

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

Отчёт по лабораторной работе № 9

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

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

Руководитель _____ А.П. Антонов
(подпись)

“07” апреля 2024 г.

Санкт-Петербург

2024

Оглавление

1.	Список иллюстраций:	2
2.	Алгоритм работы проекта:	3
3.	Ход работы:.....	3
4.	Вывод:.....	12

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. Подключения clk и reset.	4
Рис. 6. Подключение данных.	5
Рис. 7. Настройка адресов.	5
Рис. 8. Результат.....	5
Рис. 9. Настройка my_master.	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. RTL Viewer.....	10
Рис. 19. Signal Tap II.....	11
Рис. 20. Обновленная настройка my_master.	11
Рис. 21. Обновленная настройка адресов в Qsys.	11
Рис. 22. Запуск тестового файла.	12
Рис. 23. Результат запуска в Signal Tap II.....	12

2. Алгоритм работы проекта:

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

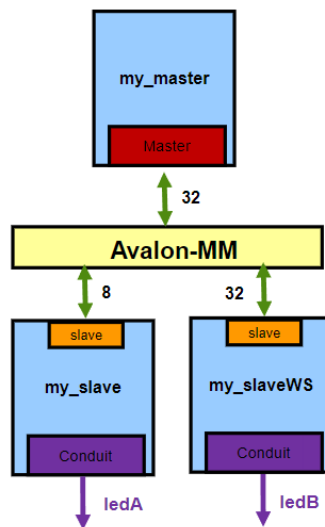


Рис. 1. Схема разрабатываемого устройства.

3. Ход работы:

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

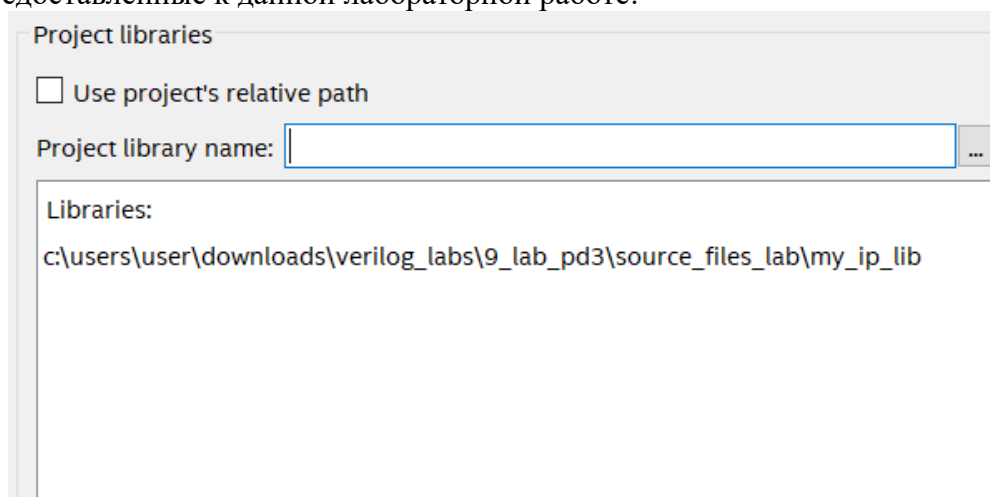



Рис. 2. Добавление папки с IP модулями.

Добавим компоненты: my_master, my_slave, my_slaveWS. Таким образом, получим следующий результат:

System: lab_PD3 Path: clk					
Use	Connections	Name	Description	Export	Clock
<input checked="" type="checkbox"/>		clk	Clock Source		
<input checked="" type="checkbox"/>		clk_in	Clock Input	clk	
<input checked="" type="checkbox"/>		clk_in_reset	Reset Input	reset	
<input checked="" type="checkbox"/>		clk	Clock Output	Double-click to export	clk
<input checked="" type="checkbox"/>		clk_reset	Reset Output	Double-click to export	
<input checked="" type="checkbox"/>		my_master	my_master	Double-click to export	unconnected
<input checked="" type="checkbox"/>		clock	Clock Input	Double-click to export	[clock]
<input checked="" type="checkbox"/>		reset	Reset Input	Double-click to export	[clock]
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	Double-click to export	[clock]
<input checked="" type="checkbox"/>		my_slave	my_slave	Double-click to export	unconnected
<input checked="" type="checkbox"/>		clock	Clock Input	Double-click to export	[clock]
<input checked="" type="checkbox"/>		reset	Reset Input	Double-click to export	[clock]
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to export	[clock]
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	Double-click to export	[clock]
<input checked="" type="checkbox"/>		my_slaveWS	my_slaveWS	Double-click to export	unconnected
<input checked="" type="checkbox"/>		clock	Clock Input	Double-click to export	[clock]
<input checked="" type="checkbox"/>		reset	Reset Input	Double-click to export	[clock]
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to export	[clock]
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	Double-click to export	[clock]

Рис. 3. Результат добавления компонентов на схему.

