

Министерство образования Республики Беларусь
Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ
КАФЕДРА ИНФОРМАТИКИ

Отчёт по лабораторной работе №1

По дисциплине «Архитектура вычислительных систем»
По теме «Программирование арифметического сопроцессора»

Выполнил:

студент гр. 253503
Телего Е.А.

Проверил:

ассистент каф.
информ.
Калиновская А.А.

Минск 2024

Цель работы: Ознакомиться с предназначением арифметического сопроцессора. Рассмотреть строение и основные принципы работы сопроцессора. Изучить команды, доступные при использовании сопроцессора. Научиться работать с сопроцессором в ходе выполнения лабораторного задания.

Задание: Значение аргумента x изменяется от a до b с шагом h . Для каждого x найти значения функции $Y(x)$, суммы $S(x)$ и число итераций n , при котором достигается требуемая точность $\varepsilon = |Y(x) - S(x)|$. Результат вывести в виде таблицы. Значения a , b , h и ε вводятся с клавиатуры.

Вариант 5

$$S(x) = \sum_{k=1}^n x^k \sin\left(\frac{\pi k}{4}\right), \quad Y(x) = \frac{x \sin(\pi/4)}{1 - 2x \cos \frac{\pi}{4} + x^2}.$$

Листинг 1 – исходный код программы

```
#include <iostream>
#include <cmath>
#include <chrono>
#include <iomanip>
using namespace std;
void task(float a, float b, float h, float eps);
float PI = 3.14f;
int main()
{
    auto
    start_program = std::chrono::high_resolution_clock::now(
    );
    float a, b, h, eps;
    cout << endl << "Enter a, b, h, eps:" << endl;
    cin >> a >> b >> h >> eps;
    //a=-1.0f, b=1.5f, h=0.2f, eps=0.1f;
    task(a, b, h, eps);

    return 0;
}

void task(float a, float b, float h, float eps)
{
    cout << setprecision(3) << scientific << showpos <<
    "eps = " << eps << endl;
```

```

        cout << " | x          | S(x)          | Y(x)          |
abs(S - Y) | n | " << noshowpos << right << setw(2)
<< endl;;
for (float x = a; x <= b; x += h)
{

float k = 1.0f;
float zero = 0.0f, four = 4.0f, one = 1.0f, two =
2.0f;
float Y = 0.0f, S = 0.0f, f = 1.0f, tempY = 0.0f;
float xk = 0.0f;
//X*sin(PI/4)
asm(
        "fld %3;"
        "fld %2;"
        "fdiv;"
        "fsin;"
        "fld %1;"
        "fmul;"
        "fstp %0;"
        : "=m" (Y)
        : "m" (x), "m" (PI), "m" (four)
        );
        //cout<<Y<<endl;

//1-2x*cos(PI/4)+x^2
asm(
        "fld %3;"
        "fld %2;"
        "fdiv;"
        "fcos;"
        "fld %1;"
        "fmul;"
        "fld %5;"
        "fmul;"
        "fld %4;"
        "fsub;"
        "fld %1;"
        "fld %1;"
        "fmul;"
        "fadd;"
        "fstp %0;"
        : "=m" (tempY)
        : "m" (x), "m" (PI), "m" (four), "m" (one),
        "m" (two)

```

```

        );

        //cout<<tempY<<endl;

//divide
asm(
    "fld %1;"
    "fld %0;"
    "fdiv;"
    "fstp %0;"
    : "+m" (Y)
    : "m" (tempY)
    );

    //cout << "Y="<< Y <<endl;

    S = 0.0f;
    float tempS = 0.0f;
    while(abs(Y-S)>eps){ //abs eps temp
                                                                    auto
start_cycle=std::chrono::high_resolution_clock::now();
        //x^k
        xk = 1.0f;
        for(int j = 0;j < k; j++)
        {
            asm(
                "fld %0;"
                "fld %1;"
                "fmul;"
                "fstp %0;"
                : "+m" (xk)
                : "m" (x)
            );
        }
        //cout << "x^k=" << xk << endl;

        //sin(PI*k/4)
asm(
    "fld %2;"
    "fld %1;"
    "fdiv;"
    "fld %3;"
    "fmul;"
    "fsin;"
    "fstp %0;"

