

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. 13. Результат Show System with Platform Designer Interconnect.	8
Рис. 14. Schematic.....	8
Рис. 15. Добавленные модули к проекту.	9
Рис. 16. RTL Viewer устройства.....	9
Рис. 17. Результат запуска тестового модуля.....	10
Рис. 18. Настройка для Signal Tap II.....	11
Рис. 19. Signal Tap II.....	11

2. Алгоритм работы проекта:

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

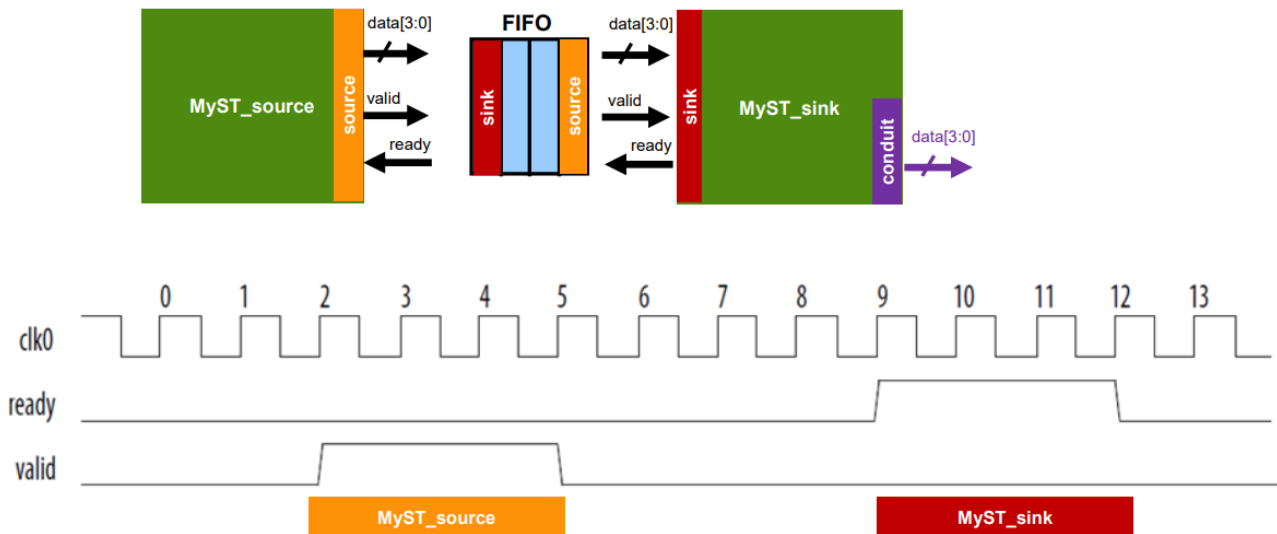


Рис. 1. Схема разрабатываемого устройства.

3. Ход работы:

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

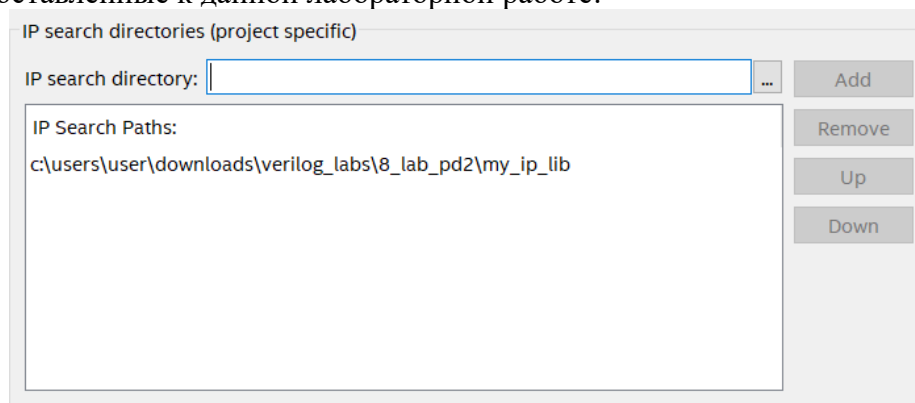


Рис. 2. Добавление папки с IP модулями.

