Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и кибербезопасности
Высшая школа компьютерных технологий и информационных систем

Отчёт по лабораторной работе № 8

Дисциплина: Автоматизация проектирования дискретных устройств (на английском языке).

Выполнил студент гр. 5130901/10101 _____ Д.Л. Симоновский
(подпись)

Руководитель _____ А.П. Антонов
(подпись)

“24” марта 2024 г.

Санкт-Петербург
2024

Оглавление

1. **Список иллюстраций: 2**

2. **Алгоритм работы проекта: 3**

3. **Ход работы: 3**

4. **Вывод: 11**

1. Список иллюстраций:

Рис. 1. Схема разрабатываемого устройства. 3
Рис. 2. Добавление папки с IP модулями. 3
Рис. 3. Результат добавления компонентов на схему. 3
Рис. 4. Настройка clk. 4
Рис. 5. Настройка очереди. 4
Рис. 6. Переименование компонентов. 5
Рис. 7. Подключения clk и reset. 5
Рис. 8. Подключение данных. 5
Рис. 9. Результат. 6
Рис. 10. Block Symbol разработанного IP модуля. 6
Рис. 11. Сигналы clk. 7
Рис. 12. Сигналы rst. 7
Рис. 17. Результат Show System with Platform Designer Interconnect. 8
Рис. 21. Schematic. 8

2. Алгоритм работы проекта:

Средствами Platform Designer создать структуру проекта, представленную на рисунке ниже:

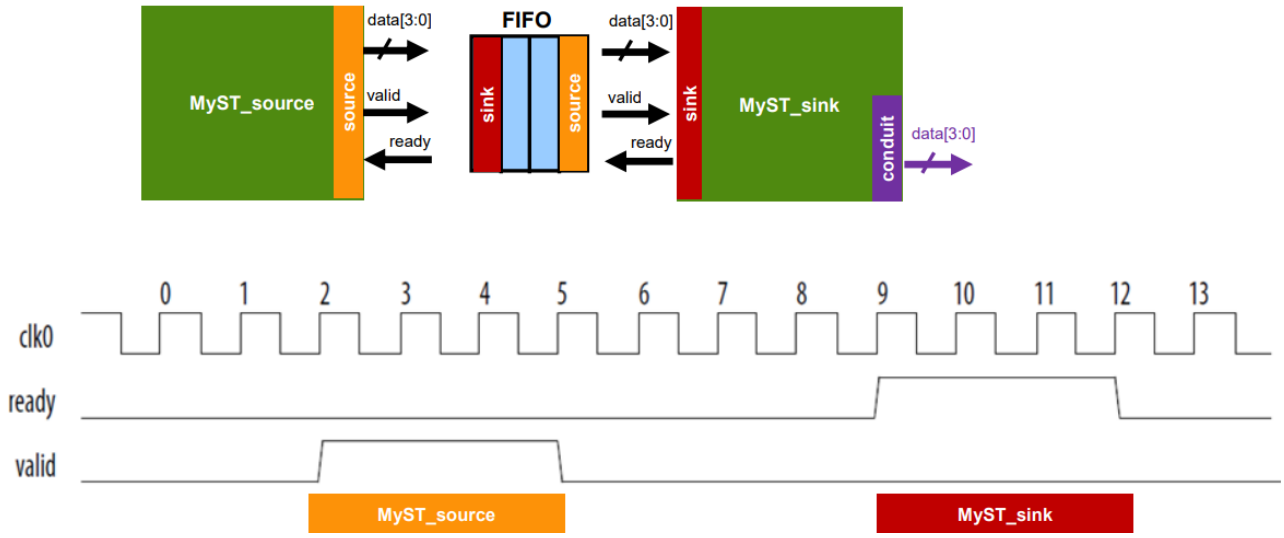

Рис. 1. Схема разрабатываемого устройства.

3. Ход работы:

Выполним создание проекта со стандартными настройками, после чего добавим в проект IP файлы, предоставленные к данной лабораторной работе:

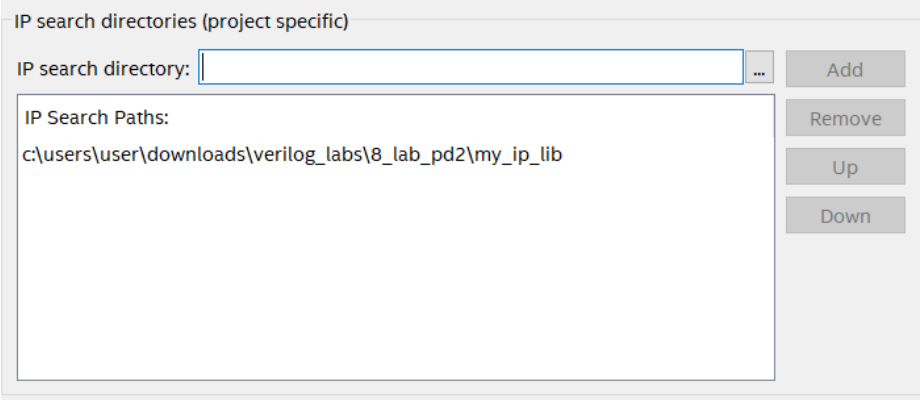

Рис. 2. Добавление папки с IP модулями.