Настройку CLK модуля:

 Clock Source - clk

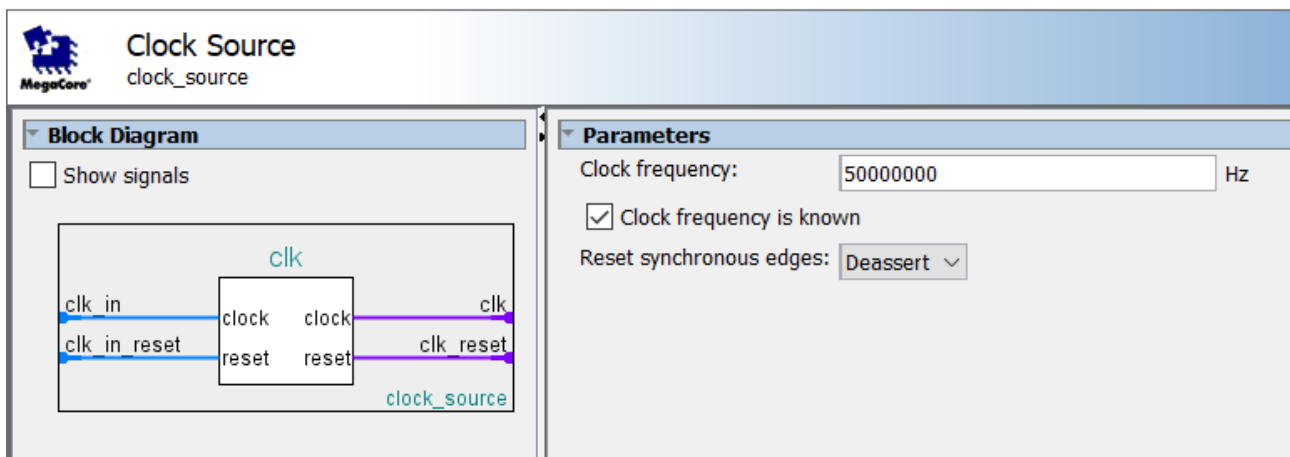


Рис. 4. Настройка clk.

Далее выполним подключения сигналов clk и reset:

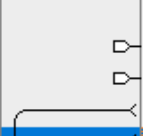



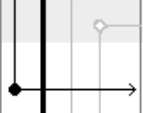
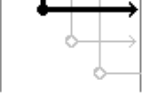
System: lab_PD3 Path: clk.clk_reset						
Use	Connections	Name	Description	Export	Clock	
<input checked="" type="checkbox"/>		<div> <div>clk</div> <div>clk_in</div> <div>clk_in_reset</div> <div>clk</div> </div>	<div>Clock Source</div> <div>Clock Input</div> <div>Reset Input</div> <div>Clock Output</div>	<div>clk</div> <div>reset</div> <div>Double-click to export</div>	<div>exported</div> <div>[clk_in]</div> <div>clk</div>	
<input checked="" type="checkbox"/>		<div>clk_reset</div>	Reset Output	Double-click to export	clk	
<input checked="" type="checkbox"/>		<div>my_master</div> <div>clock</div> <div>reset</div>	<div>my_master</div> <div>Clock Input</div> <div>Reset Input</div>	<div>Double-click to export</div> <div>Double-click to export</div>	<div>clk</div> <div>[clock]</div>	
<input checked="" type="checkbox"/>		<div>my_slave</div> <div>clock</div> <div>reset</div>	<div>my_slave</div> <div>Clock Input</div> <div>Reset Input</div>	<div>Double-click to export</div> <div>Double-click to export</div>	<div>clk</div> <div>[clock]</div>	
<input checked="" type="checkbox"/>		<div>my_slaveWS</div> <div>s0</div> <div>conduit_end_0</div>	<div>Avalon Memory Mapped Master</div> <div>Avalon Memory Mapped Slave</div> <div>Conduit</div>	<div>Double-click to export</div> <div>Double-click to export</div>	<div>clk</div> <div>[clock]</div>	
<input checked="" type="checkbox"/>		<div>my_slaveWS</div> <div>clock</div> <div>reset</div>	<div>my_slaveWS</div> <div>Clock Input</div> <div>Reset Input</div>	<div>Double-click to export</div> <div>Double-click to export</div>	<div>clk</div> <div>[clock]</div>	

Рис. 5. Подключения clk и reset.