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

Use	Connections	Name	Description	Export	Clock	Base	End	IRQ	Tags	Opcode Name
<input checked="" type="checkbox"/>		clk_0	Clock Source							
		clk_in	Clock Input	clk	exported					
		clk_in_reset	Reset Input	reset						
		clk	Clock Output	Double-click to export	clk_0					
		clk_reset	Reset Output	Double-click to export						
<input checked="" type="checkbox"/>		MyST_source_0	MyST_source_component							
		clock	Clock Input	Double-click to export	unconnecte					
		reset	Reset Input	Double-click to export	[clock]					
		out0	Avalon Streaming Source	Double-click to export	[clock]					
<input checked="" type="checkbox"/>		sc_fifo_0	Avalon-ST Single Clock FIFO							
		clk	Clock Input	Double-click to export	unconnecte					
		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]					
<input checked="" type="checkbox"/>		MyST_sink_0	MyST_sink_component							
		clock	Clock Input	Double-click to export	unconnecte					
		reset	Reset Input	Double-click to export	[clock]					
		in0	Avalon Streaming Sink	Double-click to export	[clock]					
		conduit_end_0	Conduit	Double-click to export	[clock]					

Рис. 3. Результат добавления компонентов на схему.

Настройку CLK модуля:

Clock Source
clock_source

Block Diagram
☐ Show signals

Parameters
Clock frequency: 50000000 Hz
☒ Clock frequency is known
Reset synchronous edges: Deassert

Рис. 4. Настройка clk.

Далее выполним настройку очереди:

Avalon-ST Single Clock FIFO
altera_avalon_sc_fifo

Block Diagram
☒ Show signals

Parameters
Symbols per beat: 1
Bits per symbol: 4
FIFO depth: 16
Channel width: 0
Error width: 0
☐ Use packets
☐ Use fill level
☐ Use store and forward
☐ Use almost full status
☐ Use almost empty status
☐ Enable explicit maxChannel
Explicit maxChannel: 0

Рис. 5. Настройка очереди.

Переименуем все компоненты и получим следующий результат:

System Contents Address Map Interconnect Requirements System: lab_PD2 Path: clk										
Use	Connections	Name	Description	Export	Clock	Base	End	IRQ	Tags	Opcode Name
<input checked="" type="checkbox"/>		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					
<input checked="" type="checkbox"/>		MyST_source	MyST_source_component							
		clock	Clock Input	Double-click to export	unconnecte					
		reset	Reset Input	Double-click to export	[clock]					
		out0	Avalon Streaming Source	Double-click to export	[clock]					
<input checked="" type="checkbox"/>		sc_fifo	Avalon-ST Single Clock FIFO							
		clk	Clock Input	Double-click to export	unconnecte					
		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]					
<input checked="" type="checkbox"/>		MyST_sink	MyST_sink_component							
		clock	Clock Input	Double-click to export	unconnecte					
		reset	Reset Input	Double-click to export	[clock]					
		in0	Avalon Streaming Sink	Double-click to export	[clock]					
		conduit_end_0	Conduit	Double-click to export	[clock]					

Рис. 6. Переименование компонентов.

Далее выполним подключения сигналов clk и reset:

System Contents							Address Map		Interconnect Requirements	
System: lab_PD2							Path: clk.clk			
Use	Con...	Name	Description	Export	Clock					
<input checked="" type="checkbox"/>		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					
<input checked="" type="checkbox"/>		MyST_source	MyST_source_component							
		clock	Clock Input	Double-click to export	clk					
		reset	Reset Input	Double-click to export	[clock]					
<input checked="" type="checkbox"/>		sc_fifo	Avalon-ST Single Clock FIFO							
		clk	Clock Input	Double-click to export	clk					
		clk_reset	Reset Input	Double-click to export	[clk]					
<input checked="" type="checkbox"/>		MyST_sink	MyST_sink_component							
		clock	Clock Input	Double-click to export	clk					
		reset	Reset Input	Double-click to export	[clock]					

Рис. 7. Подключения clk и reset.

