|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

Институт информационных технологий

Кафедра вычислительной техники

|  |
| --- |
| **ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №4**  **Разработка делителя на базе операционного автомата** |
| *Разработка программно-аппаратного обеспечения информационных и автоматизированных систем* |

|  |  |  |
| --- | --- | --- |
| Отчет представлен  к рассмотрению:  Студент группы ИКМО-05-18 | « » декабря 2022 г. | Пленкин А.В. |
|  |  | (подпись и расшифровка подписи) |
|  |  |  |
| Отчет допущен  к защите: | « » декабря 2022 г. |  |
|  |  | (подпись и расшифровка подписи) |
| Руководитель  от кафедры | « » декабря 2022 г. | Унгер А.Ю. |
|  |  | (подпись и расшифровка подписи) |
|  |  |  |

Москва 2022

**Оглавление**

[Введение 3](#_Toc122295308)

[Вывод 9](#_Toc122295309)

[Использованная литература 9](#_Toc122295310)

[Листинг файлов с исходным кодом 10](#_Toc122295311)

# Введение

В основе множества алгоритмов обработки данных лежат простые операции над числами. К одной из таких операций относится деление целых чисел. Эта операция как привило не реализуется на аппаратном уровне в ПЛИС начального и среднего уровня, поэтому важно уметь реализовывать ее программно. Эффективные алгоритмы деления целых чисел известны с незапамятных времен. Одним из наиболее известных алгоритмов является "деление уголком".

**Цель работы:** разработать синхронный программно-аппаратный делитель двух целых чисел на базе программируемой логической интегральной схемы.

Результатом работы такого делителя может являться:

1. Целые частное и остаток, если производится деление нацело.
2. Дробное частное в формате **"**целая часть **.** дробная часть**"**.

В лабораторной работе требуется получить результат №2, т.е. число, разделенное десятичной точкой.

Порядок выполнения.

1. Операционный автомат, реализующий операцию деления двух целых чисел с получением нецелого частного (Рисунок 1).

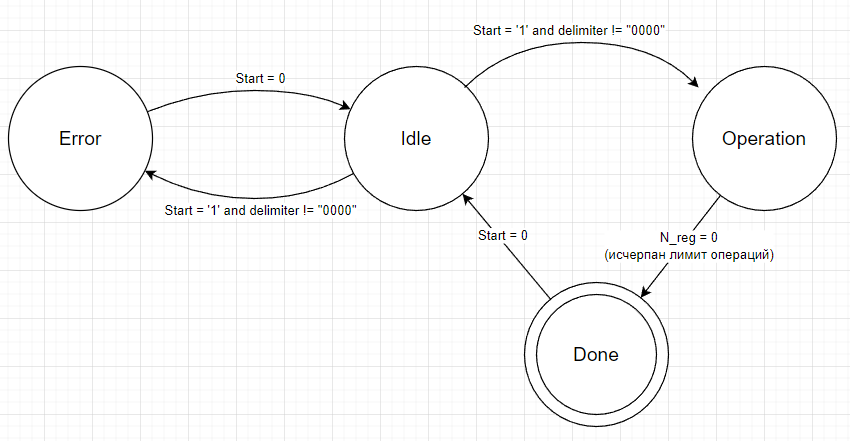


Рисунок 1 – диаграмма состояний делителя

1. Блок-схема алгоритма работы устройства на базе метода регистровых передач представлена на рисунке 2.

