**Міністерство освіти і науки України**

**Національний університет «Львівська політехніка»**

**Кафедра ЕОМ**



Звіт

до лабораторної роботи № 2

з дисципліни «Комп’ютерні системи»

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

Варіант №17

Виконав:

ст.гр. КІ-38

Онисько О.А.

Прийняв:

викл.

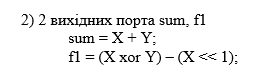
Козак Н.Б.

**Львів 2022**

Мета роботи: Ознайомлення з основними конструкціями мови моделювання System C.

**Завдання**

Х = 17. № варіанту за списком в журналі.





Y = 75 + 66 = 141. Сума ASCII code першої літери прізвища(K) + першої літери імені(B).

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.

2 вихідних порта: sum, f1

sum = X + Y;

f1 = (X xor Y) – (X << 1);

Обчислити r1 = XOR (a, b) , r2 = NOT( XOR(a, b)); операції NOT & XOR побітово.

**Код програми:**

main.cpp

#include "systemc.h"

#include "Stimulator.h"

#include "Monitor.h"

#include "S1\_unit.h"

#include "S2\_unit.h"

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

{

sc\_signal<int> XSig, YSig;

sc\_signal<int> SUM\_Sig, F1Sig;

sc\_signal<int> R1Sig, R2Sig;

sc\_clock TestClk("TestClock", 10, SC\_NS, 0.5);

Stimulator Stim1("stim");

Stim1.X(XSig);

Stim1.Y(YSig);

Stim1.CLK(TestClk);

s1\_unit first\_part("s1\_unit");

first\_part.x\_in(XSig);

first\_part.y\_in(YSig);

first\_part.sum\_out(SUM\_Sig);

first\_part.f1\_out(F1Sig);

s2\_unit second\_part("s2\_unit");

second\_part.sum\_in(SUM\_Sig);

second\_part.f1\_in(F1Sig);

second\_part.r1(R1Sig);

second\_part.r2(R2Sig);

Monitor Monitor1("monitor");

Monitor1.X(XSig);

Monitor1.Y(YSig);

Monitor1.SUM(SUM\_Sig);

Monitor1.F1(F1Sig);

Monitor1.R1(R1Sig);

Monitor1.R2(R2Sig);

Monitor1.CLK(TestClk);

sc\_start();

return 0;

}

Monitor.h

#include "systemc.h"

#include <iomanip>

SC\_MODULE(Monitor)

{

sc\_in<int> X, Y;

sc\_in<int> SUM, F1;

sc\_in<int> R1, R2;

sc\_in<bool> CLK;

void monitor()

{

cout << std::setw(15) << "Time";

cout << std::setw(7) << "X";

cout << std::setw(7) << "Y";

cout << std::setw(7) << "SUM";

cout << std::setw(7) << "F1";

cout << std::setw(7) << "R1";

cout << "\t" << std::setw(7) << "R2" << endl;

for (int i = 0; i < 3;i++)

{

cout << std::setw(15) << sc\_time\_stamp();

cout << std::setw(7) << X.read();

cout << std::setw(7) << Y.read();

cout << std::setw(7) << SUM.read();

cout << std::setw(7) << F1.read();

cout << std::setw(7) << R1.read();

cout << "\t" << std::setw(7) << R2.read() << endl;

wait();

}

sc\_stop();

}

SC\_CTOR(Monitor)

{

SC\_THREAD(monitor);

sensitive << CLK.pos();

}

};

S1\_unit.h

#pragma once

#include "systemc.h"

SC\_MODULE(s1\_unit)

{

sc\_in<int> x\_in, y\_in;

sc\_out<int> sum\_out, f1\_out;

void sum()

{

sum\_out.write(x\_in + y\_in);

}

void F1()

{

sum();

f1\_out.write((x\_in xor y\_in) - (x\_in << 1));

}

SC\_CTOR(s1\_unit)

{

SC\_METHOD(F1);

sensitive << x\_in << y\_in;

}

};

S2\_unit.h

#pragma once

#include "systemc.h"

#include "S1\_unit.h"

SC\_MODULE(s2\_unit)

{

sc\_in<int> sum\_in, f1\_in;

sc\_out<int> r1, r2;

void OutR1()

{

int result = sum\_in.read() xor f1\_in.read();

r1.write(result);

}

void OutR2()

{

OutR1();

r2.write(!(sum\_in.read() xor f1\_in.read()));

}

SC\_CTOR(s2\_unit)

{

SC\_METHOD(OutR2);

sensitive << sum\_in << f1\_in;

}

};

Simulator.h

#pragma once

#include "systemc.h"

SC\_MODULE(Stimulator)

{

sc\_out<int> X, Y;

sc\_in<bool> CLK;

void Generator()

{

X.write(17);

Y.write(75 + 66);

wait();

}

SC\_CTOR(Stimulator)

{

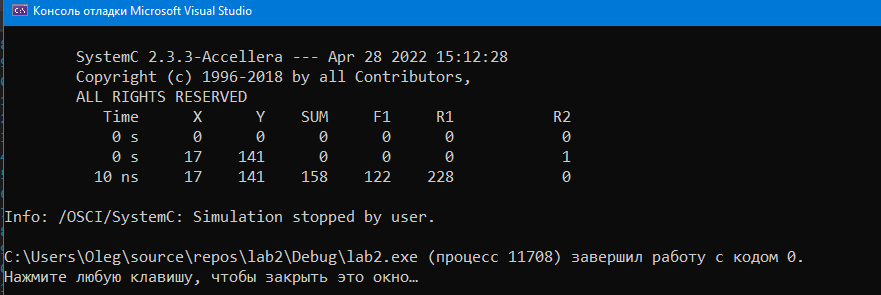
SC\_THREAD(Generator);

sensitive << CLK.pos();

}

};

**Результат виконання програми:**



*Рис.1. Результат виконання програми*

**Висновок:** Я ознайомився з термінологією, специфікацією та іншими основними поняттями мови моделювання SystemC, вивчив особливості основної мови моделювання процесів SystemC, які можуть бути реалізовані як апаратно (переважно), так і програмним шляхом. А також дав відповіді на контрольні запитання.