Добавим компоненты: MyST_source_component, Avalon-ST Single Clock FIFO и MyST_sink_component. Таким образом, получим следующий результат:

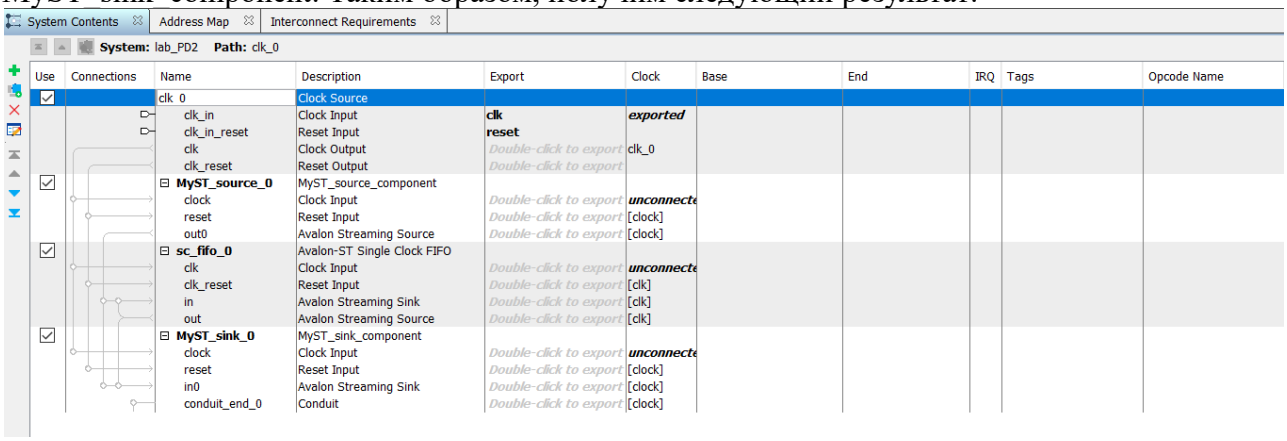

Рис. 3. Результат добавления компонентов на схему.

Настройку CLK модуля:

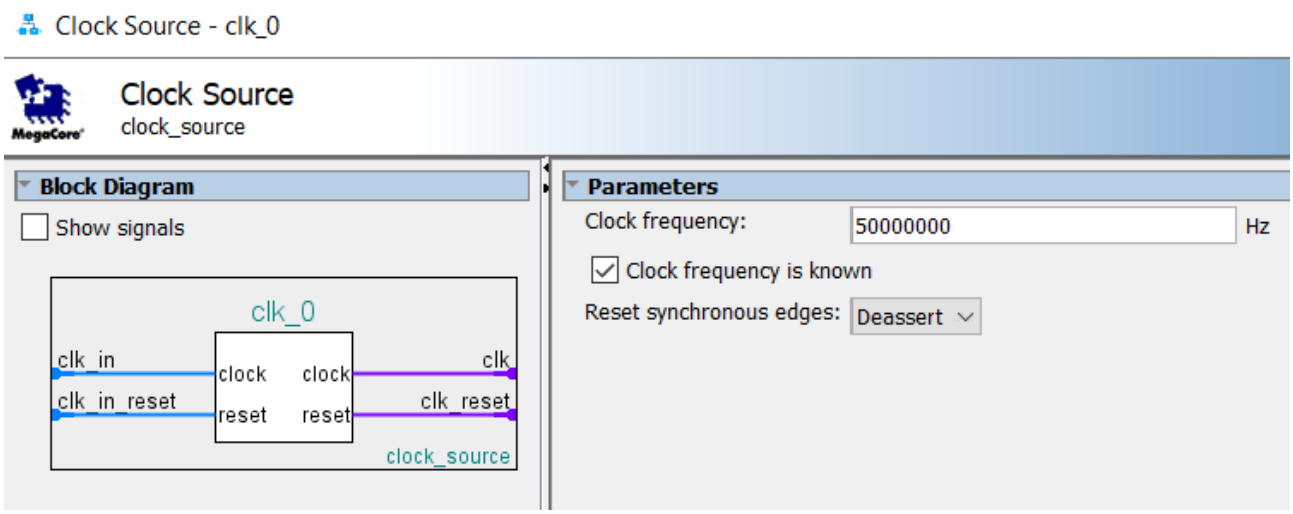

Рис. 4. Настройка clk.

Далее выполним настройку очереди:

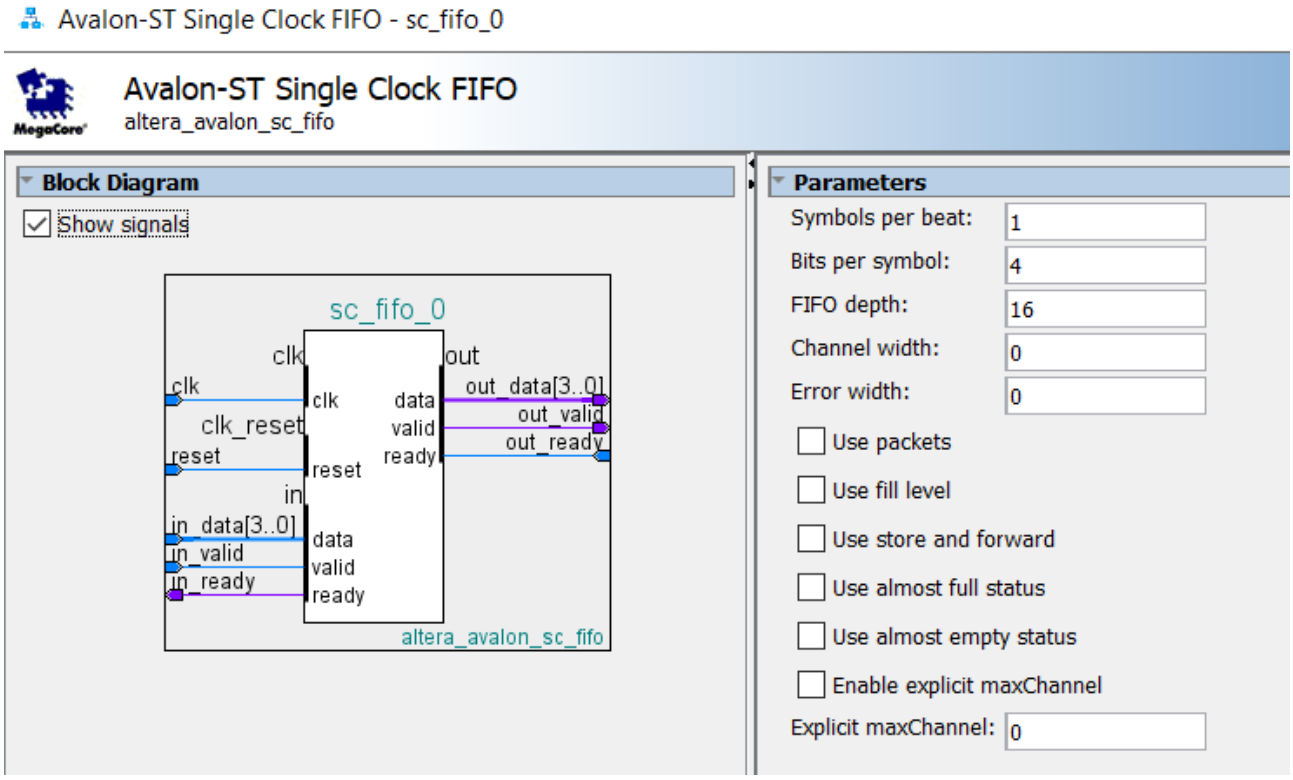

Рис. 5. Настройка очереди.

Переименуем все компоненты и получим следующий результат:

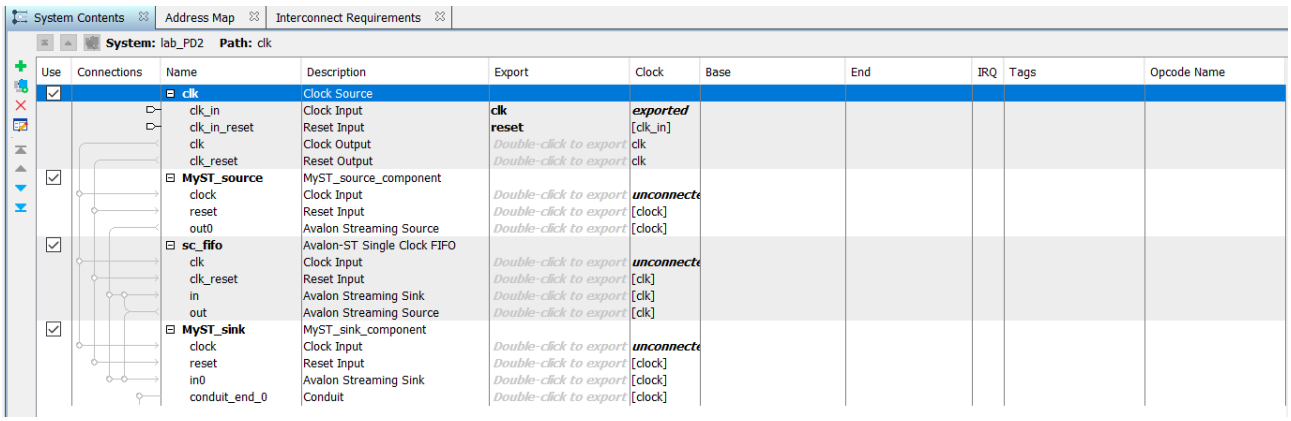

Рис. 6. Переименование компонентов.

Далее выполним подключения сигналов clk и reset:

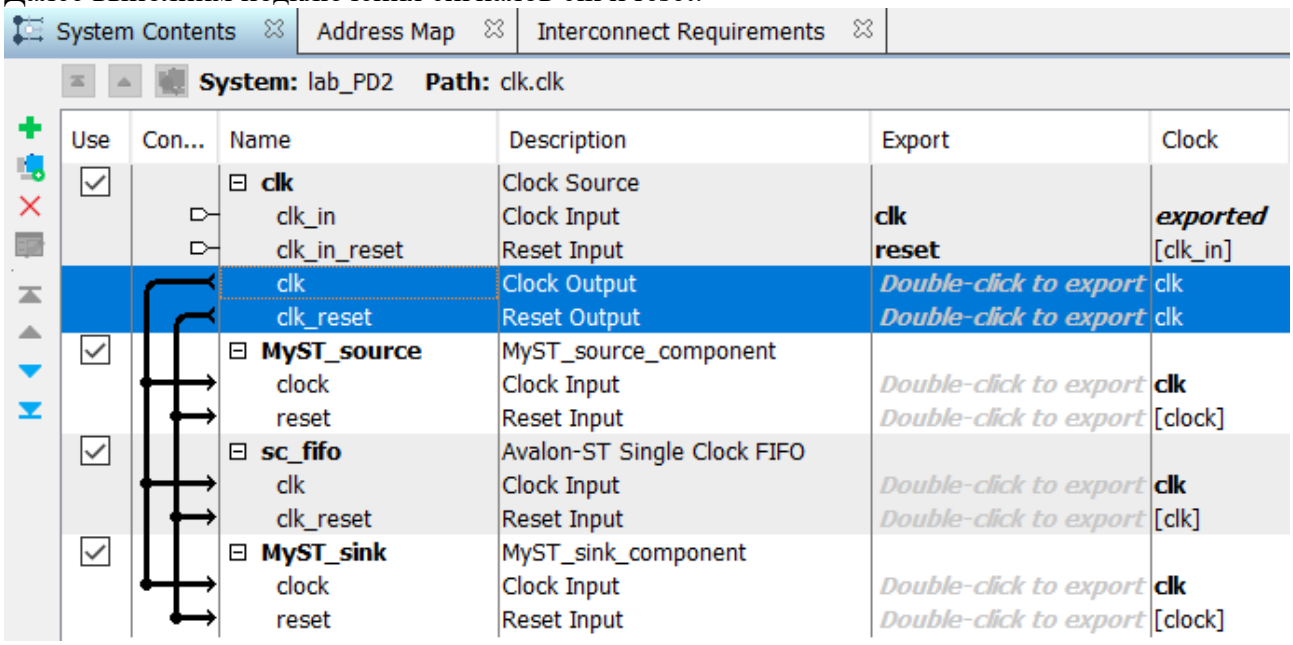

Рис. 7. Подключения clk и reset.

Выполним подключение «шины данных»:

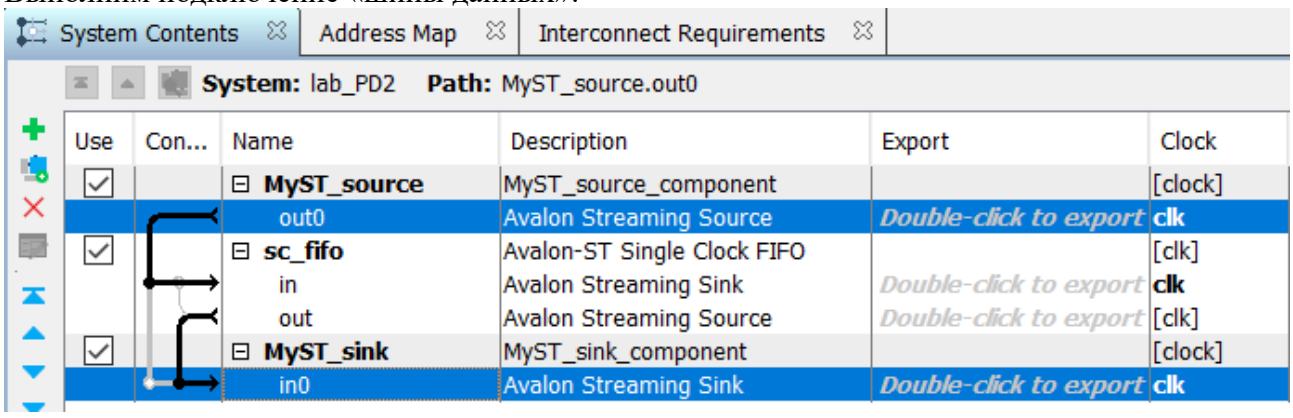

Рис. 8. Подключение данных.

Результат выглядит следующим образом:

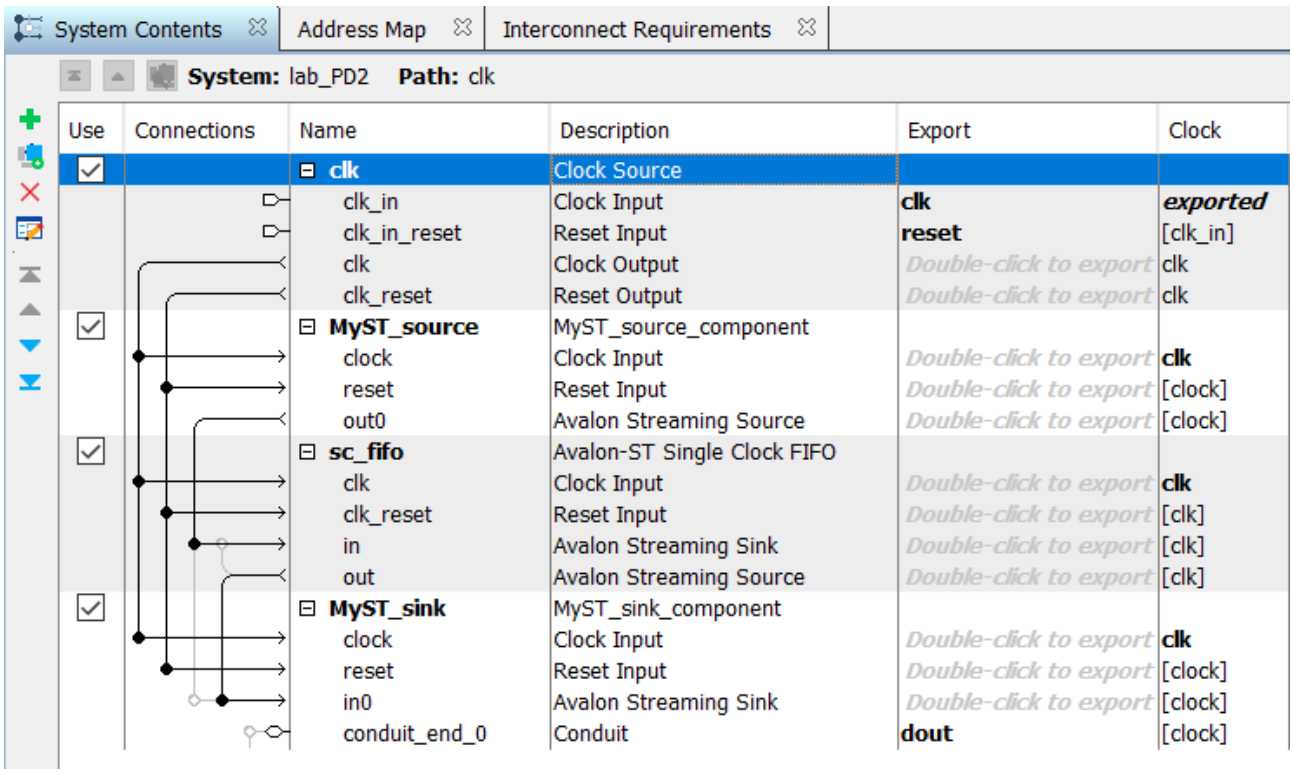

Рис. 9. Результат.