Выполним подключение «шины данных»:

Use	C...	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		my_master	my_master		[clock]		
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	Double-click to export	clk		
<input checked="" type="checkbox"/>		my_slave	my_slave		[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to export	clk	0x0000_0000	0x0000_0000
<input checked="" type="checkbox"/>		my_slaveWS	my_slaveWS		[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to export	clk	0x0000_0000	0x0000_0003

Type	Path	Message
1 Error		
	lab_PD3.my_master.m0	my_slaveWS.s0 (0x0..0x3) overlaps my_slave.s0 (0x0..0x0)

Рис. 6. Подключение данных.

Выполним настройку адресов:

Use	C...	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		my_master	my_master		[clock]		
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	Double-click to export	clk		
<input checked="" type="checkbox"/>		my_slave	my_slave		[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to export	clk	0x0000_0000	0x0000_0000
<input checked="" type="checkbox"/>		my_slaveWS	my_slaveWS		[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to export	clk	0x0000_0004	0x0000_0007

Рис. 7. Настройка адресов.

В итоге результат выглядит следующим образом:

Use	Connections	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		clk	Clock Source				
<input checked="" type="checkbox"/>		clk_in	Clock Input	clk	exported [clk_in]		
<input checked="" type="checkbox"/>		clk_in_reset	Reset Input	reset	clk		
<input checked="" type="checkbox"/>		clk	Clock Output	Double-click to export	clk		
<input checked="" type="checkbox"/>		clk_reset	Reset Output	Double-click to export	clk		
<input checked="" type="checkbox"/>		my_master	my_master				
<input checked="" type="checkbox"/>		clock	Clock Input	Double-click to export	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	Double-click to export	[clock]		
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	Double-click to export	[clock]		
<input checked="" type="checkbox"/>		my_slave	my_slave				
<input checked="" type="checkbox"/>		clock	Clock Input	Double-click to export	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	Double-click to export	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to export	[clock]	0x0000_0000	0x0000_0000
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	dout_a	[clock]		
<input checked="" type="checkbox"/>		my_slaveWS	my_slaveWS				
<input checked="" type="checkbox"/>		clock	Clock Input	Double-click to export	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	Double-click to export	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to export	[clock]	0x0000_0004	0x0000_0007
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	dout_b	[clock]		

Рис. 8. Результат.

Таже выполним настройку my_master:

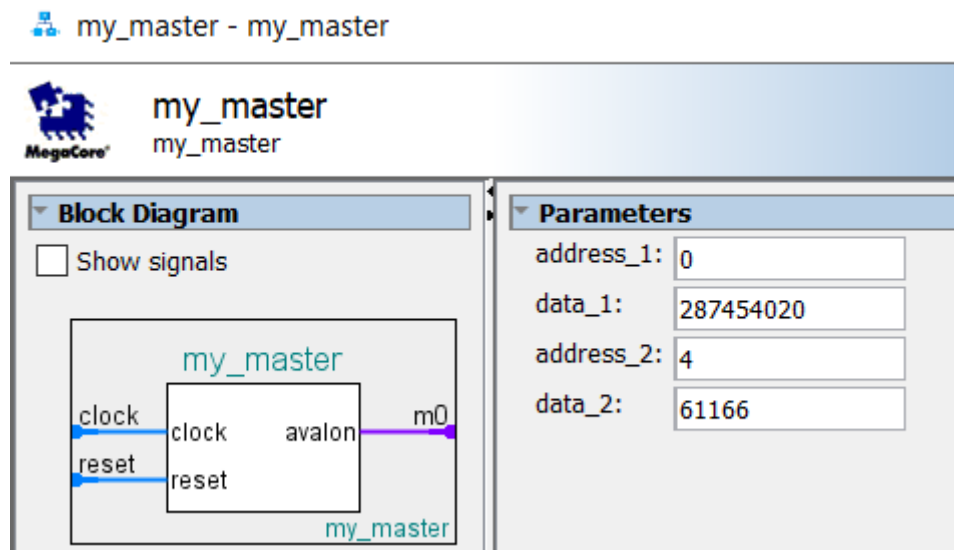


Рис. 9. Настройка my_master.

