Міністерство освіти і науки України

Національний університет “Львівська політехніка”



**Звіт**

**До** **лабораторної роботи №2**

З дисципліни: «Комп'ютерні системи»

на тему: «**Ознайомлення з основними конструкціями мови моделювання System C»**

Виконав: ст. гр. КІ-33

Харченко М.В.

Прийняв: асистент каф. ЕОМ

Козак Н.Б.

Львів-2020

**Мета роботи:** Ознайомитися з основними конструкціями мови моделювання System C.

**Хід роботи**

**Варіант – 20**

Х = 20

Y = 160

Реалізувати модулі S1 та S2, разом з логікою їх функціонування згідно варіанту, провести послідовне з’єднання S1 та S2 ініціалізувати необхідні порти на S1. На вхід S2 подати вихідні порти модуля S1. До кожного модуля заводиться зовнішній вхідний для всіх сигнал синхронізації CLK. Результати подати на модуль Display для відображення.

1. **S1**

f1 = X < Y ? X - Y : Y - X;

rep = Y > 0 ? Y! : X!;

1. **S2**

Обчислити r1 = b / a , r2 = a / b;

**Схема зв’язків модулів**

r1 = b/a

r2 = a/b

f1 = X < Y ? X - Y : Y - X;

rep = Y > 0 ? Y! : X!;

f1

rep

r1

r2

Y

X

b

a

S2

S1

**Код програми**

#pragma once

#ifndef TEST\_SYSTEMC\_S1\_H

#define TEST\_SYSTEMC\_S1\_H

#include "systemc.h"

#include "cmath"

#include "s2.h"

SC\_MODULE(S1) {

sc\_in<int> x;

sc\_in<double> y;

sc\_out<double> o1;

sc\_out<double> o2;

sc\_in<bool> clk;

SC\_CTOR(S1) {

SC\_METHOD(calc\_o2);

sensitive << x << y;

sensitive\_pos << clk;

SC\_METHOD(calc\_o1);

sensitive << x << y;

sensitive\_pos << clk;

}

void calc\_o2() {

int x\_val = x.read();

double y\_val = y.read();

double acc;

acc = ((x\_val + y\_val) / 2) && (x\_val >> 2);

o2.write(acc);

}

void calc\_o1() {

int x\_val = x.read();

double y\_val = y.read();

o1.write(x\_val - y\_val);

}

};

#endif //TEST\_SYSTEMC\_S1\_H// s2.h

#pragma once

#ifndef TEST\_SYSTEMC\_S2\_H

#define TEST\_SYSTEMC\_S2\_H

#include "systemc.h"

SC\_MODULE(S2) {

sc\_in<double> a;

sc\_in<double> b;

sc\_out<double> r1;

sc\_out<double> r2;

sc\_in<bool> clk;

SC\_CTOR(S2) {

SC\_METHOD(calc\_r1);

sensitive << a << b;

sensitive\_pos << clk;

SC\_METHOD(calc\_r2)

sensitive << a << b;

sensitive\_pos << clk;

}

void calc\_r1() {

double a\_val = a.read();

double b\_val = b.read();

r1.write(b\_val / a\_val);

}

void calc\_r2() {

double a\_val = a.read();

double b\_val = b.read();

r2.write(a\_val / b\_val);

}

};

#endif //TEST\_SYSTEMC\_S2\_H

Main.cpp

#include <iostream>

#include <systemc.h>

#include "s1.h"

#include "s2.h"

using namespace std;

int sc\_main(int argc, char\* argv[]) {

sc\_signal<int> x{ "x", 20 };

sc\_signal<double> y{ "y", 152 };

sc\_signal<double> o1{ "o1" };

sc\_signal<double> o2{ "o2" };

sc\_signal<double> r1{ "r2" };

sc\_signal<double> r2{ "r1" };

sc\_clock clk("clk", 1, SC\_NS, 0.5);

S1 s1("s1");

S2 s2("s2");

s1.clk(clk);

s2.clk(clk);

s1.x(x);

s1.y(y);

s1.o1(o1);

s1.o2(o2);

s2.a(o1);

s2.b(o2);

s2.r1(r1);

s2.r2(r2);

sc\_start(10, SC\_NS);

cout << "x: " << x.read() << endl;

cout << "y: " << y.read() << endl;

cout << "r1: " << r1.read() << endl;

cout << "r2: " << r2.read() << endl;

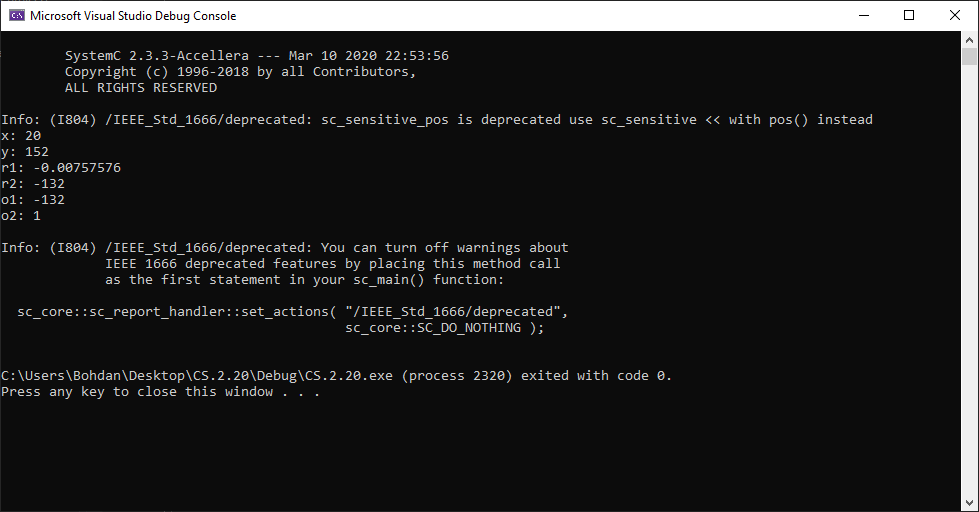
cout << "o1: " << o1.read() << endl;

cout << "o2: " << o2.read() << endl;

return 0;

}

**Результат виконання**



**Висновок:** на даній лабораторній роботі я ознайомився з основними конструкціями мови моделювання System C. Створив програму та відлагодив її за допомогою System C .