```

```

        : "=m" (tempS)
        : "m" (PI), "m" (four), "m" (k)
        );
        //cout<<"sin="<<S<<endl;

        //mul
asm(
        "fld %1;"
        "fld %2;"
        "fmul;"
        "fld %0;"
        "fadd;"
        "fstp %0;"
        : "+m" (S)
        : "m" (xk), "m" (tempS)
        );
        //cout<<"S="<<S<<endl;
        k+=1.0f;

auto
end_cycle=std::chrono::high_resolution_clock::now();
        std::chrono::duration<double>
elapsed_iterations = end_cycle - start_cycle;
        //cout<<"Time for iteration
"<<elapsed_iterations.count()<<" seconds"<<endl;
        if(k>20.0f){
                break;
        }
    }
    int n = k;
    cout << setprecision(3) << scientific << showpos
<< " | "
        << x << " | " << S << " | " << Y << " | " <<
abs(S - Y)
        << " | " << noshowpos << right << setw(2) << n <<
" | " << endl;

}
return;
}

```

```

Enter a, b, h, eps:
0
1.5
0.3
0.3
eps = +3.000e-01

```

x	S(x)	Y(x)	abs(S - Y)	n
+0.000e+00	+0.000e+00	+0.000e+00	+0.000e+00	1
+3.000e-01	+2.120e-01	+3.186e-01	+1.065e-01	2
+6.000e-01	+7.841e-01	+8.297e-01	+4.562e-02	3
+9.000e-01	+1.446e+00	+1.185e+00	+2.609e-01	3
+1.200e+00	+8.482e-01	+1.143e+00	+2.945e-01	2
+1.500e+00	+1.060e+00	+9.401e-01	+1.202e-01	2

Рисунок 1 – работа программы на промежутке $[0;1.5]$ с шагом 0.3 и точностью 0.3

```

Enter a, b, h, eps:
-1.0
1.5
0.2
0.1
eps = +1.000e-01

```

x	S(x)	Y(x)	abs(S - Y)	n
-1.000e+00	-4.100e-01	-2.070e-01	+2.030e-01	21
-8.000e-01	-2.879e-01	-2.040e-01	+8.393e-02	4
-6.000e-01	-2.170e-01	-1.920e-01	+2.502e-02	4
-4.000e-01	-1.227e-01	-1.638e-01	+4.108e-02	3
-2.000e-01	-1.414e-01	-1.069e-01	+3.451e-02	2
-2.980e-08	+0.000e+00	-2.107e-08	+2.107e-08	1
+2.000e-01	+1.414e-01	+1.867e-01	+4.537e-02	2
+4.000e-01	+4.427e-01	+4.759e-01	+3.317e-02	3
+6.000e-01	+7.841e-01	+8.297e-01	+4.562e-02	3
+8.000e-01	+1.205e+00	+1.113e+00	+9.274e-02	3
+1.000e+00	+2.420e+00	+1.208e+00	+1.212e+00	21
+1.200e+00	+4.519e+01	+1.143e+00	+4.405e+01	21
+1.400e+00	+9.896e-01	+1.010e+00	+2.090e-02	2

Рисунок 2 – работа программы на промежутке $[-1;1.5]$ с шагом 0.2 и точностью 0.1