Посмотрим на получившийся Block Symbol:

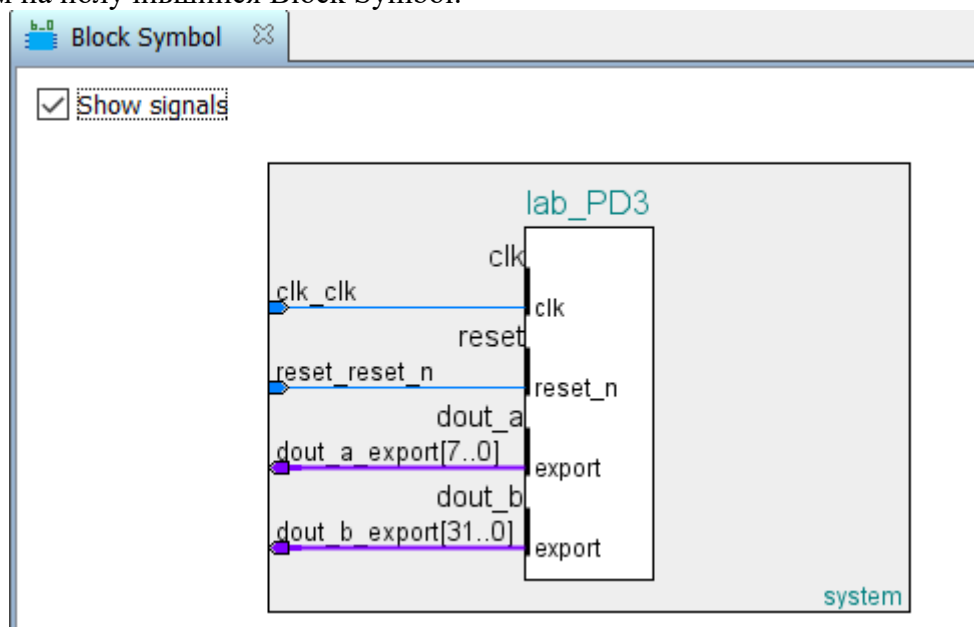


Рис. 10. Block Symbol разработанного IP модуля.

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

System: lab_PD3 Path: clk							
Use	Connections	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		clk	Clock Source				
<input checked="" type="checkbox"/>		clk_in	Clock Input	clk	exported		
<input checked="" type="checkbox"/>		clk_in_reset	Reset Input	reset	[clk_in]		
<input checked="" type="checkbox"/>		clk	Clock Output	clk	clk		
<input checked="" type="checkbox"/>		clk_reset	Reset Output	clk_reset	clk		
<input checked="" type="checkbox"/>		my_master	my_master				
<input checked="" type="checkbox"/>		clock	Clock Input	clock	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	reset	[clock]		
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	m0	[clock]		
<input checked="" type="checkbox"/>		my_slave	my_slave				
<input checked="" type="checkbox"/>		clock	Clock Input	clock	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	reset	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	s0	[clock]		
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	conduit_end_0	[clock]	0x0000_0000	0x0000_0000
<input checked="" type="checkbox"/>		my_slaveWS	my_slaveWS				
<input checked="" type="checkbox"/>		clock	Clock Input	clock	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	reset	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	s0	[clock]	0x0000_0004	0x0000_0007
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	conduit_end_0	[clock]		

Рис. 11. Сигналы clk.

System: lab_PD3 Path: clk							
Use	Connections	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		clk	Clock Source				
<input checked="" type="checkbox"/>		clk_in	Clock Input	clk	exported		
<input checked="" type="checkbox"/>		clk_in_reset	Reset Input	reset	[clk_in]		
<input checked="" type="checkbox"/>		clk	Clock Output	clk	clk		
<input checked="" type="checkbox"/>		clk_reset	Reset Output	clk_reset	clk		
<input checked="" type="checkbox"/>		my_master	my_master				
<input checked="" type="checkbox"/>		clock	Clock Input	clock	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	reset	[clock]		
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	m0	[clock]		
<input checked="" type="checkbox"/>		my_slave	my_slave				
<input checked="" type="checkbox"/>		clock	Clock Input	clock	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	reset	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	s0	[clock]		
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	conduit_end_0	[clock]	0x0000_0000	0x0000_0000
<input checked="" type="checkbox"/>		my_slaveWS	my_slaveWS				
<input checked="" type="checkbox"/>		clock	Clock Input	clock	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	reset	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	s0	[clock]	0x0000_0004	0x0000_0007
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	conduit_end_0	[clock]		

Рис. 12. Сигналы rst.