Выполним подключение «шины данных»:

System Contents Address Map Interconnect Requirements

System: lab_PD2 Path: MyST_source.out0

Use	Con...	Name	Description	Export	Clock
<input checked="" type="checkbox"/>		MyST_source	MyST_source_component		[clock]
		out0	Avalon Streaming Source	Double-click to export	clk
<input checked="" type="checkbox"/>		sc_fifo	Avalon-ST Single Clock FIFO		[clk]
		in	Avalon Streaming Sink	Double-click to export	clk
		out	Avalon Streaming Source	Double-click to export	[clk]
<input checked="" type="checkbox"/>		MyST_sink	MyST_sink_component		[clock]
		in0	Avalon Streaming Sink	Double-click to export	clk

Рис. 8. Подключение данных.

Результат выглядит следующим образом:

System Contents Address Map Interconnect Requirements					
System: lab_PD2 Path: clk					
Use	Connections	Name	Description	Export	Clock
<input checked="" type="checkbox"/>		clk	Clock Source		
		clk_in	Clock Input	clk	exported
		clk_in_reset	Reset Input	reset	[clk_in]
		clk	Clock Output	<i>Double-click to export</i>	clk
		clk_reset	Reset Output	<i>Double-click to export</i>	clk
<input checked="" type="checkbox"/>		MyST_source	MyST_source_component		
		clock	Clock Input	<i>Double-click to export</i>	clk
		reset	Reset Input	<i>Double-click to export</i>	[clock]
		out0	Avalon Streaming Source	<i>Double-click to export</i>	[clock]
<input checked="" type="checkbox"/>		sc_fifo	Avalon-ST Single Clock FIFO		
		clk	Clock Input	<i>Double-click to export</i>	clk
		clk_reset	Reset Input	<i>Double-click to export</i>	[clk]
		in	Avalon Streaming Sink	<i>Double-click to export</i>	[clk]
		out	Avalon Streaming Source	<i>Double-click to export</i>	[clk]
<input checked="" type="checkbox"/>		MyST_sink	MyST_sink_component		
		clock	Clock Input	<i>Double-click to export</i>	clk
		reset	Reset Input	<i>Double-click to export</i>	[clock]
		in0	Avalon Streaming Sink	<i>Double-click to export</i>	[clock]
		conduit_end_0	Conduit	dout	[clock]

Рис. 9. Результат.

Посмотрим на получившийся Block Symbol:

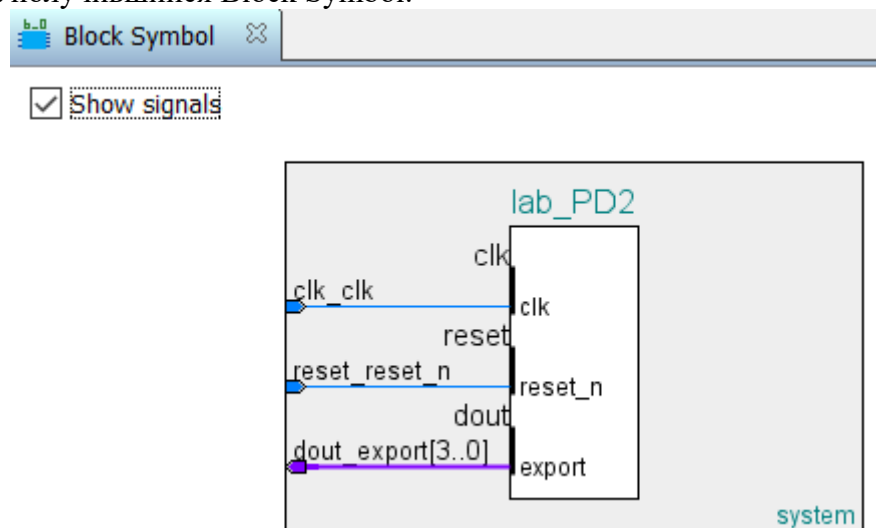


Рис. 10. Block Symbol разработанного IP модуля.

