

**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 №27**

**Виконав:**  
**ст.гр. КІ-38**  
**Швець А.Ю.**  
**Прийняв:**  
**Козак Н. Б.**

**Львів 2022 р.**

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

### Завдання

$X = 27$ . № варіанту за списком в журналі.

$Y = 83 + 65 = 148$ . Сума ASCII code першої літери прізвища(S) + першої літери імені(A).

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: f1, f2  $f1 = Y + X!$ ;  $f2 = X / Y$ ; (+ check if  $Y \neq 0$ )

Обчислити  $r1 = \text{AND}(a, b)$ ,  $r2 = \text{NOT}(\text{AND}(a, b))$ ; операції NOT & AND побітово.

### Хід роботи

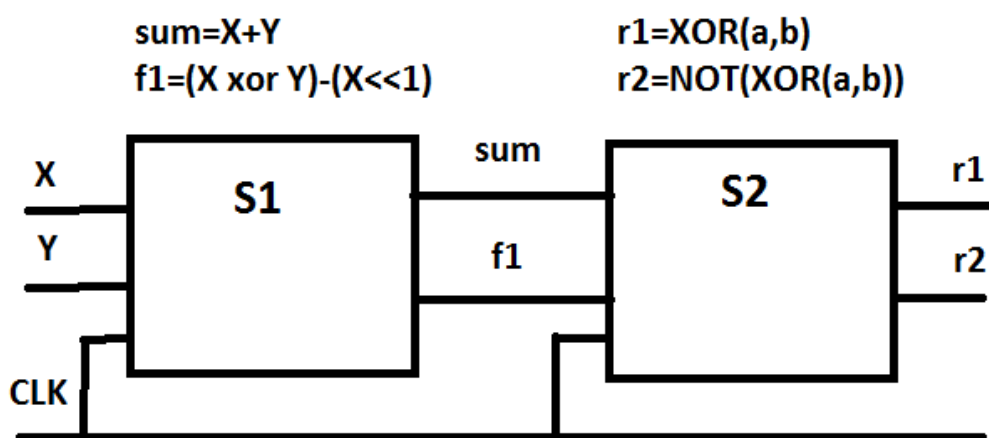


Рис.1. Створена схема відповідно до варіанту

```
SystemC 2.3.3-Accellera --- Feb 21 2022 19:14:31
Copyright (c) 1996-2018 by all Contributors,
ALL RIGHTS RESERVED

Time      X      Y      Sum      F1      R1      R2
0 s       0      0       0       0       0       0
0 s       27     148      0       0       0       1
10 ns     27     148     175     89     246      0

Info: /OSCI/SystemC: Simulation stopped by user.
```

Рис.2. Результат виконання програми

## Код програми

### S1.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 outSum()
    {
        sum_out.write(y_in.read() + x_in.read());
    }

    void outF1()
    {
        outSum();

        f1_out.write((x_in.read() ^ y_in.read())-(x_in.read() << 1));
    }
    SC_CTOR(s1_unit)
    {
        SC_METHOD(outF1);

        sensitive << x_in << y_in;
    }
};
```

### S2.h

```
#pragma once
#include "systemc.h"
#include "S1.h"

SC_MODULE(s2_unit)
{
    sc_in<int> sum_in, f1_in;
    sc_out<int> r1, r2;

    void OutR1()
    {
        int result = sum_in.read() ^ f1_in.read();
        r1.write(result);
    }

    void OutR2()
    {
        OutR1();
        r2.write(!(sum_in.read() ^ f1_in.read()));
    }

    SC_CTOR(s2_unit)
    {
        SC_METHOD(OutR2);

        sensitive << sum_in << f1_in;
    }
};
```

### Monitor.h

```
#pragma once
#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();
    }
};

```

## Stimulator.h

```

#pragma once
#include "systemc.h"

SC_MODULE(Stimulator)
{
    sc_out<int> X, Y;
    sc_in<bool> CLK;

    void Generator()
    {
        X.write(27); //option
        Y.write(static_cast<int>('A') + static_cast<int>('S')); //summary of
ASCII codes of 'A' and 'S'
        wait();
    }

    SC_CTOR(Stimulator)
    {
        SC_THREAD(Generator);
        sensitive << CLK.pos();
    }
};

```

## Source.cpp

```

#include "systemc.h"
#include "Stimulator.h"
#include "Monitor.h"
#include "S1.h"
#include "S2.h"

```

```

int sc_main(int argc, char* argv[])
{
    sc_signal<int> XSig, YSig;
    sc_signal<int> SumSig, 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(SumSig);
    first_part.f1_out(F1Sig);

    s2_unit second_part("s2_unit");
    second_part.sum_in(SumSig);
    second_part.f1_in(F1Sig);
    second_part.r1(R1Sig);
    second_part.r2(R2Sig);

    Monitor Monitor1("monitor");
    Monitor1.X(XSig);
    Monitor1.Y(YSig);
    Monitor1.Sum(SumSig);
    Monitor1.F1(F1Sig);
    Monitor1.R1(R1Sig);
    Monitor1.R2(R2Sig);
    Monitor1.CLK(TestClk);

    sc_start();

    return 0;
}

```

**Висновок:** у результаті виконання даної лабораторної роботи 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.