Выполним команду Show System with Platform Designer Interconnect:

System: lab_PD3							
Use	Connections	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		mm_interconnect...	MM Interconnect				
		clk_clk	Clock Input	Double-click to export	clk		
		my_master_reset...	Reset Input	Double-click to export	[clk_clk]		
		my_master_m0	Avalon Memory Mapped Slave	Double-click to export	[clk_clk]	0x0000_0000	0xffff_ffff
		my_slave_s0	Avalon Memory Mapped Master	Double-click to export	[clk_clk]		
		my_slaveWS_s0	Avalon Memory Mapped Master	Double-click to export	[clk_clk]		
<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"/>		my_master	my_master				
		clock	Clock Input	Double-click to export	clk		
		reset	Reset Input	Double-click to export	[clock]		
		m0	Avalon Memory Mapped Master	Double-click to export	[clock]		
<input checked="" type="checkbox"/>		my_slave	my_slave				
		clock	Clock Input	Double-click to export	clk		
		reset	Reset Input	Double-click to export	[clock]		
		s0	Avalon Memory Mapped Slave	Double-click to export	[clock]	0x0000	0x0000
		conduit_end_0	Conduit	dout_a	[clock]		
<input checked="" type="checkbox"/>		my_slaveWS	my_slaveWS				
		clock	Clock Input	Double-click to export	clk		
		reset	Reset Input	Double-click to export	[clock]		
		s0	Avalon Memory Mapped Slave	Double-click to export	[clock]	0x0000	0x0003
		conduit_end_0	Conduit	dout_b	[clock]		

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

Появившийся модуль обеспечивает передачу данных между мастером (32 бит) и слейвом (8 бит).

Посмотрим на соединения настроенных модулей в Schematic:

System: lab_PD3							
Use	Connections	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		mm_interconnect...	MM Interconnect				
		clk_clk	Clock Input	Double-click to export	clk		
		my_master_reset...	Reset Input	Double-click to export	[clk_clk]		
		my_master_m0	Avalon Memory Mapped Slave	Double-click to export	[clk_clk]	0x0000_0000	0xffff_ffff
		my_slave_s0	Avalon Memory Mapped Master	Double-click to export	[clk_clk]		
		my_slaveWS_s0	Avalon Memory Mapped Master	Double-click to export	[clk_clk]		
<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"/>		my_master	my_master				
		clock	Clock Input	Double-click to export	clk		
		reset	Reset Input	Double-click to export	[clock]		
		m0	Avalon Memory Mapped Master	Double-click to export	[clock]		
<input checked="" type="checkbox"/>		my_slave	my_slave				
		clock	Clock Input	Double-click to export	clk		
		reset	Reset Input	Double-click to export	[clock]		
		s0	Avalon Memory Mapped Slave	Double-click to export	[clock]	0x0000	0x0000
		conduit_end_0	Conduit	dout_a	[clock]		
<input checked="" type="checkbox"/>		my_slaveWS	my_slaveWS				
		clock	Clock Input	Double-click to export	clk		
		reset	Reset Input	Double-click to export	[clock]		
		s0	Avalon Memory Mapped Slave	Double-click to export	[clock]	0x0000	0x0003
		conduit_end_0	Conduit	dout_b	[clock]		

Рис. 14. Schematic.

Добавим созданные модули в проект:

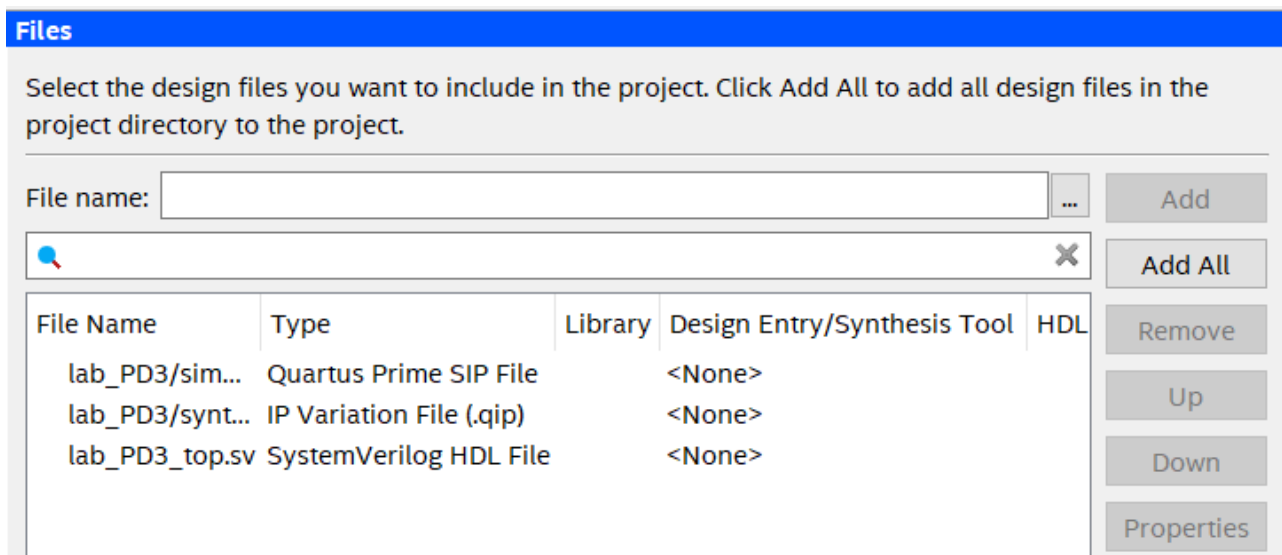


Рис. 15. Добавленные модули к проекту.