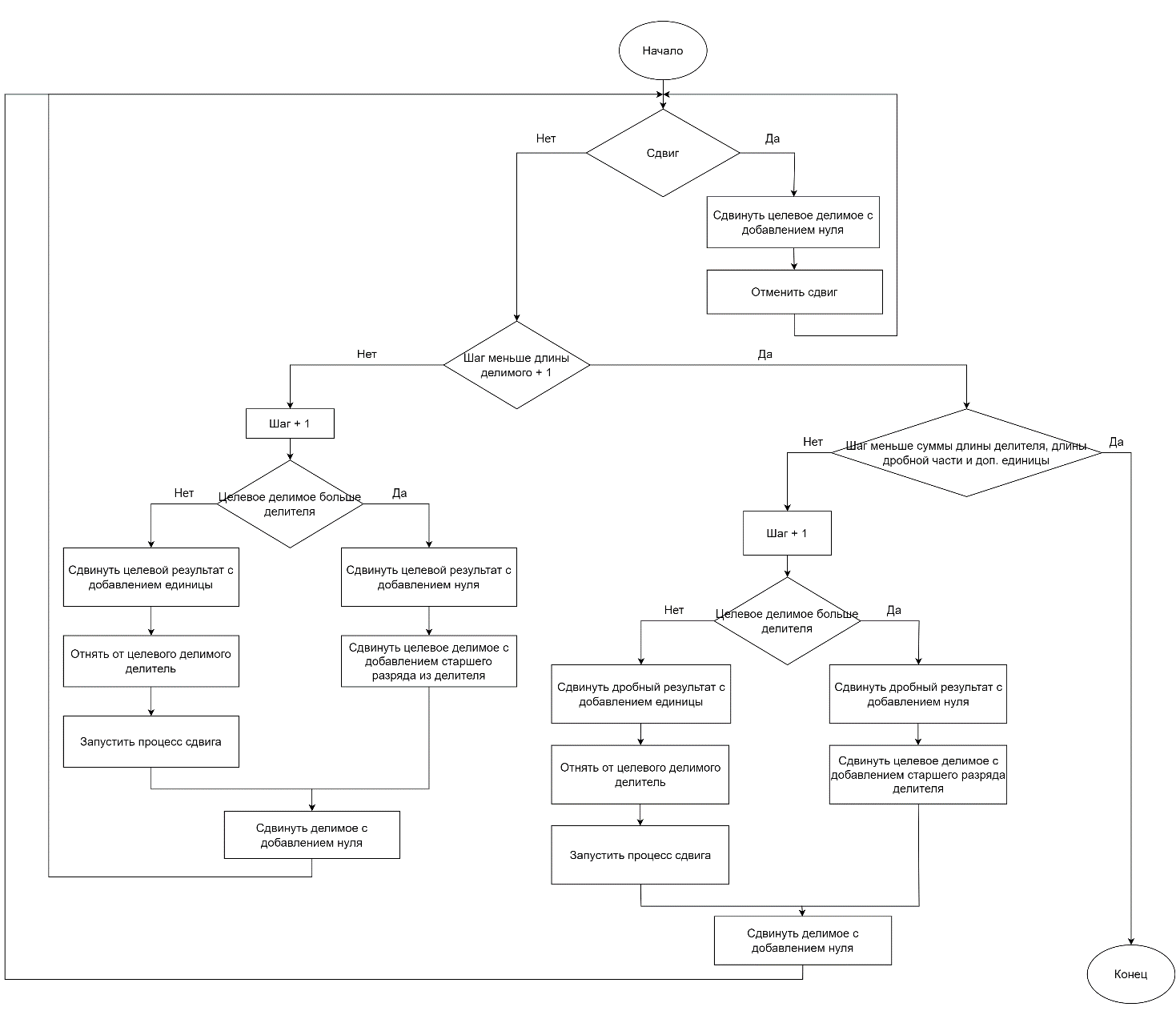


Рисунок 2 – блок схема алгоритма

1. Интерфейс устройства со всеми необходимыми сигналами ввода, вывода и управления представлен на рисунке 3.

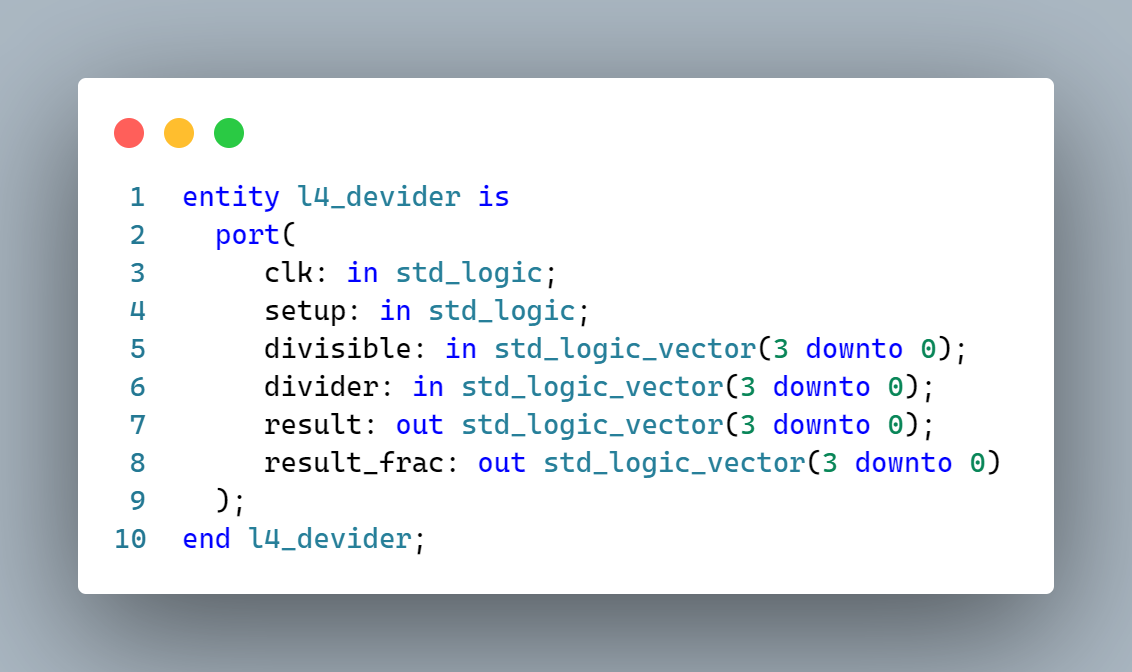


Рисунок 3 – интерфейс делителя

1. Архитектура делителя с разделением на контуры управления и данных представлена листингом 2.

