ФГБОУ ВО "Чувашский государственный университет им. И. Н. Ульянова" Факультет: ИВТ

Кафедра: Вычислительной техники

Предмет: Электронно-вычислительная машина и периферийные устройства

Лабораторная работа №3 **Архитектура и программирование арифметического сопроцессора**

Выполнил: студент группы ИВТ-41-20

Галкин Дмитрий

Проверил: доцент Андреева А.А.

Цель работы

Изучить архитектуру сопроцессора и запрограммировать арифметических уравнений с помощью арифметического сопроцессора.

Текст задания

Вычислить функцию $y = \ln(1+x)$ при |x| < 1 двумя способами: а) с использованием трансцендентных команд сопроцессора; б) путем разложения функции в ряд $y = x - \frac{x^2}{2} + \frac{x^3}{4} - \dots$

Алгоритм на языке высокого уровня

```
double x = 0.2;
double y = Math.log(x + 1);
System.out.println("Result_1: " + y);
```

```
double sum = x;
double eps = 1.0E-05;
double chisl = x;
double znam = 1;

do {
    znam++;
    chisl *= (-x);
    sum += (chisl / znam);
} while (Math.abs(chisl) / znam > eps);

System.out.println("Result_2: " + sum);
```

```
Result_1: 0.1823215567939546
Result_2: 0.1823218285714286
```

Текст программы

Code1

;ВЫЧИСЛЕНИЕ ЭКСПОНЕНТЫ С ПОМОЩЬЮ ФУНКЦИЙ СОПРОЦЕССОРА

- ; Вычислить функцию y = ln(1+x) при |x| < 1 двумя способами:
- ; а) с использованием трансцендентных команд соцпроцессора
- ; б) путем разложения функции в ряд $y = x-(x^3/3) + (x^5/5)-...$

```
assume cs:cseg,ds:cseg x dd 0.2; аргумент функции y dd?; результат вычисления функции beg: mov ax,cs; настроить сегментные mov ds,ax; регистры finit; инициализировать; сопроцессор fldln2; загрузить ln(2) fld x; загрузить x fyl2xp1; вычислить ln(x+1)=ln(2)*log2(x+1) fstp y; сохранить результат mov ax,4c00h; вернуться в DOS int 21h Cseg ends end beg
```

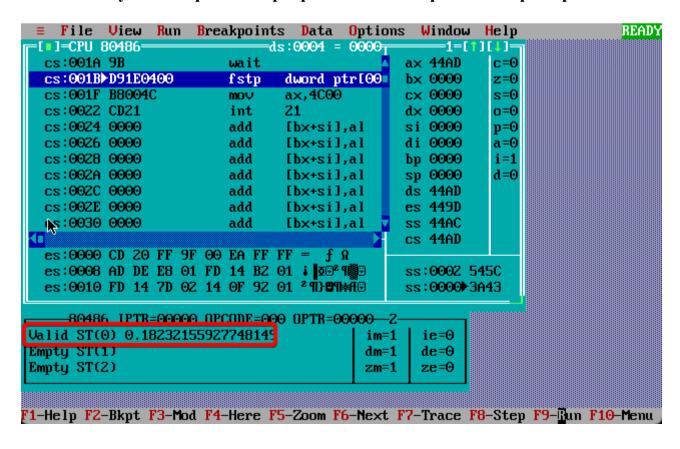
Code2

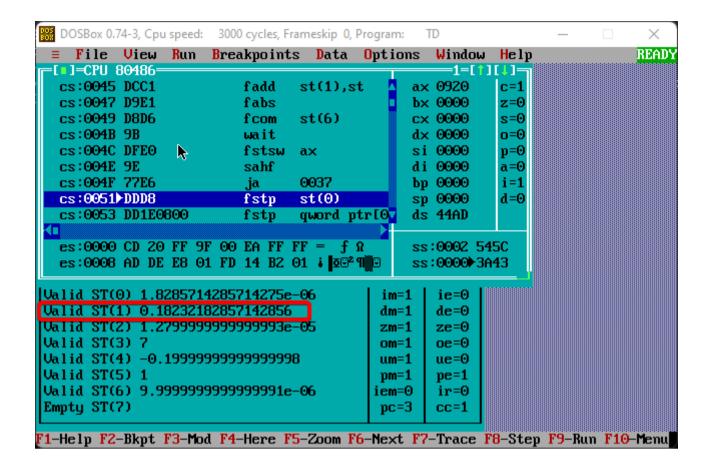
```
;ВЫЧИСЛЕНИЕ ЭКСПОНЕНТЫ С ПОМОЩЬЮ РАЗЛОЖЕНИЯ В РЯД ; Вычислить функцию y = \ln(1+x) при |x| < 1 двумя способами: ; б) путем разложения функции в ряд y = x - (x^3/3) + (x^5/5) - \dots ; Алгоритм вычисления ; 1. znam=1; chisl=x; S=x ; 2. znam=znam+1; chisl=chisl *(-x); chisl/znam; S=S+chisl/znam ; 3. Если ABS(chisl/znam)>eps идти к 2, иначе - закончить распределение регистров ; ST(0) - рабочий, ST(1) - S, ST(2) - chisl, ; ST(3) - znam, ST(4) - -x, ST(5) - 1, ST(6) - eps
```

Cseg segment assume cs:Cseg,ds:Cseg х dq 0.2; аргумент функции sum dq?; результат вычисления функции eps dq 1.0E-05; точность вычисления start: mov ax,cs; настроить сегментные mov ds, ax; регистры finit; инициализировать сопроцессор .386 ; реализация первого шага алгоритмов fld eps; загрузка точности eps fld1; загрузка 1 fld x; загрузить x fchs fld1; загрузка 1 fld x; загрузка 1 fld x; загрузка x fld1; загрузка 1

```
; шаг 2
calc:
      fxch st(3); обмен значений st(0) <-> st(3) - znam
      fadd st(0), st(5)
      fst st(3)
      fxch st(2); обмен значений st(0) <-> st(2) - chisl
      fmul st(0), st(4); chisl * x
      fst st(2)
      fdiv st(0), st(3); chisl/znam
      fadd st(1),st(0)
      fabs; ABS(chisl)
      fcomi st,st(6); ABS(chis1)/znam > eps
      :db 0dbh.0f6h : машинный код команды
      fcom st(6)
      fstsw ax
      sahf
      ja calc
      fstp st;
      fstp sum; запоминание результата
      finit
mov ax,4c00h; вернуться в DOS
int 21h
Cseg ends
end start
```

Результаты работы программы на контрольных примерах





Вывод

Изучил архитектуру сопроцессора и запрограммировал арифметическое уравнение с помощью арифметического сопроцессора по заданному варианту задания.