Посмотрим на получившийся Block Symbol:

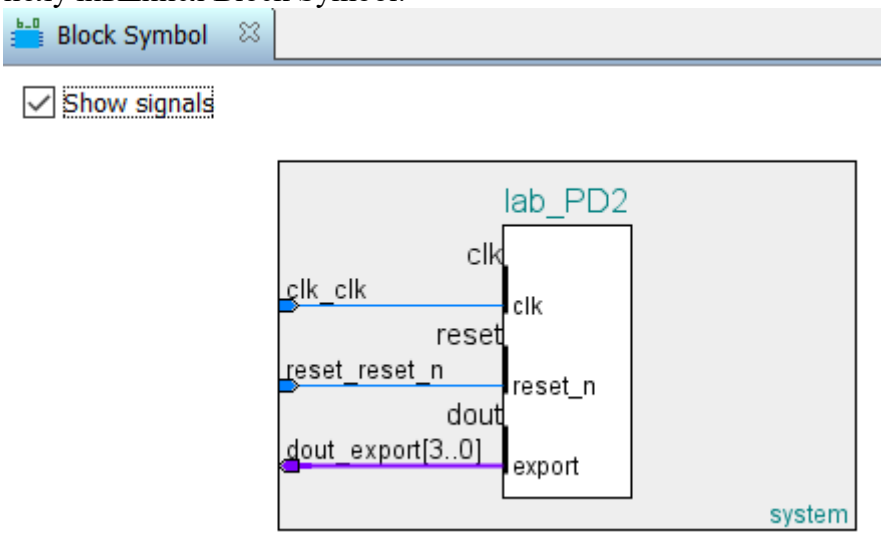

Рис. 10. Block Symbol разработанного IP модуля.

Далее посмотрим на сигналы clk и reset, что не возникает никаких дополнительных контроллеров:

System Contents | Address Map | Interconnect Requirements

System: lab_PD2 **Path:** clk

Use	Connections	Name	Description	Export	Clock
☑		**clk**	Clock Source		
		clk_in	Clock Input	**clk**	***exported***
		clk_in_reset	Reset Input	**reset**	[clk_in]
		clk	Clock Output	*Double-click to export*	clk
		clk_reset	Reset Output	*Double-click to export*	clk
☑		**MyST_source**	MyST_source_component		
		clock	Clock Input	*Double-click to export*	**clk**
		reset	Reset Input	*Double-click to export*	[clock]
		out0	Avalon Streaming Source	*Double-click to export*	[clock]
☑		**sc_fifo**	Avalon-ST Single Clock FIFO		
		clk	Clock Input	*Double-click to export*	**clk**
		clk_reset	Reset Input	*Double-click to export*	[clk]
		in	Avalon Streaming Sink	*Double-click to export*	[clk]
		out	Avalon Streaming Source	*Double-click to export*	[clk]
☑		**MyST_sink**	MyST_sink_component		
		clock	Clock Input	*Double-click to export*	**clk**
		reset	Reset Input	*Double-click to export*	[clock]
		in0	Avalon Streaming Sink	*Double-click to export*	[clock]
		conduit_end_0	Conduit	**dout**	[clock]

Рис. 11. Сигналы clk.

System Contents | Address Map | Interconnect Requirements

System: lab_PD2 **Path:** clk

Use	Connections	Name	Description	Export	Clock
☑		**clk**	Clock Source		
		clk_in	Clock Input	**clk**	***exported***
		clk_in_reset	Reset Input	**reset**	[clk_in]
		clk	Clock Output	*Double-click to export*	clk
		clk_reset	Reset Output	*Double-click to export*	clk
☑		**MyST_source**	MyST_source_component		
		clock	Clock Input	*Double-click to export*	**clk**
		reset	Reset Input	*Double-click to export*	[clock]
		out0	Avalon Streaming Source	*Double-click to export*	[clock]
☑		**sc_fifo**	Avalon-ST Single Clock FIFO		
		clk	Clock Input	*Double-click to export*	**clk**
		clk_reset	Reset Input	*Double-click to export*	[clk]
		in	Avalon Streaming Sink	*Double-click to export*	[clk]
		out	Avalon Streaming Source	*Double-click to export*	[clk]
☑		**MyST_sink**	MyST_sink_component		
		clock	Clock Input	*Double-click to export*	**clk**
		reset	Reset Input	*Double-click to export*	[clock]
		in0	Avalon Streaming Sink	*Double-click to export*	[clock]
		conduit_end_0	Conduit	**dout**	[clock]

Рис. 12. Сигналы rst.