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
<input checked="" type="checkbox"/>		clk	Clock Source		
		clk_in	Clock Input	clk	exported
		clk_in_reset	Reset Input	reset	[clk_in]
		clk	Clock Output	<i>Double-click to export</i>	clk
		clk_reset	Reset Output	<i>Double-click to export</i>	clk
<input checked="" type="checkbox"/>		MyST_source	MyST_source_component		
		clock	Clock Input	<i>Double-click to export</i>	clk
		reset	Reset Input	<i>Double-click to export</i>	[clock]
		out0	Avalon Streaming Source	<i>Double-click to export</i>	[clock]
<input checked="" type="checkbox"/>		sc_fifo	Avalon-ST Single Clock FIFO		
		clk	Clock Input	<i>Double-click to export</i>	clk
		clk_reset	Reset Input	<i>Double-click to export</i>	[clk]
		in	Avalon Streaming Sink	<i>Double-click to export</i>	[clk]
		out	Avalon Streaming Source	<i>Double-click to export</i>	[clk]
<input checked="" type="checkbox"/>		MyST_sink	MyST_sink_component		
		clock	Clock Input	<i>Double-click to export</i>	clk
		reset	Reset Input	<i>Double-click to export</i>	[clock]
		in0	Avalon Streaming Sink	<i>Double-click to export</i>	[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
<input checked="" type="checkbox"/>		clk	Clock Source		
		clk_in	Clock Input	clk	exported
		clk_in_reset	Reset Input	reset	[clk_in]
		clk	Clock Output	<i>Double-click to export</i>	clk
		clk_reset	Reset Output	<i>Double-click to export</i>	clk
<input checked="" type="checkbox"/>		MyST_source	MyST_source_component		
		clock	Clock Input	<i>Double-click to export</i>	clk
		reset	Reset Input	<i>Double-click to export</i>	[clock]
		out0	Avalon Streaming Source	<i>Double-click to export</i>	[clock]
<input checked="" type="checkbox"/>		sc_fifo	Avalon-ST Single Clock FIFO		
		clk	Clock Input	<i>Double-click to export</i>	clk
		clk_reset	Reset Input	<i>Double-click to export</i>	[clk]
		in	Avalon Streaming Sink	<i>Double-click to export</i>	[clk]
		out	Avalon Streaming Source	<i>Double-click to export</i>	[clk]
<input checked="" type="checkbox"/>		MyST_sink	MyST_sink_component		
		clock	Clock Input	<i>Double-click to export</i>	clk
		reset	Reset Input	<i>Double-click to export</i>	[clock]
		in0	Avalon Streaming Sink	<i>Double-click to export</i>	[clock]
		conduit_end_0	Conduit	dout	[clock]

Рис. 12. Сигналы rst.

Выполним команду Show System with Platform Designer Interconnect:

Edit System View

System: lab_PD2					
Use	Connections	Name	Description	Export	Clock
<input checked="" type="checkbox"/>		clk	Clock Source	clk	exported
		clk_in	Clock Input		[clk_in]
		clk_in_reset	Reset Input	reset	clk
		clk	Clock Output	Double-click to export	clk
		clk_reset	Reset Output	Double-click to export	clk
<input checked="" type="checkbox"/>		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]
<input checked="" type="checkbox"/>		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]
<input checked="" type="checkbox"/>		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	Double-click to export	[clock]
				dout	[clock]

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

Посмотрим на соединения настроенных модулей в Schematic:

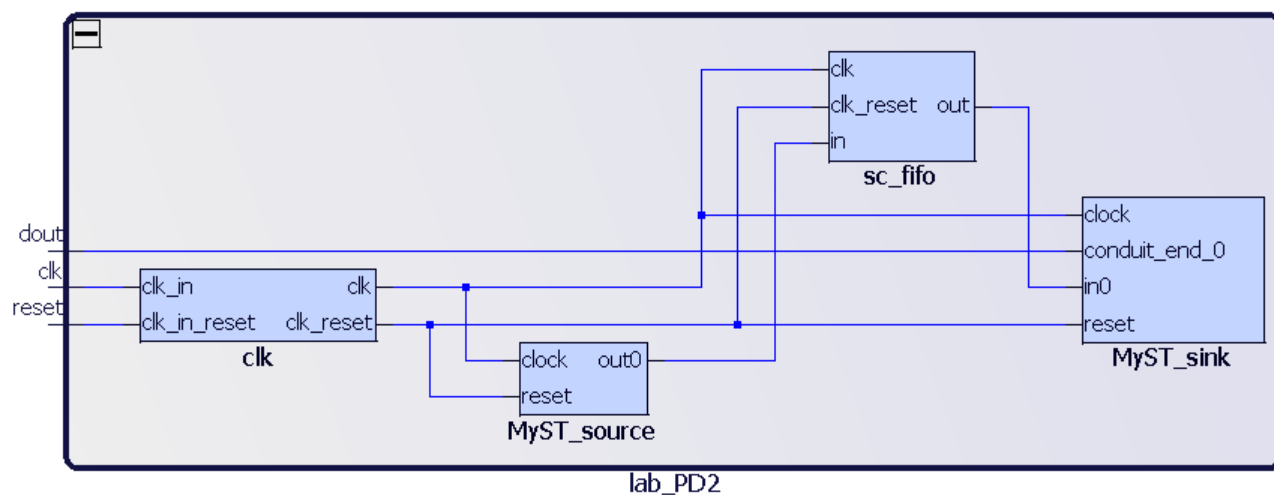


Рис. 14. Schematic.

Добавим созданные модули в проект:

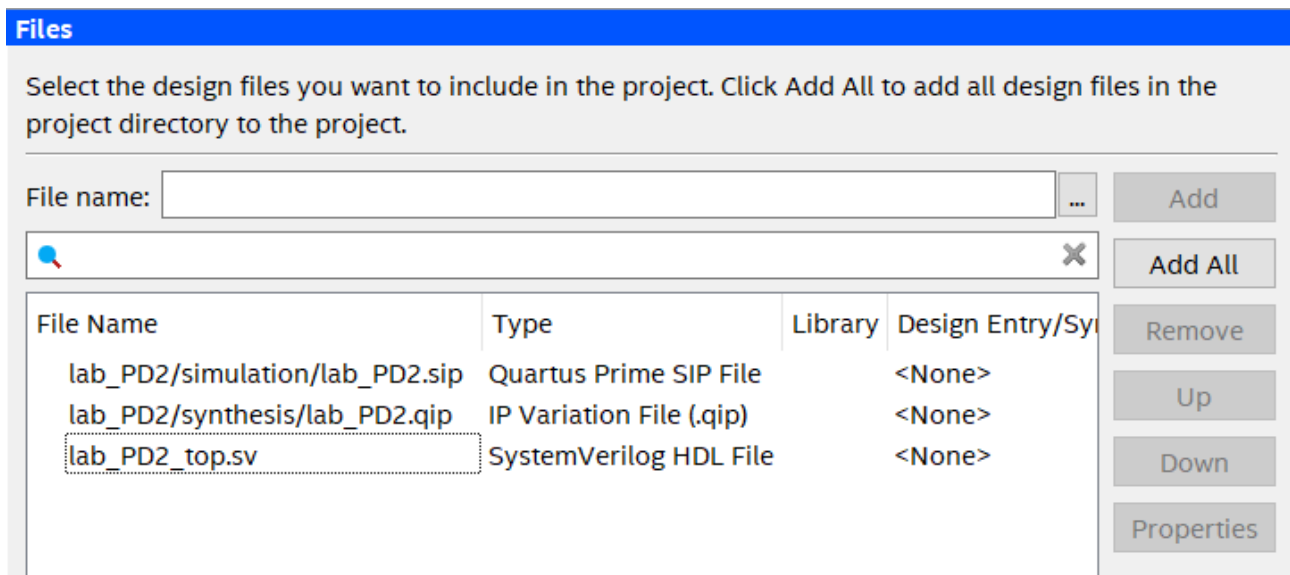


Рис. 15. Добавленные модули к проекту.

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:

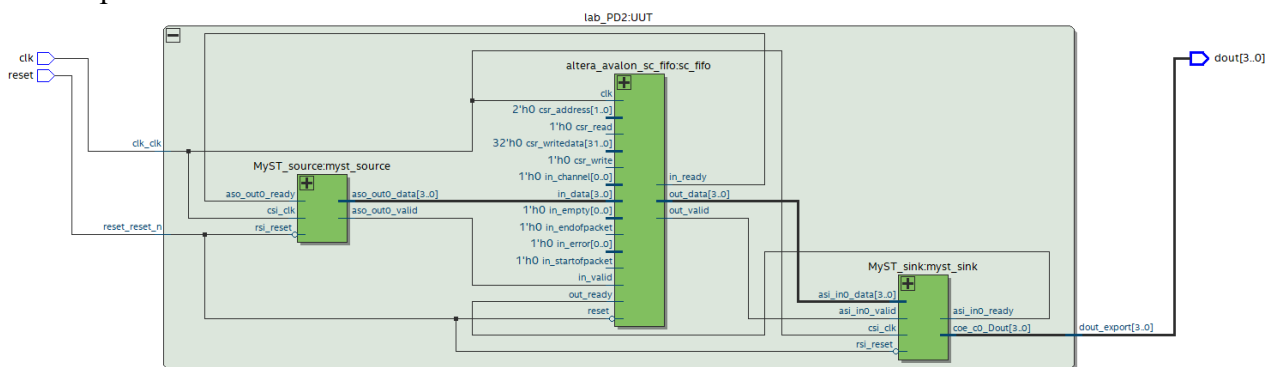


Рис. 16. RTL Viewer устройства.

Протестируем проект, используя следующий тестовый файл:

```

1 `timescale 1 ns / 1 ns
2 module tb_lab_PD2_top ();
3     bit clk;
4     bit reset ;
5     bit[3:0] dout;
6
7     always
8         #50 clk = ~ clk;
9     initial
10    begin
11        clk      = 1'b0;
12        reset    = 1'b0;
13        #500;
14        reset    = 1'b1;
15        #4000;
16        $stop;
17    end
18
19    lab_PD2_top UUT (.*);
20 endmodule

```

Результат запуска приведен ниже:

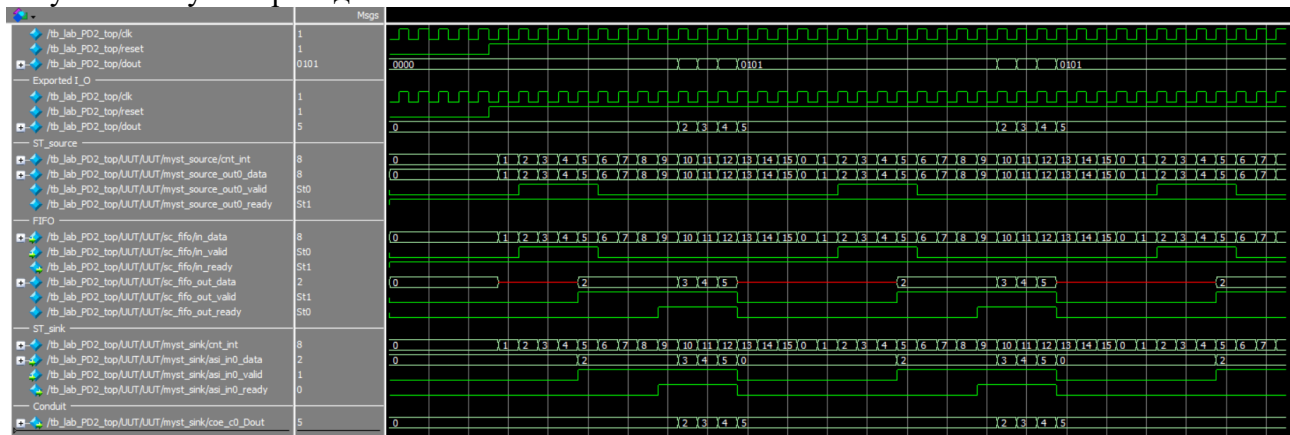


Рис. 17. Результат запуска тестового модуля.

Как мы видим, разработанное устройство соответствует ожиданиям.

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

```

1 module db_lab_PD2_top (
2     (* altera_attribute = "-name IO_STANDARD \"3.3-V LVC MOS\"", chip_pin = "23" *)
3     input bit clk
4 );
5
6     bit      reset;
7     bit [3:0] dout;
8
9     SP_unit u0 (
10        .source      (reset),
11        .source_clk   (clk)
12    );
13
14    lab_PD2_top UUT (.*);
15
16 endmodule

```

Выполним следующую настройку для Signal Tap II:

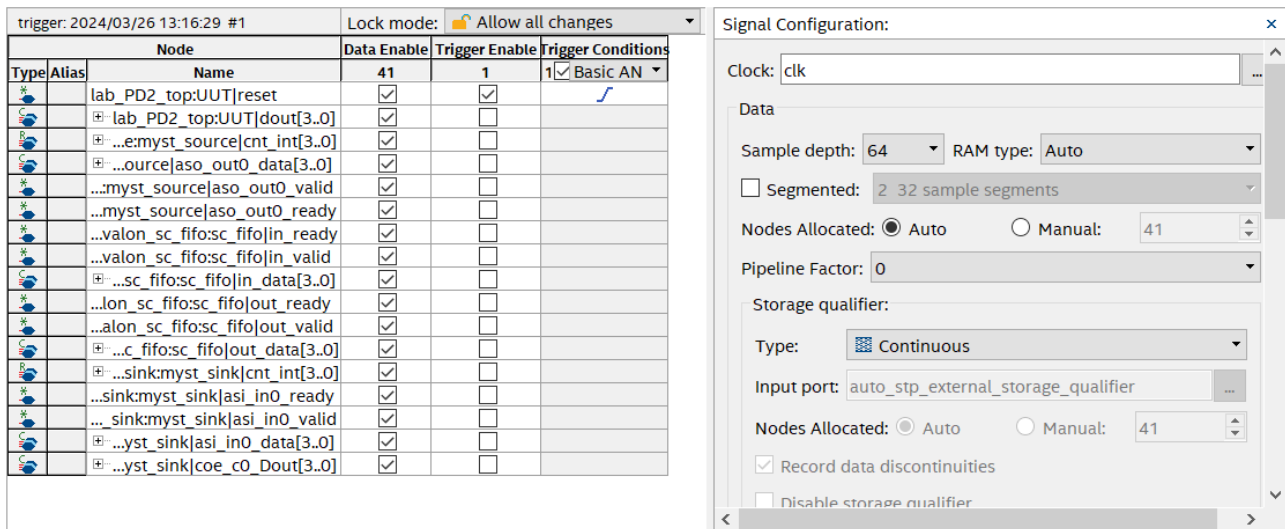


Рис. 18. Настройка для Signal Tap II.

Результат запуска приведен ниже:



Рис. 19. Signal Tap II.

Как мы видим, результат совпадает с тестами и ожиданиями, что свидетельствует 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.