Подключим к проекту файл верхнего уровня и выполним компиляцию

```

1  `timescale 1 ns / 1 ns
2  module lab_PD3_top (
3      input bit clk,
4      input bit reset,
5      output bit [7:0] ledA,
6      output bit [31:0] ledB
7  );
8  lab_PD3 lab3_1_inst (
9      .clk_clk (clk),
10     .reset_reset_n (reset),
11     .dout_a_export (ledA),
12     .dout_b_export (ledB)
13 );
14 endmodule

```

Посмотрим на RTL Viewer:

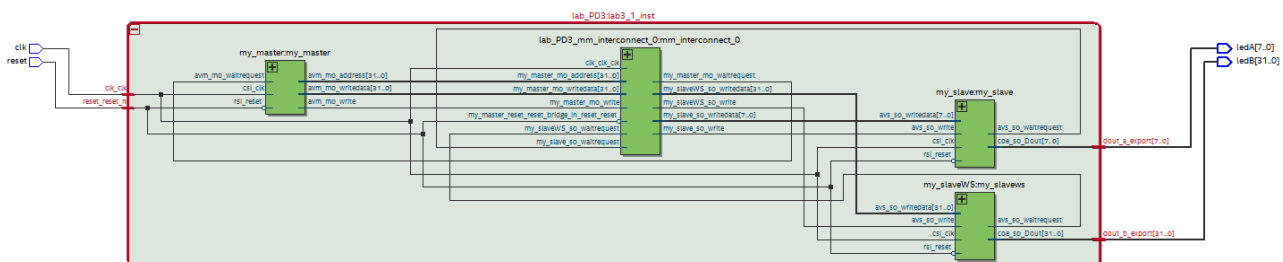


Рис. 16. RTL Viewer устройства.

Протестируем проект, используя следующий тестовый файл:

```

1  `timescale 1 ns / 1 ns
2  module tb_lab_PD3_top ();
3  bit clk;
4  bit reset;
5  bit[7:0] ledA;
6  bit[31:0] ledB;
7
8  always #50 clk = ~ clk;
9
10 initial
11 begin
12     clk      = 1'b0;
13     reset    = 1'b0;
14     #200;
15     reset    = 1'b1;
16     #1000;
17     $stop;
18 end
19
20 lab_PD3_top Lab3_top_inst (.*);
21
22 endmodule

```

Результат запуска приведен ниже:

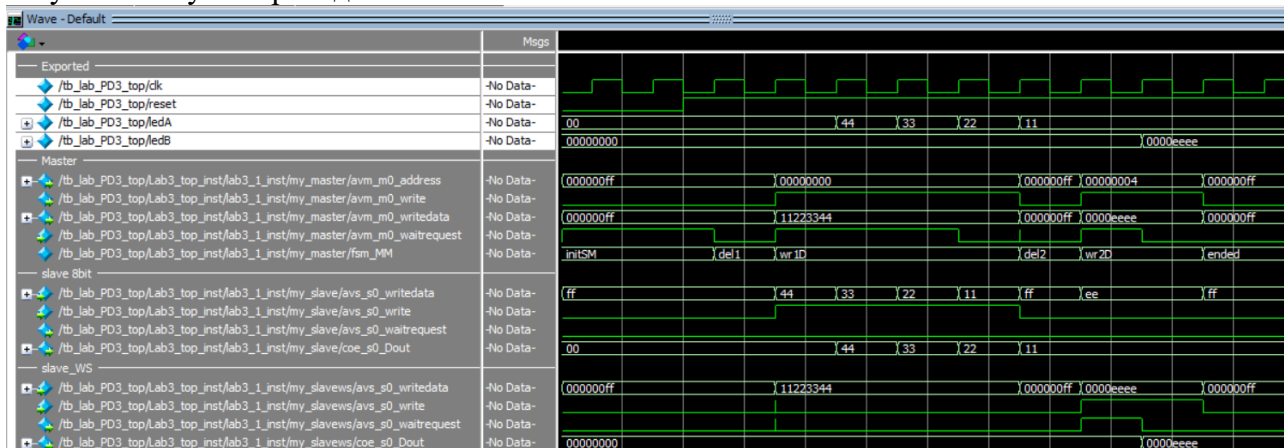


Рис. 17. Результат запуска тестового модуля.