Листинг 2 – архитектура устройства

|  |
| --- |
| library IEEE;  use IEEE.STD\_LOGIC\_1164.ALL;  use IEEE.NUMERIC\_STD.ALL;  architecture Behavioral of l4\_devider is    signal divisible\_prev: std\_logic\_vector(3 downto 0);    signal divisible\_next: std\_logic\_vector(3 downto 0);    signal m\_divisible\_prev: std\_logic\_vector(3 downto 0);    signal m\_divisible\_next: std\_logic\_vector(3 downto 0);    signal result\_prev: std\_logic\_vector(3 downto 0);    signal result\_next: std\_logic\_vector(3 downto 0);    signal float\_result\_prev: std\_logic\_vector(3 downto 0);    signal float\_result\_next: std\_logic\_vector(3 downto 0);    signal step\_prev: unsigned(3 downto 0);    signal step\_next: unsigned(3 downto 0);    signal shift\_prev: std\_logic;    signal shift\_next: std\_logic;  begin    process(clk, setup, divisible) begin      if (setup = '1') then        divisible\_prev <= divisible;        m\_divisible\_prev <= "0000";        result\_prev <= "0000";        float\_result\_prev <= "0000";        step\_prev <= (others => '0');        shift\_prev <= '0';      elsif (clk'event and clk = '1') then        divisible\_prev <= divisible\_next;        m\_divisible\_prev <= m\_divisible\_next;        result\_prev <= result\_next;        float\_result\_prev <= float\_result\_next;        step\_prev <= step\_next;        shift\_prev <= shift\_next;      end if;    end process;    process(divisible\_prev, m\_divisible\_prev, result\_prev, divider) begin      divisible\_next <= divisible\_prev;      m\_divisible\_next <= m\_divisible\_prev;      result\_next <= result\_prev;      float\_result\_next <= float\_result\_prev;      step\_next <= step\_prev;      shift\_next <= shift\_prev;      if (shift\_prev = '1') then        m\_divisible\_next <= m\_divisible\_prev(2 downto 0) & divisible\_prev(3 downto 3);        shift\_next <= '0';      else        if (step\_prev <= 5) then          step\_next <= step\_prev + 1;          if (m\_divisible\_prev >= divider) then            result\_prev <= Result\_prev(2 downto 0) & "1";            m\_divisible\_next <= std\_logic\_vector(unsigned(m\_divisible\_prev) - unsigned(divider));            shift\_next <= '1';          else            result\_prev <= Result\_prev(2 downto 0) & "0";            m\_divisible\_next <= m\_divisible\_prev(2 downto 0) & divisible\_prev(3 downto 3);          end if;          divisible\_next <= divisible\_prev(2 downto 0) & "0";          result <= result\_prev;        elsif (step\_prev <= 9) then          step\_next <= step\_prev + 1;          if (m\_divisible\_prev >= divider) then            float\_result\_next <= float\_result\_prev(2 downto 0) & "1";            m\_divisible\_next <= std\_logic\_vector(unsigned(m\_divisible\_prev(3 downto 0)) - unsigned(divider(3 downto 0)));            shift\_next <= '1';          else            float\_result\_next <= float\_result\_prev(2 downto 0) & "0";            m\_divisible\_next <= m\_divisible\_prev(2 downto 0) & divisible\_prev(3 downto 3);          end if;          divisible\_next <= divisible\_prev(2 downto 0) & "0";          result\_frac <= float\_result\_next;        end if;      end if;    end process;  end Behavioral; |

1. Для тестирования архитектуры был написан Test Bench. Код Test Bench’a представлен на рисунке 4.