Выполним команду Show System with Platform Designer Interconnect:

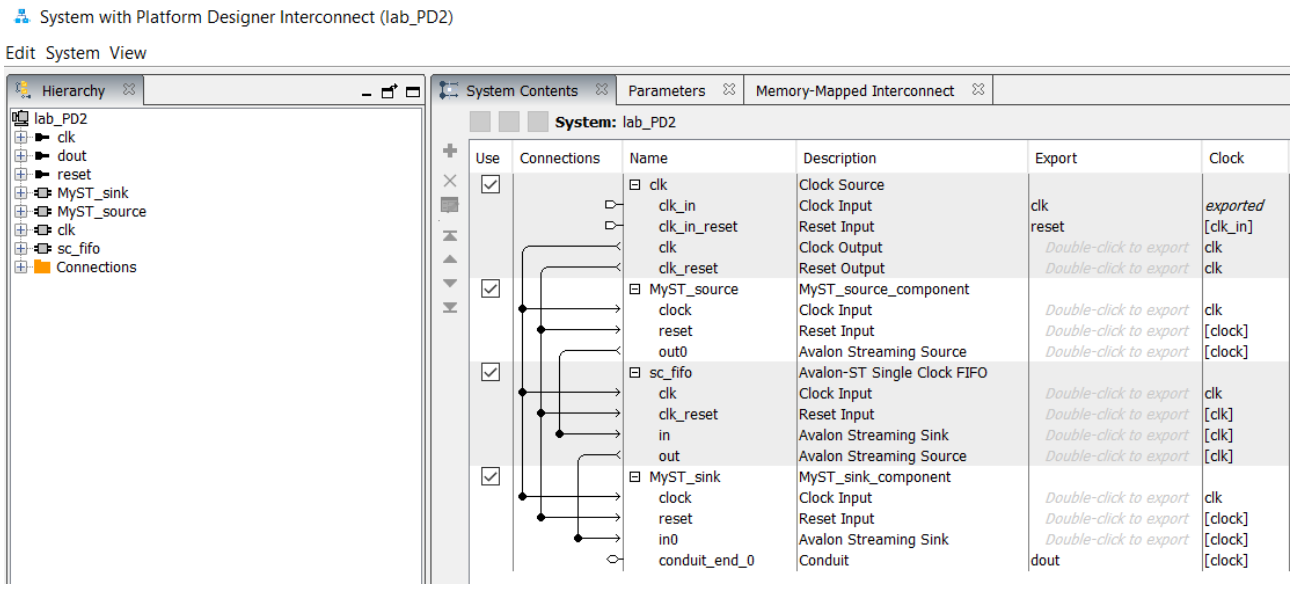

Рис. 13. Результат Show System with Platform Designer Interconnect.

Посмотрим на соединения настроенных модулей в Schematic:

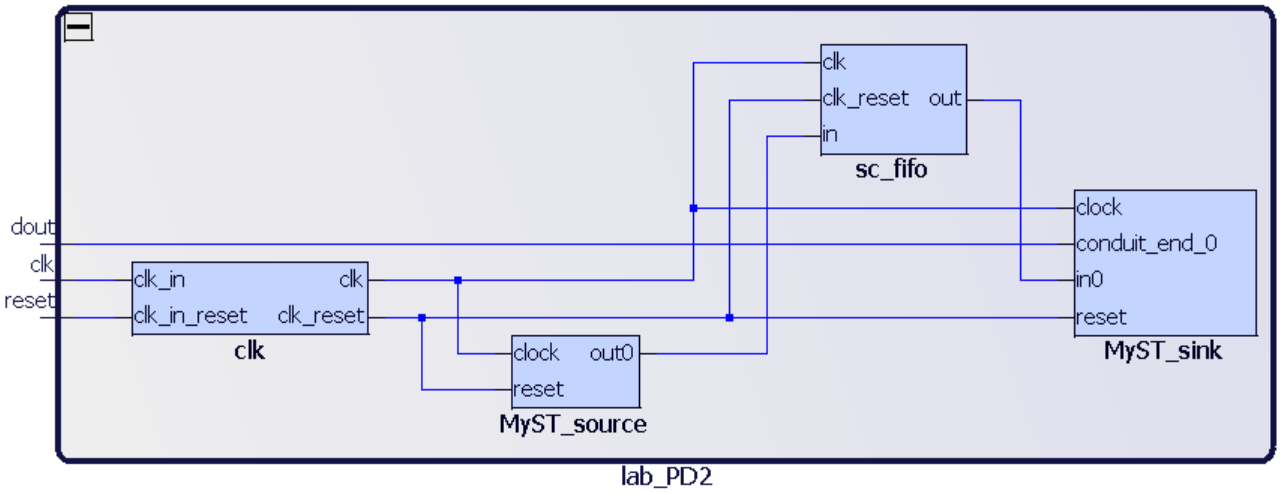

Рис. 14. Schematic.

Добавим созданные модули в проект:

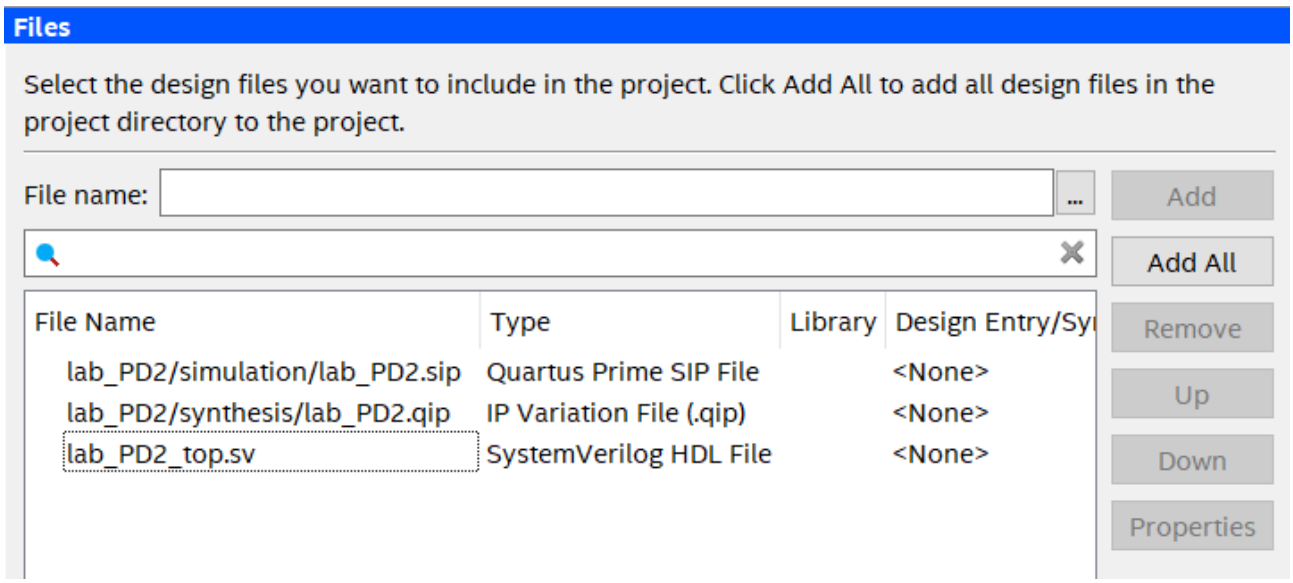

Рис. 15. Добавленные модули к проекту.

Подключим к проекту файл верхнего уровня и выполним компиляцию

```
1  `timescale 1 ns / 1 ns
2  module lab_PD2_top (
3      input bit clk,
4      input bit reset,
5      output bit [3:0] dout
6  );
7  lab_PD2 UUT (
8      .clk_clk       (clk),
9      .reset_reset_n (reset),
10     .dout_export   (dout)
11 );
12 endmodule
```

Посмотрим на RTL Viewer:

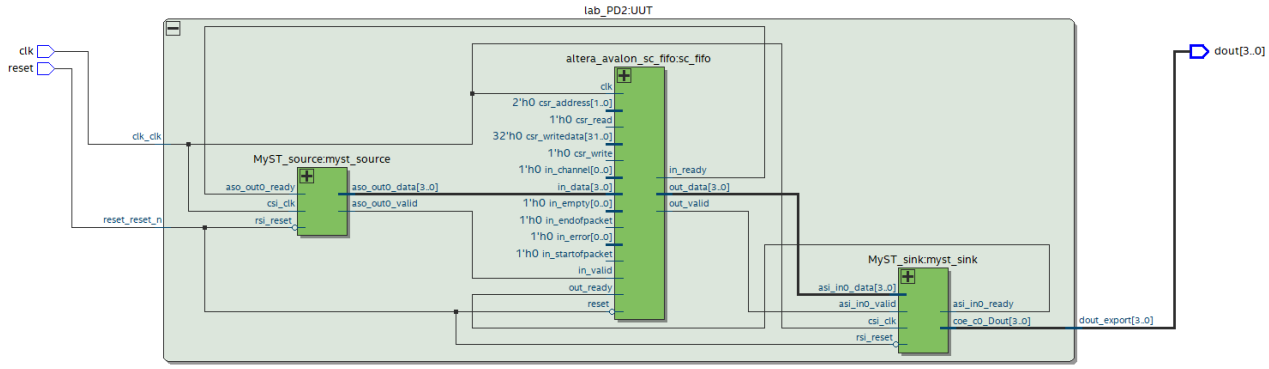

Рис. 16. RTL Viewer устройства.

Протестируем проект, используя следующий тестовый файл:

```
`timescale 1 ns / 1 ns
module tb_lab_PD2_top ();
    bit clk;
    bit reset ;
    bit[3:0] dout;

    always
        #50 clk = ~ clk;
    initial
    begin
        clk     = 1'b0;
        reset   = 1'b0;
        #500;
        reset   = 1'b1;
        #4000;
        $stop;
    end

    lab_PD2_top UUT (.*);
endmodule
```

Результат запуска приведен ниже:

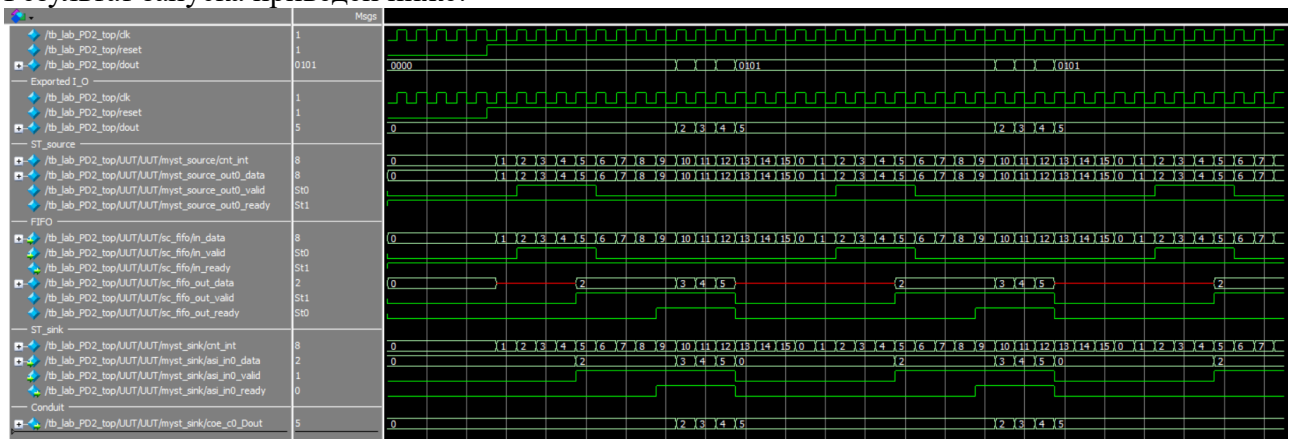

Рис. 17. Результат запуска тестового модуля.

