

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

### **Отчёт по лабораторной работе Lab\_MS\_SV\_3**

Дисциплина: Автоматизация проектирования дискретных устройств (на  
английском языке)

Выполнил студент гр. 5130901/10101 \_\_\_\_\_ М.Т. Непомнящий  
(подпись)

Руководитель \_\_\_\_\_ А.А. Федотов  
(подпись)

Санкт-Петербург  
2024

## Оглавление

1.	Задание.....	4
2.	Ход работы.....	4
2.1.	Создание проекта.....	4
	Подготовка проекта.....	4
	Начало работы в PD .....	5
2.2.	Подключение тактового сигнала .....	7
2.3.	Подключение Avalon-MM интерфейсов .....	8
2.4.	Настройка компонентов.....	9
	Настройка my_masterA .....	9
	Настройка my_masterB.....	10
	Настройка st_splitter .....	10
	Настройка st_delay.....	11
2.5.	Подключение Avalon-ST интерфейсов.....	11
2.6.	Анализ системы .....	12
	Проверка блока .....	12
	Отладка модулей с проблемными подключениями .....	13
	Анализ с помощью Schematic .....	14
2.7.	Генерация системы.....	15
2.8.	Конец работы .....	16
3.	Вывод.....	17

## Список иллюстраций

Рис. 1 – Структура проекта.....	4
Рис. 2 – Детали проекта .....	4
Рис. 3 – Задания пути к библиотеке IP .....	5
Рис. 4 – Исходное окно PD .....	5
Рис. 5 – Добавление компонентов .....	6
Рис. 6 – Изменение имён компонентов .....	6
Рис. 7 – Подключение тактового сигнала (1).....	7
Рис. 8 – Подключение тактового сигнала (2).....	7
Рис. 9 – Подключение Avalon-MM интерфейсов .....	8
Рис. 10 – Редактирование адресов .....	8
Рис. 11 – Фиксация адресов.....	9
Рис. 12 – Проверка корректности адресов .....	9
Рис. 13 – Настройка компонента my_masterA .....	9
Рис. 14 – Настройка компонента my_masterB .....	10
Рис. 15 – Настройка компонента st_splitter.....	10
Рис. 16 – Настройка компонента st_delay .....	11
Рис. 17 – Установка подключений компонентов .....	11
Рис. 18 – Экспорт выводов .....	11
Рис. 19 – Проверка поля Messages на отсутствие ошибок .....	12
Рис. 20 – Символ системы .....	12
Рис. 21 – Show System with QSYS Interconnect.....	12
Рис. 22 – Анализ проблемных подключений.....	13
Рис. 23 – Исправленный модуль clk .....	13
Рис. 24 – Повторное выполнение Show System with QSYS Interconnect.....	14
Рис. 25 – Проверка отсутствия ошибок в системе .....	14
Рис. 26 – Schematic (фильтр по in) .....	14
Рис. 27 – Schematic (фильтр по clk) .....	15
Рис. 28 – Предустановки окна Generation.....	15
Рис. 29 – Проверка успешности генерации HDL .....	16
Рис. 30 – Сообщение с указанием пакетов, которые можно подключить .....	16
Рис. 31 – Анализ рабочей папки проекта .....	16

# 1. Задание

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

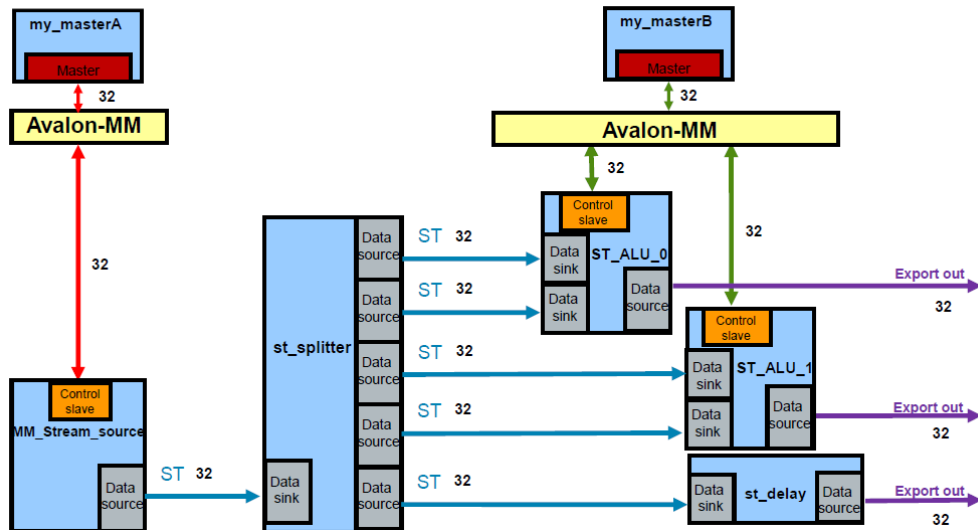


Рис. 1 – Структура проекта

## 2. Ход работы

### 2.1. Создание проекта

#### Подготовка проекта

Создадим проект, установив следующие значения:

Summary	
When you click Finish, the project will be created with the following settings:	
Project directory:	D:\Users\Legion\Quartus and Verilog\Verilog\Third year\6th semester\lectures\lecture 4 - PD\Q_PD\lab_PD1
Project name:	lab_PD1
Top-level design entity:	lab_PD1
Number of files added:	0
Number of user libraries added:	0
Device assignments:	
Design template:	n/a
Family name:	Cyclone IV E
Device:	EP4CE6E22C8
Board:	n/a
EDA tools:	
Design entry/synthesis:	<None> (<None>)
Simulation:	ModelSim-Altera (SystemVerilog HDL)
Timing analysis:	()
Operating conditions:	
VCCINT voltage:	1.2V
Junction temperature range:	0-85 °C

Рис. 2 – Детали проекта

Перейдём по пути Tools → Options → IP Settings → IP Catalog Search Locations и зададим путь к библиотеке IP:

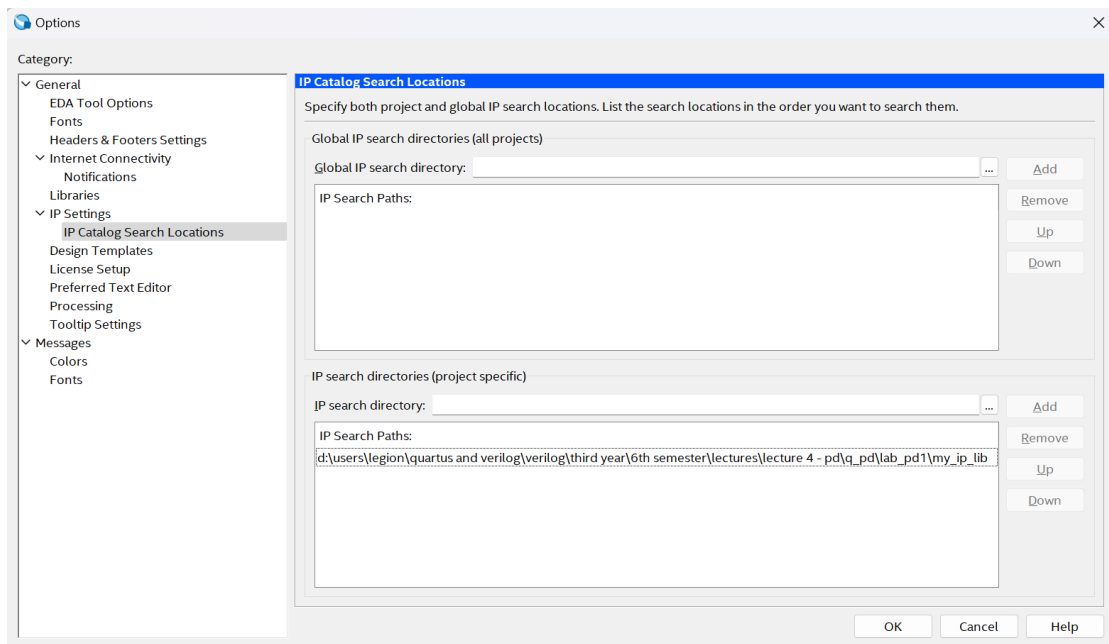


Рис. 3 – Задания пути к библиотеке IP

## Начало работы в PD

Откроем PD и сохраним систему:

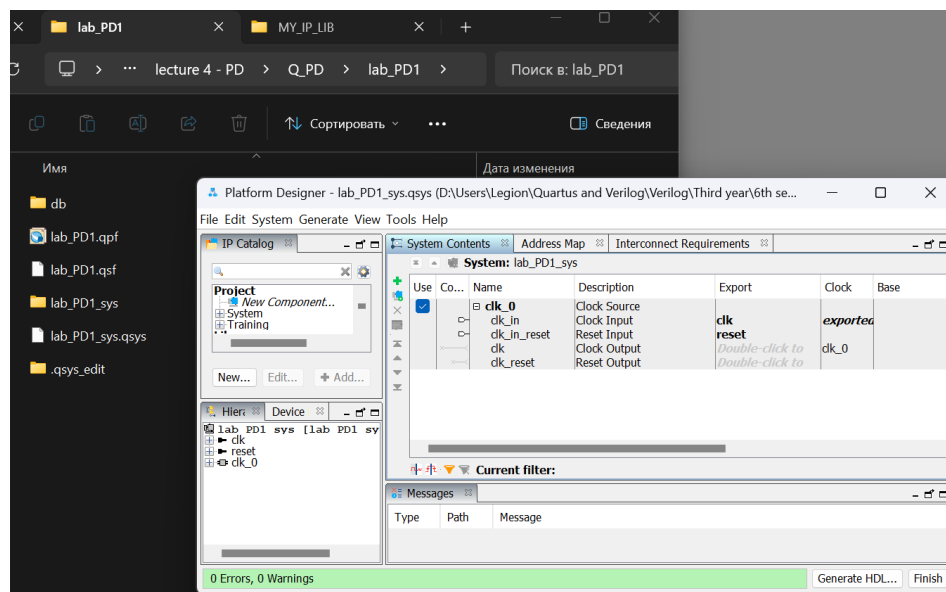


Рис. 4 – Исходное окно PD

Добавим компоненты: `my_masterA_component`, `MM_stream_source_component`, `my_masterB_component`, Avalon-ST Splitter, Avalon-ST Delay и `ST_ALU_component` (2 компонента). Таким образом, получим следующую картинку (в окне Hierarchy слева отображаются все добавленные компоненты):

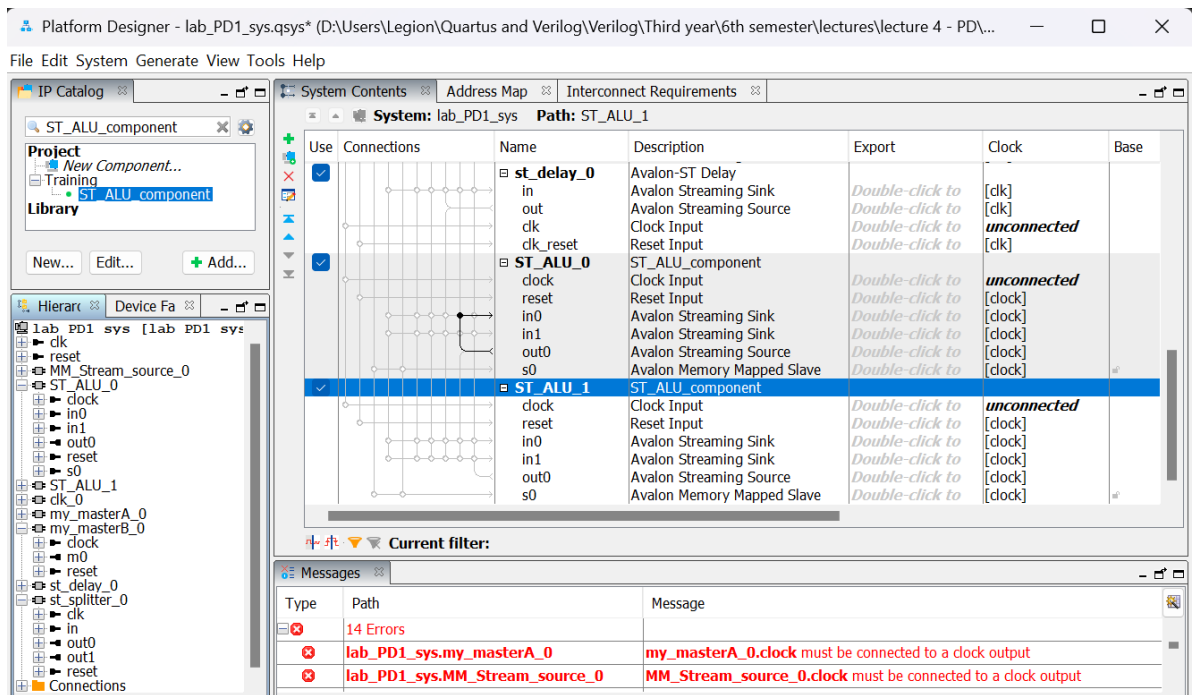


Рис. 5 – Добавление компонентов

Наличие ошибок связано с тем, что настройка модулей не производилась, т. к. она будет рассмотрена дальше.