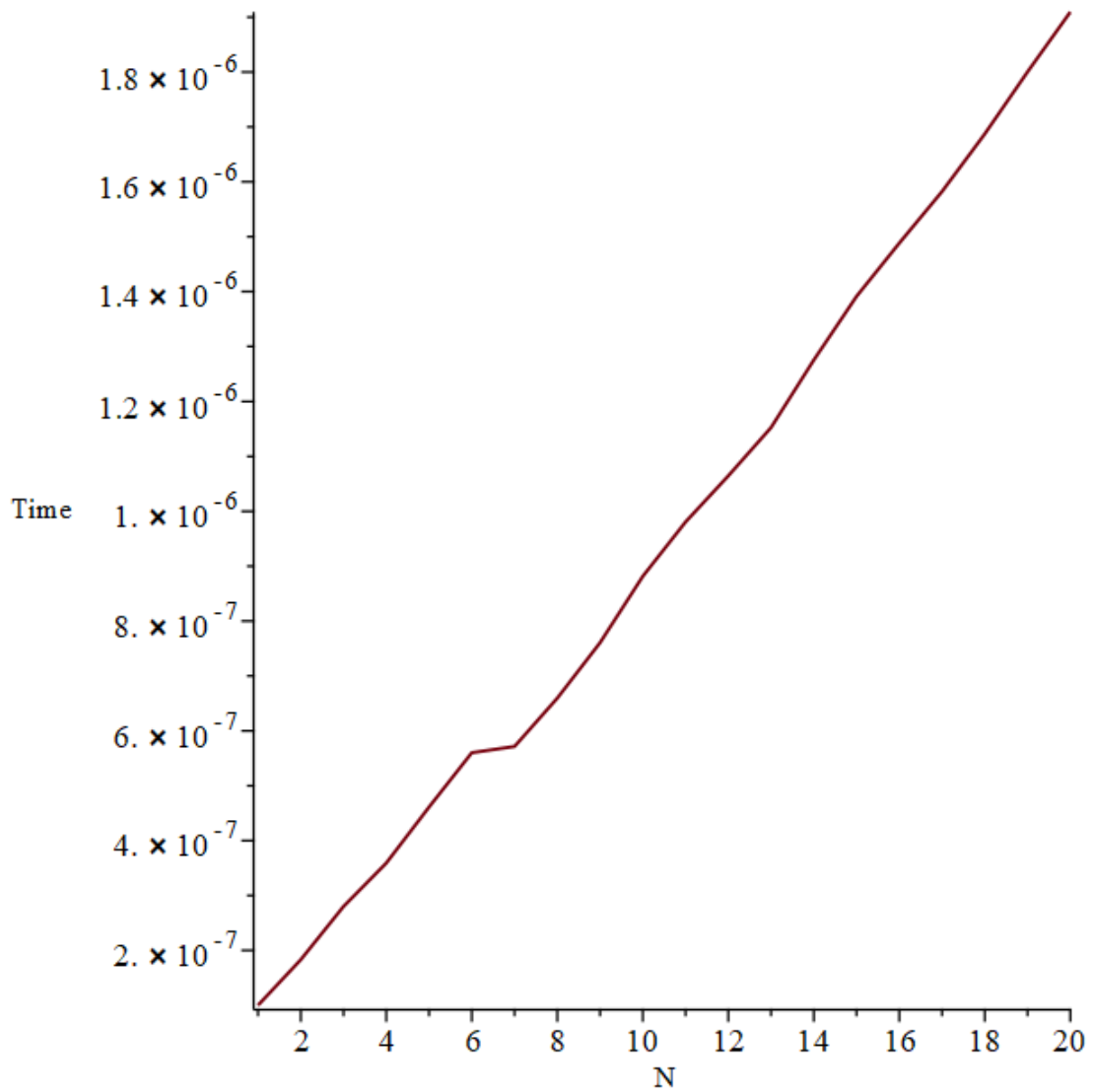


Рисунок 3 – график количества времени в секундах от итераций программы при x равном -1 и точностью 0.1