Как мы видим, разработанное устройство соответствует ожиданиям.

Для тестирования на плате создадим следующий модуль, который позволяет изменять rst, используя SP unit:

```
module db_lab_PD2_top (
  (* altera_attribute = "-name IO_STANDARD \"3.3-V LVCMOS\"", chip_pin = "23" *)
  input bit clk
);

  bit       reset;
  bit [3:0] dout;

  SP_unit u0 (
    .source    (reset),
    .source_clk(clk)
  );

  lab_PD2_top UUT (.*);

endmodule
```

Выполним следующую настройку для Signal Tap II:

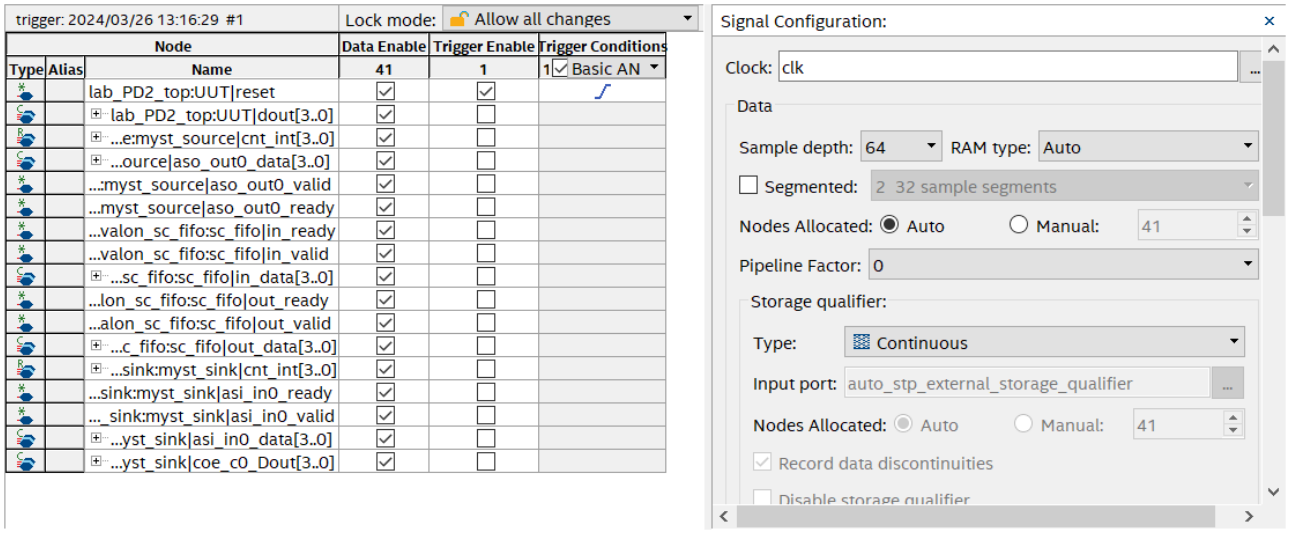

Рис. 18. Настройка для Signal Tap II.

Результат запуска приведен ниже:

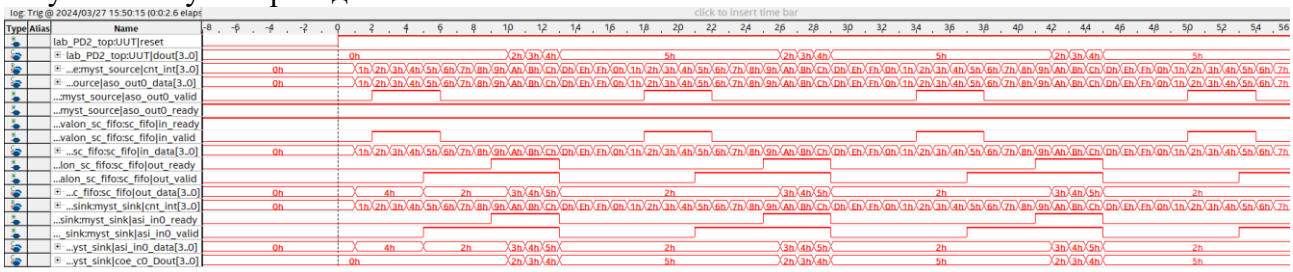

Рис. 19. Signal Tap II.

Как мы видим, результат совпадает с тестами и ожиданиями, что сведительствует о корректности выполненной работы.

4. Вывод:

В ходе выполнения лабораторной работы была создана структура проекта с использованием Platform Designer. Этот инструмент предоставляет графический интерфейс для интеграции и настройки IP-блоков, что упрощает и ускоряет процесс разработки в сравнении с ручным написанием кода.

Преимущества Platform Designer включают в себя:

1. Ускорение разработки: Платформа предлагает готовые IP-блоки, которые могут быть легко интегрированы в проект, сокращая время разработки.
2. Упрощение процесса: Использование графического интерфейса позволяет избежать сложностей в написании и отладке кода, особенно для начинающих разработчиков.
3. Модульность и повторное использование: IP-блоки могут быть использованы в разных проектах, что способствует повторному использованию кода и упрощает поддержку проектов.
4. Визуализация системы: Платформа предоставляет графическое представление структуры проекта, что облегчает понимание и анализ системы.

Таким образом, использование Platform Designer обеспечивает эффективное управление проектом, ускоряет его разработку и повышает его надежность за счет готовых компонентов и удобного интерфейса.