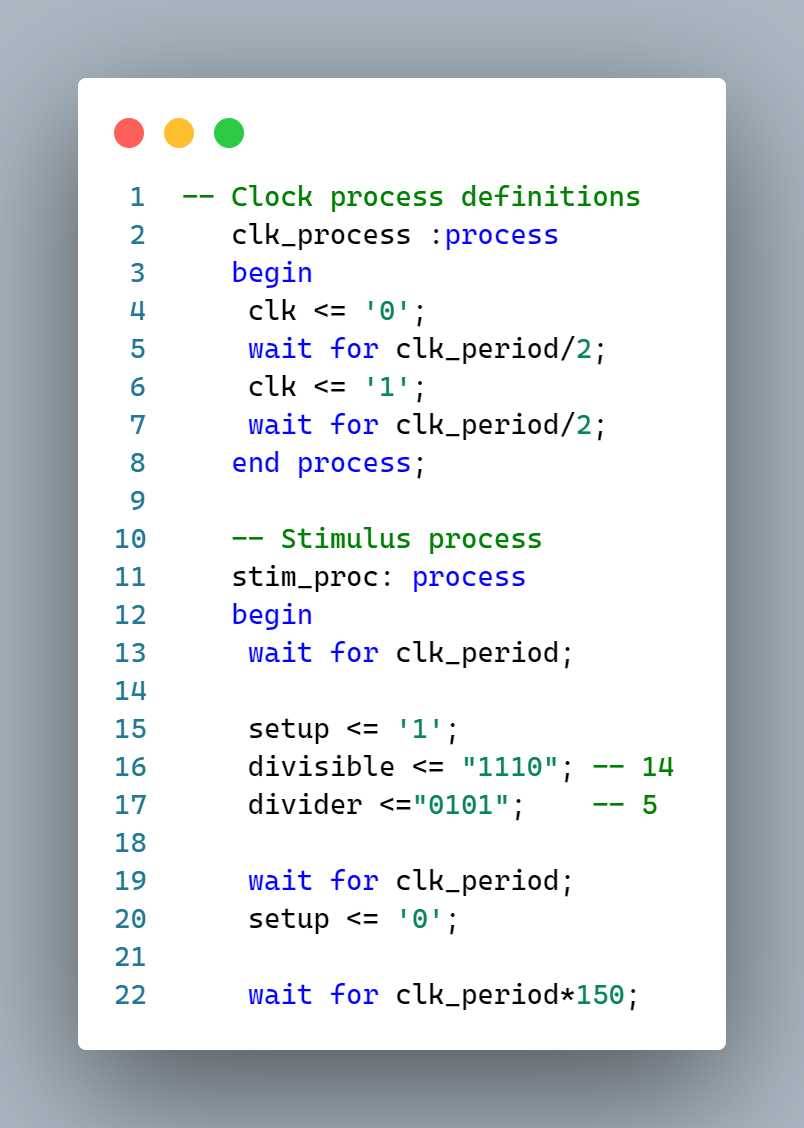


Рисунок 4 – тестовый модуль для делителя

Для тестирования работы устройства поделим divisible (14) на divider (5). Результат должен быть равен , что на диаграмме ниже и было получено.

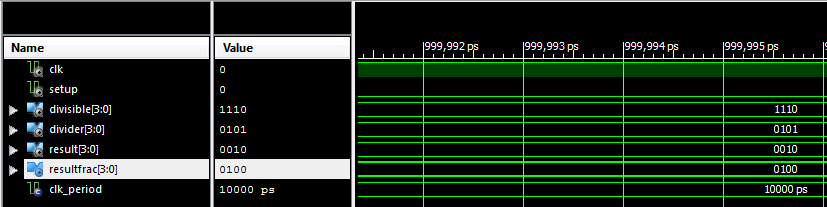


Рисунок 5 – результат симуляции деления 14 на 5

1. Написать вывод о проделанной работе.

# Вывод

В ходе выполнения данной лабораторной работы был разработан и протестирован делитель двух целых чисел на базе программируемой логической интегральной схемы (ПЛИС).

# Использованная литература

1. Потехин Д. С., Тарасов И. Е. Разработка программно-аппаратного обеспечения информационных и автоматизированных систем [Электронный ресурс]:конспект лекций. – М.: РТУ МИРЭА, 2020.

2. Строгонов А. В. Цифровая обработка сигналов в базисе программируемых логических интегральных схем [Электронный ресурс]:. - Санкт-Петербург: Лань, 2021.

3. Зайцев Е. И., Халабия Р. Ф. Организация ЭВМ и систем:учебное пособие. - М.: РТУ МИРЭА, 2019.

4. Певцов Е. Ф., Тарасов И. Е., Миннебаев В. М. Автоматизированное проектирование цифровых схем [Электронный ресурс]:учебное пособие. - М.: МИРЭА, 2016.

5. Красников А. К., Волосенков В. О., Никончук Я. С. Программирование микропроцессорных систем [Электронный ресурс]:учебное пособие. - М.: МГТУ МИРЭА, 2014.

6. Деменкова Т. А., Яровов Д. Д. Аппаратная реализация алгоритмов на FPGA [Электронный ресурс]:монография. - М.: МИРЭА, 2016.