Как мы видим, разработанное устройство соответствует ожиданиям: данные успешно передаются как слейву с 8 битами, так и с 32 битами.

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

```

1  module lab2_PD3_top (
2      (* altera_attribute = "-name IO_STANDARD \"3.3-V LVCMOS\"; chip_pin = \"23\" *)
3      input bit clk
4  );
5
6  bit reset;
7  bit [7:0] ledA;
8  bit [31:0] ledB;
9
10 SP_unit u0 (
11     .source (reset),
12     .source_clk (clk)
13 );
14
15 lab_PD3_top Lab3_top_inst (.*);
16
17 endmodule

```

Получим следующий RTL Viewer:

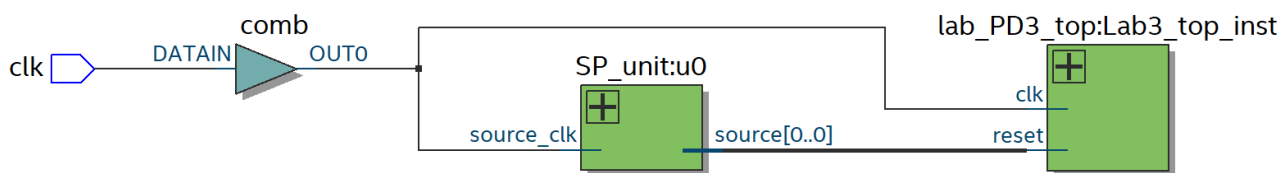


Рис. 18. RTL Viewer.

Запустим Signal Tap II и переключим rst из 0 в 1. Результат запуска приведен ниже:

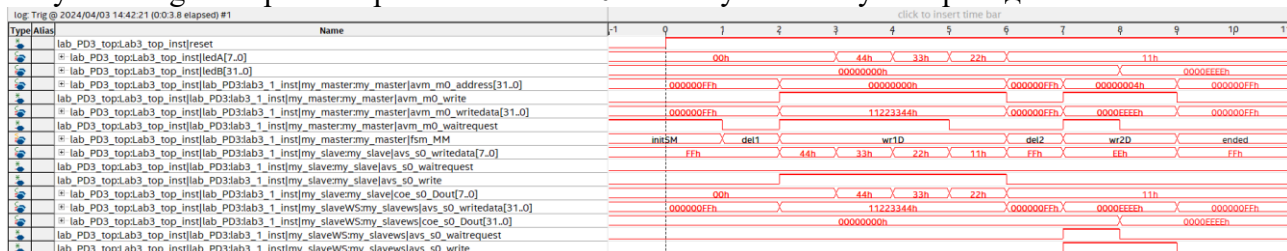


Рис. 19. Signal Tap II.

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

Поменяем адрес и данные второго слейва в соответствии с вариантом:

my_master - my_master

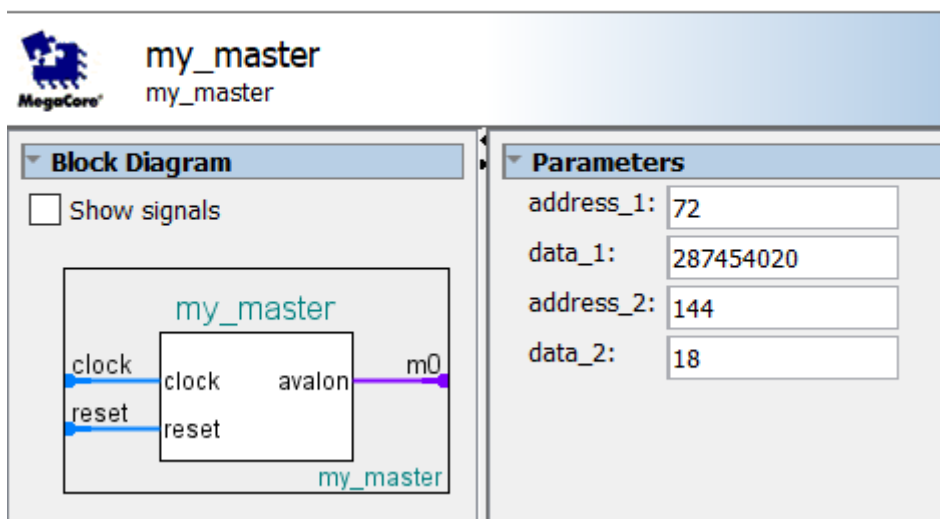


Рис. 20. Обновленная настройка my master.