Переименуем некоторые из компонентов и сохраним систему:

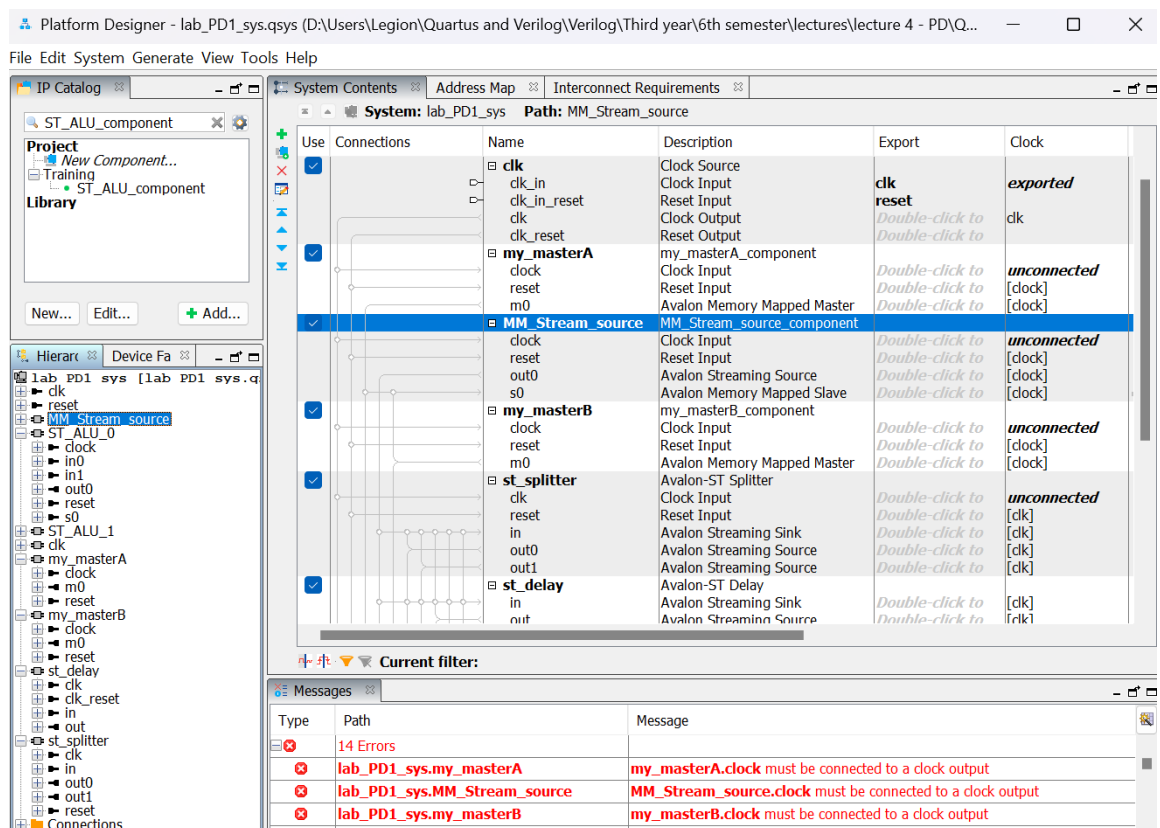


Рис. 6 – Изменение имён компонентов

## 2.2. Подключение тактового сигнала

Выделим интерфейс clk компонента clk, и, открыв его соединения, выберем подключение ко всем тактовым входам:

System: lab_PD1_sys Path: clk.clk				
Connected to: clk.clk				
Connected	Connection	Clock Crossing	Data Width	Burst
<input checked="" type="checkbox"/>	clk.clk/MM_Stream_source.c...			
<input checked="" type="checkbox"/>	clk.clk/ST_ALU_0.clock			
<input checked="" type="checkbox"/>	clk.clk/ST_ALU_1.clock			
<input checked="" type="checkbox"/>	clk.clk/my_masterA.clock			
<input checked="" type="checkbox"/>	clk.clk/my_masterB.clock			
<input checked="" type="checkbox"/>	clk.clk/st_delay.clk			
<input checked="" type="checkbox"/>	clk.clk/st_splitter.clk			

Рис. 7 – Подключение тактового сигнала (1)

Подключим тактовый сигнал, выполнив Filter → Clock and Reset Interfaces, убедимся, что соединения выполнены. Также, подключим сигнал Reset, выполнив System → Create Global Reset Network, и убедимся, что соединения для reset также выполнены:

Use	Co...	Name	Description	Export	Clock	Base
<input checked="" type="checkbox"/>		clk	Clock Source	clk	exported	
<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	clk		
<input checked="" type="checkbox"/>		clk_reset	Reset Output	reset		
<input checked="" type="checkbox"/>		my_masterA	my_masterA_component			
<input checked="" type="checkbox"/>		clk	Clock Input	clk [clock]		
<input checked="" type="checkbox"/>		reset	Reset Input	reset		
<input checked="" type="checkbox"/>		MM_Stream_source	MM_Stream_source_component			
<input checked="" type="checkbox"/>		clk	Clock Input	clk [clock]		
<input checked="" type="checkbox"/>		reset	Reset Input	reset		
<input checked="" type="checkbox"/>		my_masterB	my_masterB_component			
<input checked="" type="checkbox"/>		clk	Clock Input	clk [clock]		
<input checked="" type="checkbox"/>		reset	Reset Input	reset		
<input checked="" type="checkbox"/>		st_splitter	Avalon-ST Splitter			
<input checked="" type="checkbox"/>		clk	Clock Input	clk [clk]		
<input checked="" type="checkbox"/>		reset	Reset Input	reset		
<input checked="" type="checkbox"/>		st_delay	Avalon-ST Delay			
<input checked="" type="checkbox"/>		clk	Clock Input	clk [clk]		
<input checked="" type="checkbox"/>		clk_reset	Reset Input	reset		
<input checked="" type="checkbox"/>		ST_ALU_0	ST_ALU_component			
<input checked="" type="checkbox"/>		clk	Clock Input	clk [clock]		
<input checked="" type="checkbox"/>		reset	Reset Input	reset		
<input checked="" type="checkbox"/>		ST_ALU_1	ST_ALU_component			
<input checked="" type="checkbox"/>		clk	Clock Input	clk [clock]		
<input checked="" type="checkbox"/>		reset	Reset Input	reset		

