Міністерство освіти і науки України

Національний університет “Львівська політехніка”



**Звіт**

**До** **Лабораторної роботи №4**

З дисципліни: «Комп'ютерні системи»

на тему: «**Аналіз програмної моделі процесу роботи арифметичного конвеєра, ч.2»**

**Підготував:** ст.гр. КІ-33

Барсієнко В.О.

**Прийняв:**

Асистент Козак Н.Б.

Львів 2020

Мета роботи: Навчитись здійснювати аналіз програмних моделей комп’ютерних систем, виконаних на мові System C.

**Хід роботи**

**Завдання**

Здійснити модернізацію функцій або параметрів арифметичного конвеєра (див. лабораторну роботу № 3), шляхом під’єднання розроблених модулів S1 та S2 (див. лабораторну роботу № 2). Порядок та тип з’єднання мають бути обгрунтовані, можливо розробка буферних або додаткових модулів з метою надавання нових властивостей тестувальній моделі.

**Лістинг змінених файлів тестової програми**

S1.h

#ifndef S1\_H

#define S1\_H

// X = 1 - номер варіанту

// Y = B + V = 66 + 86 = 152 - сума ASCII першої літери Пр і Імя

// out o1 = X - Y, o2 = (X + Y/2.0) && (x >> 2)

const int X = 1;

const int Y = 152;

SC\_MODULE(S1) {

    sc\_out<long> o1, o2;  //output 1

    sc\_in<bool>    clk;  //clock

    void mainFunc();       //method implementing functionality

    //Counstructor

    SC\_CTOR(S1) {

        SC\_METHOD(mainFunc);   //Declare addsub as SC\_METHOD and

        sensitive\_pos << clk;  //make it sensitive to positive clock edge

    }

};

#endif

S1.cpp

#include "systemc.h"

#include "S1.h"

//definition of multdiv method

void S1::mainFunc(){

o1.write(X - Y);

o2.write((X + Y / 2.0) && (X >> 2));

} // end of multdiv

S2.cpp

#include "systemc.h"

#include "S2.h"

//definition of multdiv method

void S2::mainFunc(){

long a, b;

a = in1.read();

b = in2.read();

o1.write(a \* b);

o2.write(b - a);

} // end of multdiv

S2.h

#ifndef S2\_H

#define S2\_H

SC\_MODULE(S2) {

    sc\_in<long> in1, in2;

    sc\_out<double> o1, o2;

    sc\_in<bool>    clk;  //clock

    void mainFunc();       //method implementing functionality

    //Counstructor

    SC\_CTOR(S2) {

        SC\_METHOD(mainFunc);   //Declare addsub as SC\_METHOD and

        sensitive\_pos << clk;  //make it sensitive to positive clock edge

    }

};

#endif

TestModule.h

#ifndef TestModule\_H

#define TestModule\_H

#include "S1.h"

#include "S2.h"

SC\_MODULE(TestModule) {

    sc\_out<double> o1, o2;

    sc\_in<bool> clk;  //clock

S1 s1;

S2 s2;

sc\_signal<long> toA, toB;

    void mainFunc();       //method implementing functionality

    //Counstructor

    SC\_CTOR( TestModule ) : s1("S1"), s2("S2") {

s1.o1(toA);

s1.o2(toB);

s2.in1(toA);

s2.in2(toB);

s1.clk(clk);

s2.clk(clk);

s2.o1(o1);

s2.o2(o2);

sensitive\_pos << clk;  //make it sensitive to positive clock edge

    }

};

#endif

TestModule.cpp

#include "systemc.h"

#include "TestModule.h"

#include "S1.h"

#include "S2.h"

//definition of multdiv method

void TestModule::mainFunc(){

s1.o1(toA);

s1.o2(toB);

s2.in1(toA);

s2.in2(toB);

s1.clk(clk);

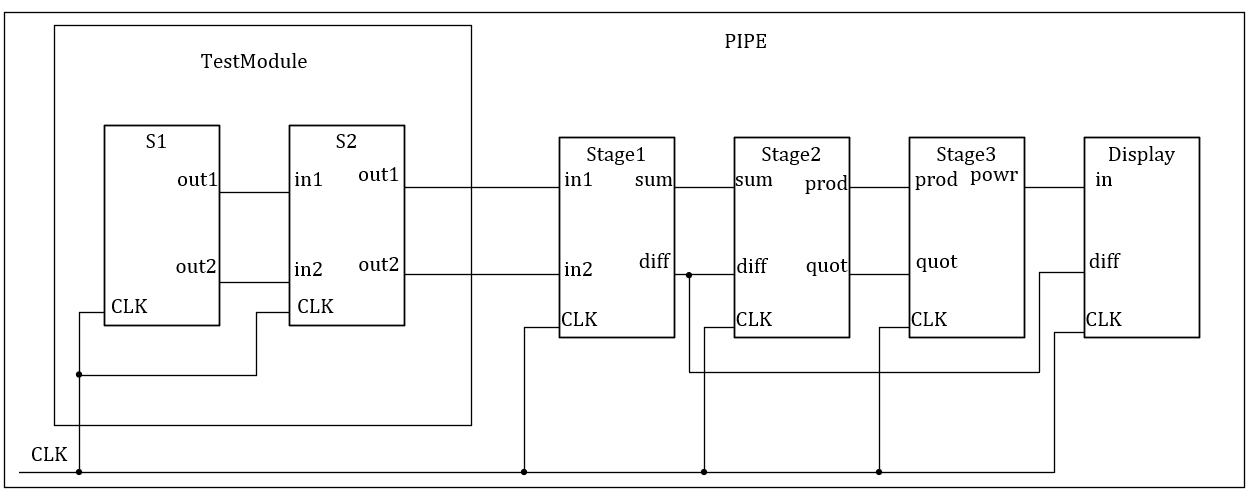
s2.clk(clk);

s2.o1(o1);

s2.o2(o2);

} // end of multdiv

**Модель арифметичного конвеєра з внесеними змінами**



**Висновок:** на даній лабораторній роботі навчитився здійснювати аналіз програмних моделей комп’ютерних систем, виконаних на мові System C.