1 to 10 do

begin

if b[i]=1 then

begin

suma(a,c,c);

end;

turn\_right(c);

end;

end;

procedure turn\_left(var r:Tarray);

Var

i:integer;

Begin

For i:=10 downto 2 do

r[i]:=r[i-1];

r[1]:=0;

end;

Var

RPA,RPB,RPC,RMA,RMB,RMC:Tarray;

i:integer;

Begin

Writeln('Write poryadok A');

For i:=1 to 10 do

Readln(RPA[i]);

Writeln('Write mantusa A');

For i:=1 to 10 do

Readln(RMA[i]);

Writeln('Write poryadok B');

For i:=1 to 10 do

Readln(RPB[i]);

Writeln('Write mantusa B');

For i:=1 to 10 do

Readln(RMB[i]);

suma(RPA,RPB,RPC);

dob(RMA,RMB,RMC);

If RMC[1]=0 then

begin

turn\_left(RMC);

RPC[10]:=RPC[10]-1;

end;

For i:=1 to 10 do

begin

write(RMC[i]);

end;

writeln;

For i:=1 to 10 do

write(RPC[i]);

end.