Current filter: Clock and Reset Interfaces

Type	Path	Message
Warning	lab_PD1_sys.MM_Stream_source.out0	MM_Stream_source.out0 must be connected to an Avalon-ST sink
Warning	lab_PD1_sys.st_splitter.in	st_splitter.in must be connected to an Avalon-ST source
Warning	lab_PD1_sys.st_splitter.out0	st_splitter.out0 must be connected to an Avalon-ST sink
Warning	lab_PD1_sys.st_splitter.out1	st_splitter.out1 must be connected to an Avalon-ST sink

0 Errors, 15 Warnings

Рис. 8 – Подключение тактового сигнала (2)

## 2.3. Подключение Avalon-MM интерфейсов

Выполним Filter → Avalon-MM Interfaces и выберем соединения так, как показано на картинке ниже

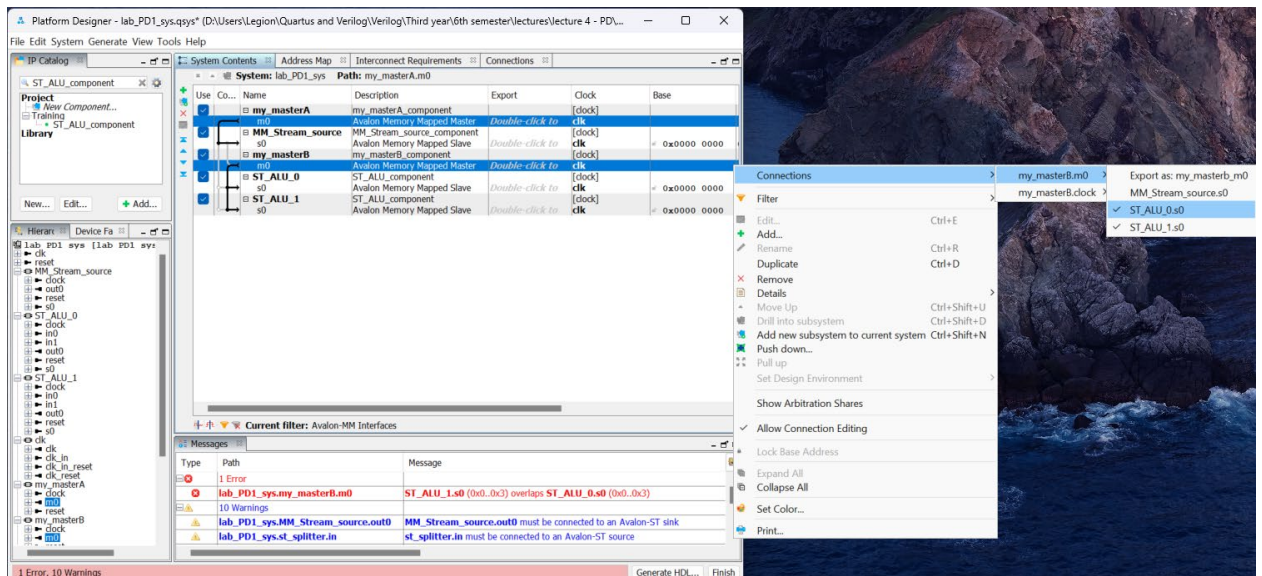



Рис. 9 – Подключение Avalon-MM интерфейсов

Заметим, что адреса не совпадают (сверху так, как должно быть, снизу то, что получилось):

Use	Co...	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		my_masterA	my_masterA_component		[clock]		
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	Double-click to	clk_0		
<input checked="" type="checkbox"/>		MM_Stream_source	MM_Stream_source_com...	Double-click to	clk_0	0x0000_0000	0x0000_0003
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to	clk_0		
<input checked="" type="checkbox"/>		my_masterB	my_masterB_component		[clock]		
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	Double-click to	clk_0		
<input checked="" type="checkbox"/>		ST_ALU_0	ST_ALU_component	Double-click to	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to	clk_0	0x0000_0004	0x0000_0007
<input checked="" type="checkbox"/>		ST_ALU_1	ST_ALU_component	Double-click to	[clock]		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave	Double-click to	clk_0	0x0000_0000	0x0000_0003

- Если адреса назначены иначе – отредактируйте их
  - Дважды щелкните в поле Base адреса и введите правильный адрес (как на картинке)
- Зафиксируйте адреса – нажмите на символ  у каждого адреса

Platform Designer - lab\_PD1\_sys.qsys\* (D:\Users\Legion\Quartus and Verilog\Verilog\Third year\6th semester\lectures\lecture 4 - PD\Q\_PD\lab\_PD1\lab\_PD1\_sys.qsys)

le Edit System Generate View Tools Help

IP Catalog

Project

Library

System Contents

Address Map

Interconnect Requirements

Connections

System: lab\_PD1\_sys Path: my\_masterA.m0

Use Co... Name Description Export Clock Base End I... Tags

my\_masterA my\_masterA\_component [clock]

m0 Avalon Memory Mapped Master Double-click to clk\_0

MM\_Stream\_source MM\_Stream\_source\_component Double-click to clk\_0

s0 Avalon Memory Mapped Slave Double-click to clk\_0

my\_masterB my\_masterB\_component [clock]

m0 Avalon Memory Mapped Master Double-click to clk\_0

ST\_ALU\_0 ST\_ALU\_component Double-click to [clock]

s0 Avalon Memory Mapped Slave Double-click to clk\_0

ST\_ALU\_1 ST\_ALU\_component Double-click to [clock]

s0 Avalon Memory Mapped Slave Double-click to clk\_0

Рис. 10 – Редактирование адресов

Поменяем адреса, проверим, что они верны, и зафиксируем их (нажать символ замка у каждого из адресов):