Use	Connections	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		clk	Clock Source				
<input checked="" type="checkbox"/>		clk_in	Clock Input	clk	exported		
<input checked="" type="checkbox"/>		clk_in_reset	Reset Input	reset	[clk_in]		
<input checked="" type="checkbox"/>		clk	Clock Output	clk	clk		
<input checked="" type="checkbox"/>		clk_reset	Reset Output	reset	clk		
<input checked="" type="checkbox"/>		my_master	my_master				
<input checked="" type="checkbox"/>		clock	Clock Input	clk	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	reset	[clock]		
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master		[clock]		
<input checked="" type="checkbox"/>		my_slave	my_slave				
<input checked="" type="checkbox"/>		clock	Clock Input	clk	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	reset	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave		[clock]	0x0000_0048	0x0000_0048
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	dout_a	[clock]		
<input checked="" type="checkbox"/>		my_slaveWS	my_slaveWS				
<input checked="" type="checkbox"/>		clock	Clock Input	clk	clk		
<input checked="" type="checkbox"/>		reset	Reset Input	reset	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave		[clock]	0x0000_0090	0x0000_0093
<input checked="" type="checkbox"/>		conduit_end_0	Conduit	dout_b	[clock]		

Рис. 21. Обновленная настройка адресов в Qsys.

Запустим тестовый файл:

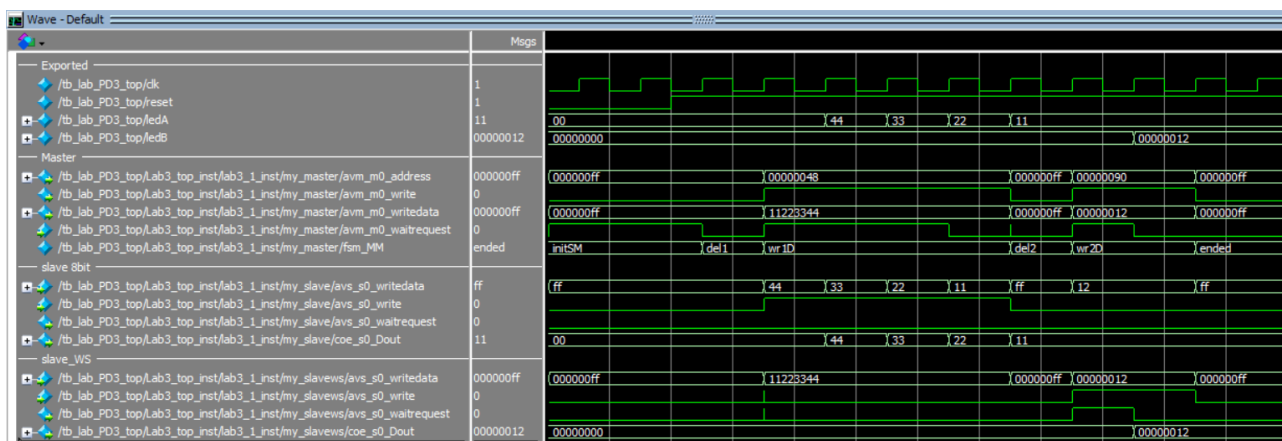


Рис. 22. Запуск тестового файла.

Как мы видим, данные действительно обновились в соответствии с вариантом.

Проверим результат в Signal Tap II:

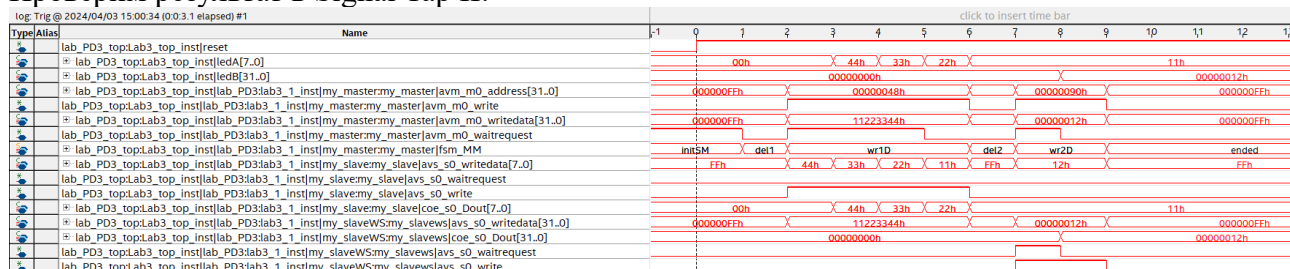


Рис. 23. Результат запуска в Signal Tap II.

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.