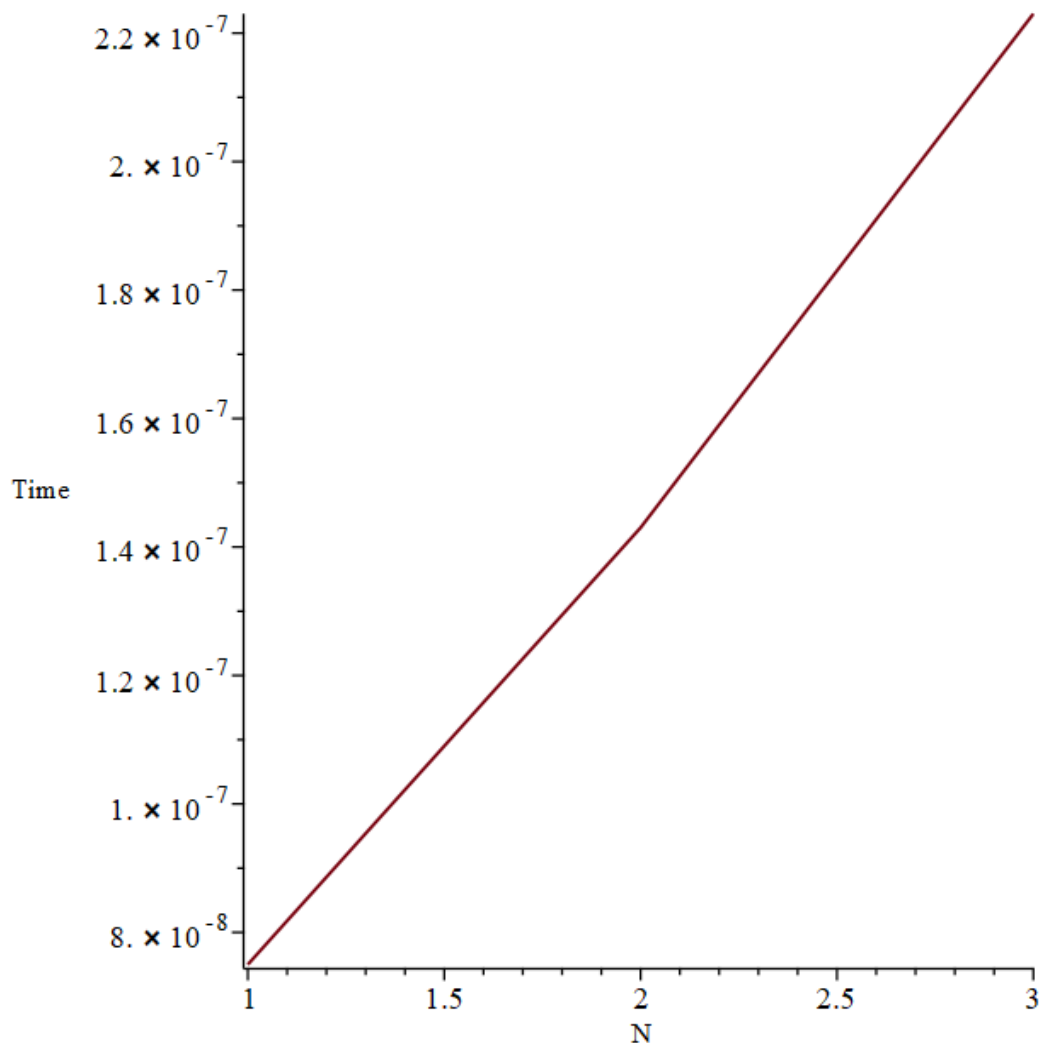


Рисунок 4 – график количества времени в секундах от итераций программы при x равном -0.8 и точностью 0.1

Выводы: В ходе выполнения лабораторной работы были изучены понятия арифметического сопроцессора, сопроцессорных конфигураций, программной модели сопроцессора. Рассмотрены различные форматы представления численных данных: двоичные целые числа, упакованные десятичные числа, вещественные числа. Выяснены режимы работы и состояния арифметического сопроцессора, система команд сопроцессора, особенности задания команд, различные группы команд сопроцессора (команды управления, команды передачи данных, команды загрузки, команды запоминания, арифметические команды).