Use	Co...	Name	Description	Export	Clock	Base	End	I...	Tags
<input checked="" type="checkbox"/>		<b>my_masterA</b>	my_masterA_component		[dock]				
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	Double-click to	clk				
<input checked="" type="checkbox"/>		MM_Stream_source	MM_Stream_source_component	Double-click to	clk				
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave			0x0000_0000	0x0000_0003		
<input checked="" type="checkbox"/>		my_masterB	my_masterB_component		[dock]				
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped Master	Double-click to	clk				
<input checked="" type="checkbox"/>		ST_ALU_0	ST_ALU_component	Double-click to	clk				
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave			0x0000_0004	0x0000_0007		
<input checked="" type="checkbox"/>		ST_ALU_1	ST_ALU_component	Double-click to	clk				
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped Slave			0x0000_0000	0x0000_0003		

Рис. 11 – Фиксация адресов

Откроем окно закладок View → Address Map. В окне появятся Ведущие (master) и ведомые (slave) шины, т. е. столбцы и строки Avalon MM соответственно. Проверим, что они выведены корректно:

System: lab_PD1_sys	Path: my_masterA.m0	
	my_masterA.m0	my_masterB.m0
MM_Stream_source.s0	0x0000_0000 - 0x0000_0003	
ST_ALU_0.s0		0x0000_0004 - 0x0000_0007
ST_ALU_1.s0		0x0000_0000 - 0x0000_0003

Рис. 12 – Проверка корректности адресов

Запомним базовые адреса:

- MM\_Stream\_source = 0,
- ST\_ALU\_0 = 4,
- ST\_ALU\_1 = 0.

## 2.4. Настройка компонентов

### Настройка my\_masterA

Зададим адрес, по которому Ведущий **my\_masterA** будет записывать данные (он = 0, т. к. это базовый адрес Ведомого MM\_Stream\_source). Остальное зададим так, как показано на рисунке ниже:

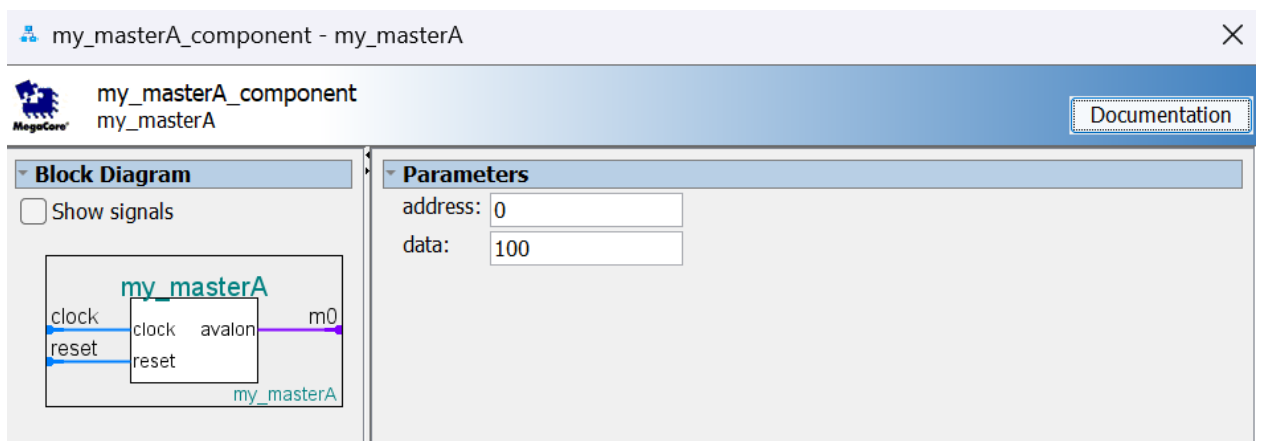


Рис. 13 – Настройка компонента my\_masterA

Запись данных будет происходить следующим образом: 100 – счёт на сложение, 200 – счёт на вычитание.

## Настройка my\_masterB

Адреса, по которому Ведущий my\_masterB будет записывать данные:

- 0 (это базовый адрес Ведомого ST\_ALU\_1)
- 4 (это базовый адрес Ведомого ST\_ALU\_0)

Записываемые данные:

- 111—сложение (тип операции ST\_ALU\_1)
- 222—умножение (тип операции ST\_ALU\_0)

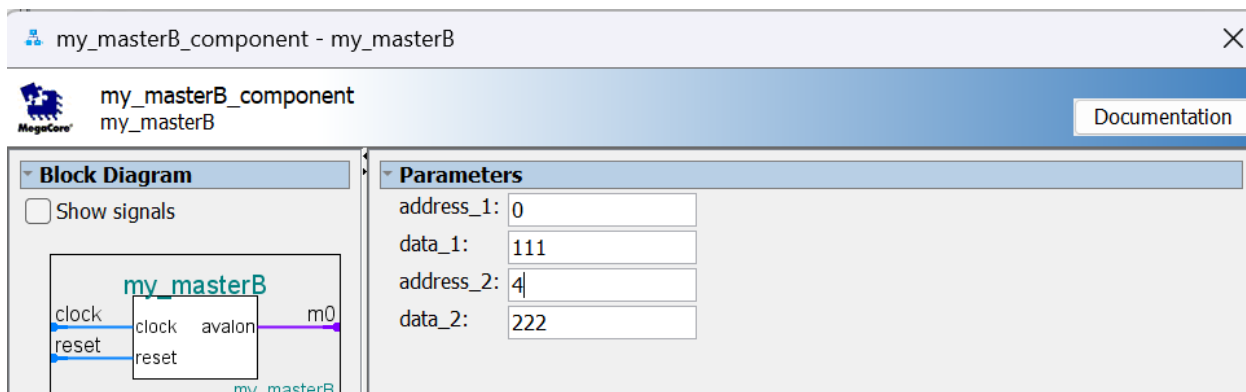


Рис. 14 – Настройка компонента my\_masterB

## Настройка st\_splitter

Установим значения:

- NUMBER\_OF\_OUTPUTS = 5
- DATA\_WIDTH = 32

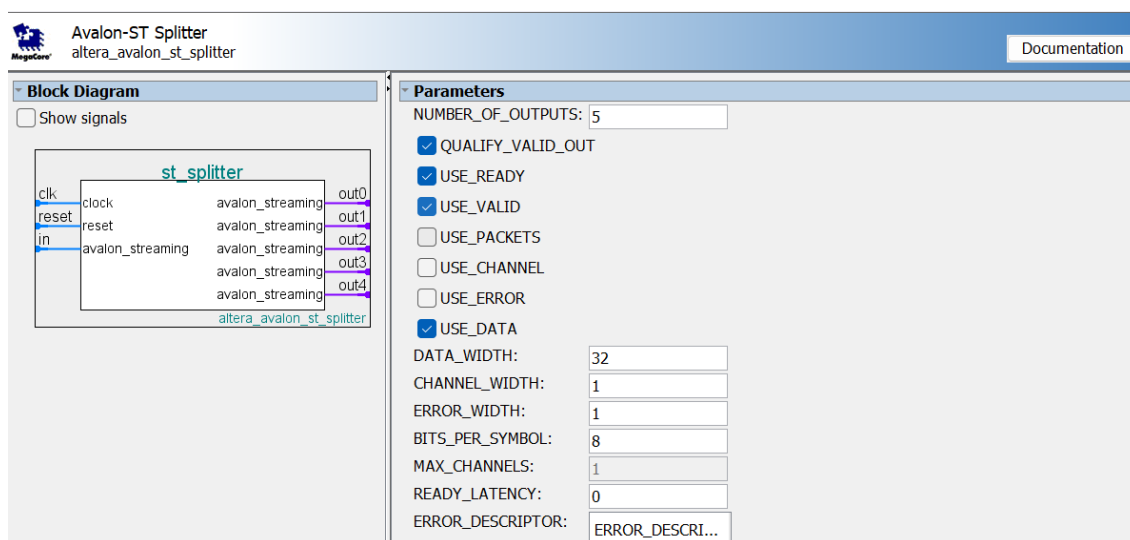


Рис. 15 – Настройка компонента st\_splitter

## Настройка st\_delay

Установим значение DATA\_WIDTH = 32

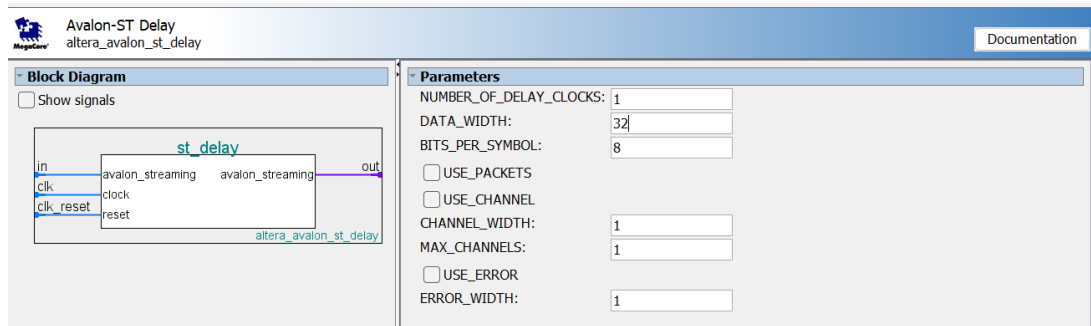


Рис. 16 – Настройка компонента st\_delay

## 2.5. Подключение Avalon-ST интерфейсов

Установим подключения в столбце Connections так, как показано на рисунке ниже:

Use	Connections	Name	Description	Export	Clock	Base	End	I...	Ta
<input checked="" type="checkbox"/>		MM_Stream_source	MM_Stream_source_component	Double-click to	[clock]	0x0000_0000	0x0000_0003		
<input checked="" type="checkbox"/>		out0	Avalon Streaming Source	Double-click to	clk				
<input checked="" type="checkbox"/>		st_splitter	Avalon-ST Splitter	Double-click to	[clk]				
<input checked="" type="checkbox"/>		in	Avalon Streaming Sink	Double-click to	clk				
<input checked="" type="checkbox"/>		out0	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		out1	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		out2	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		out3	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		out4	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		st_delay	Avalon-ST Delay	Double-click to	[clk]				
<input checked="" type="checkbox"/>		in	Avalon Streaming Sink	Double-click to	clk				
<input checked="" type="checkbox"/>		out	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		ST_ALU_0	ST_ALU_component	Double-click to	[clock]	0x0000_0004	0x0000_0007		
<input checked="" type="checkbox"/>		in0	Avalon Streaming Sink	Double-click to	clk				
<input checked="" type="checkbox"/>		in1	Avalon Streaming Sink	Double-click to	[clock]				
<input checked="" type="checkbox"/>		out0	Avalon Streaming Source	Double-click to	[clock]				
<input checked="" type="checkbox"/>		ST_ALU_1	ST_ALU_component	Double-click to	[clock]	0x0000_0000	0x0000_0003		
<input checked="" type="checkbox"/>		in0	Avalon Streaming Sink	Double-click to	clk				
<input checked="" type="checkbox"/>		in1	Avalon Streaming Sink	Double-click to	[clock]				
<input checked="" type="checkbox"/>		out0	Avalon Streaming Source	Double-click to	[clock]				

Рис. 17 – Установка подключений компонентов

Проведём экспорт выводов путём задания имён для выделенных модулей в столбце Export:

Use	Connections	Name	Description	Export	Clock	Base	End	I...	Ta
<input checked="" type="checkbox"/>		MM_Stream_source	MM_Stream_source_component	Double-click to	[clock]	0x0000_0000	0x0000_0003		
<input checked="" type="checkbox"/>		out0	Avalon Streaming Source	Double-click to	clk				
<input checked="" type="checkbox"/>		st_splitter	Avalon-ST Splitter	Double-click to	[clk]				
<input checked="" type="checkbox"/>		in	Avalon Streaming Sink	Double-click to	clk				
<input checked="" type="checkbox"/>		out0	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		out1	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		out2	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		out3	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		out4	Avalon Streaming Source	Double-click to	[clk]				
<input checked="" type="checkbox"/>		st_delay	Avalon-ST Delay	Double-click to	[clk]				
<input checked="" type="checkbox"/>		in	Avalon Streaming Sink	Double-click to	clk				
<input checked="" type="checkbox"/>		out	Avalon Streaming Source	delay_out	[clk]				
<input checked="" type="checkbox"/>		ST_ALU_0	ST_ALU_component	Double-click to	[clock]	0x0000_0004	0x0000_0007		
<input checked="" type="checkbox"/>		in0	Avalon Streaming Sink	Double-click to	clk				
<input checked="" type="checkbox"/>		in1	Avalon Streaming Sink	Double-click to	[clock]				
<input checked="" type="checkbox"/>		out0	Avalon Streaming Source	alu0_out	[clock]				
<input checked="" type="checkbox"/>		ST_ALU_1	ST_ALU_component	Double-click to	[clock]	0x0000_0000	0x0000_0003		
<input checked="" type="checkbox"/>		in0	Avalon Streaming Sink	Double-click to	clk				
<input checked="" type="checkbox"/>		in1	Avalon Streaming Sink	Double-click to	[clock]				
<input checked="" type="checkbox"/>		out0	Avalon Streaming Source	alu1_out	[clock]				

Рис. 18 – Экспорт выводов

Убедимся в том, что система не содержит ошибок и в поле Messages есть только 1 информационное сообщение:

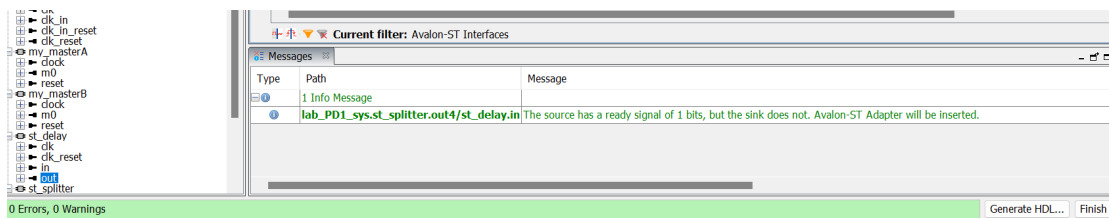


Рис. 19 – Проверка поля Messages на отсутствие ошибок

## 2.6. Анализ системы

### Проверка блока

Выполним View → Block Symbol и убедимся в том, что символ системы построен правильно:

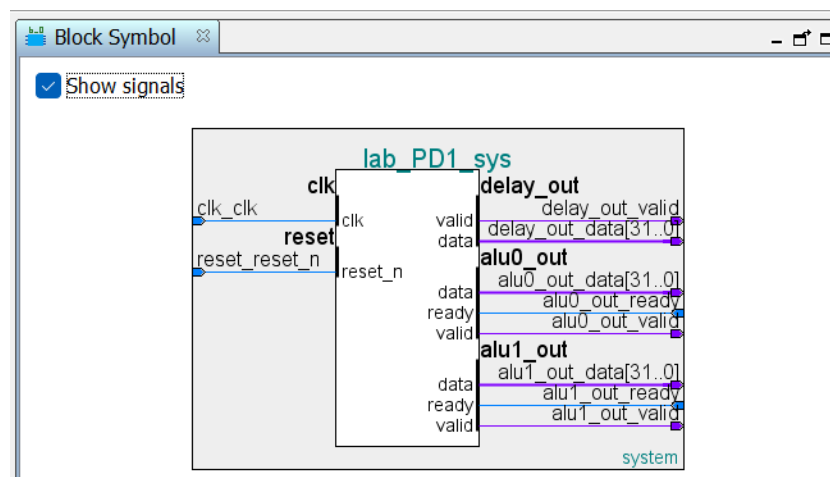


Рис. 20 – Символ системы

Выполним команду System → Show System with PD Interconnect (Show System with QSYS Interconnect). Заметим, что добавились новые модули mm\_interconnect\_0, mm\_interconnect\_1, avalon\_st\_adapter и rst\_controller (см. рисунок ниже). Обратим внимание на наличие модуля rst\_controller, который является адаптером сигнала Reset.

System: lab_PD1_sys										
Use	Connections	Name	Description	Export	Clock	Base	End	I...	Tags	Opcode Name
<input checked="" type="checkbox"/>		mm_interconnect_0	MM Interconnect	Double-click to	clk					
<input checked="" type="checkbox"/>		clk_clk	Clock Input	Double-click to	[clk_clk]					
<input checked="" type="checkbox"/>		my_masterB_reset_re...	Reset Input	Double-click to	[clk_clk]					
<input checked="" type="checkbox"/>		my_masterB_m0	Avalon Memory Mapped Slave	Double-click to	[clk_clk]					
<input checked="" type="checkbox"/>		ST_ALU_0_s0	Avalon Memory Mapped Master	Double-click to	[clk_clk]					
<input checked="" type="checkbox"/>		ST_ALU_1_s0	Avalon Memory Mapped Master	Double-click to	[clk_clk]					
<input checked="" type="checkbox"/>		mm_interconnect_1	MM Interconnect	Double-click to	clk					
<input checked="" type="checkbox"/>		clk_clk	Clock Input	Double-click to	[clk_clk]					
<input checked="" type="checkbox"/>		my_masterA_reset_r...	Reset Input	Double-click to	[clk_clk]					
<input checked="" type="checkbox"/>		my_masterA_m0	Avalon Memory Mapped Slave	Double-click to	[clk_clk]					
<input checked="" type="checkbox"/>		MM_Stream_source_s0	Avalon Memory Mapped Master	Double-click to	[clk_clk]					
<input checked="" type="checkbox"/>		avalon_st_adapter	Avalon-ST Adapter	Double-click to	clk					
<input checked="" type="checkbox"/>		in_clk_0	Clock Input	Double-click to	[in_clk_0]					
<input checked="" type="checkbox"/>		in_rst_0	Reset Input	Double-click to	[in_clk_0]					
<input checked="" type="checkbox"/>		in_0	Avalon Streaming Sink	Double-click to	[in_clk_0]					
<input checked="" type="checkbox"/>		out_0	Avalon Streaming Source	Double-click to						
<input checked="" type="checkbox"/>		rst_controller	Merlin Reset Controller	Double-click to						
<input checked="" type="checkbox"/>		reset_in0	Reset Input	Double-click to	clk					
<input checked="" type="checkbox"/>		clk	Clock Input	Double-click to						
<input checked="" type="checkbox"/>		reset_out	Reset Output	Double-click to						

Рис. 21 – Show System with QSYS Interconnect

Выполним View → Clock domains Beta, выберем режим отображения Reset. Проблемные подключения будут отображаться в виде красных точек, при наведении курсора на них, можно увидеть сообщение с информацией о них:

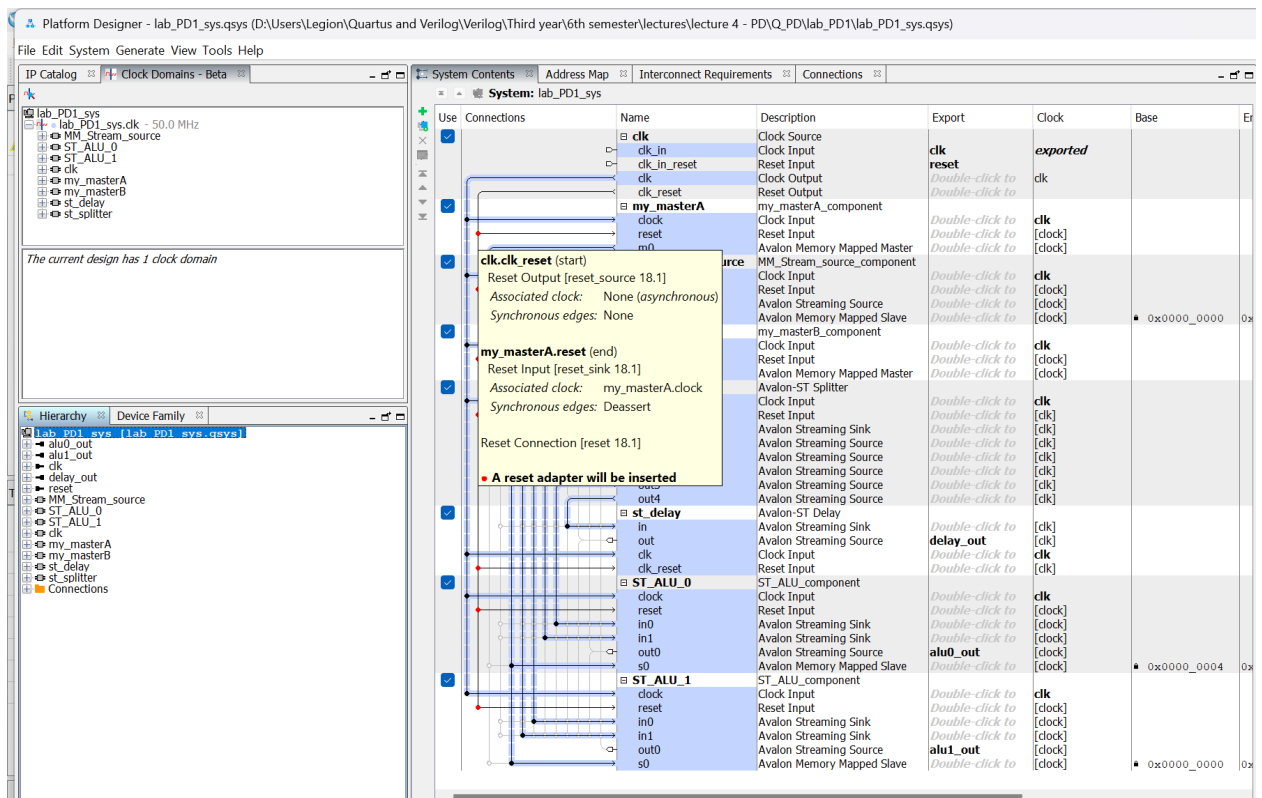


Рис. 22 – Анализ проблемных подключений

## Отладка модулей с проблемными подключениями

Внесём исправления в модуль clk:

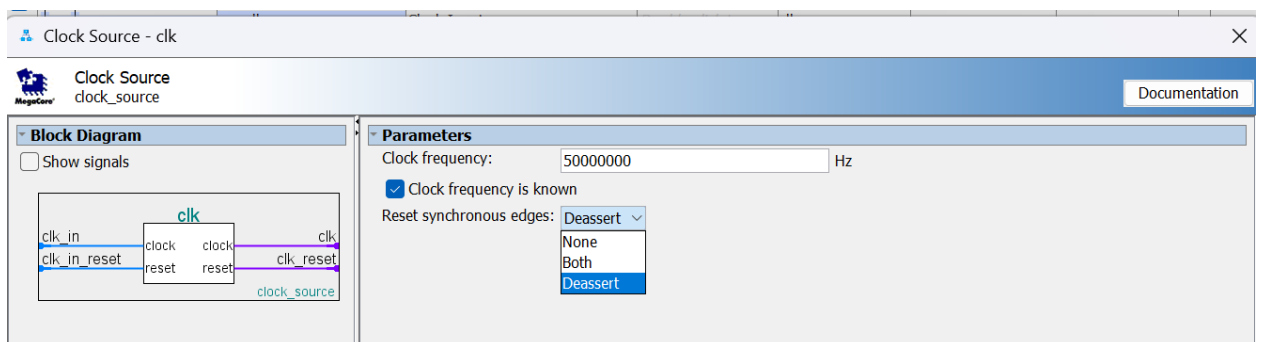


Рис. 23 – Исправленный модуль clk

Заместим, что красные точки исчезли. Теперь выполним повторно System → Show System with PD Interconnect. Теперь адаптер Reset, из-за которого возникали проблемы, должен отсутствовать в системе. Убедимся в этом:

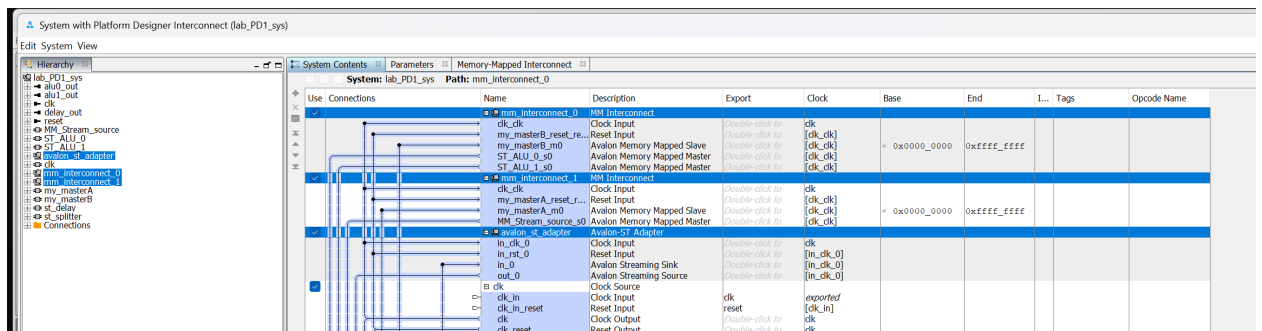


Рис. 24 – Повторное выполнение Show System with QSYS Interconnect

Назначение добавленных модулей

- mm\_interconnect\_0 система межсоединений для Ведущего my\_masterA
- mm\_interconnect\_1 система межсоединений для Ведущего my\_masterB
- avalon\_st\_adapter – адаптер между модулем st\_splitter\_0 и st\_delay\_0
  - Этот адаптер потребовался в системе т. к. у модуля нет выхода Ready, а у модуля st\_splitter\_0 есть – на закладке сообщений есть сообщение с информацией

Убедимся, что в системе нету ошибок:

Type	Path	Message
1 Info Message	lab_PD1_sys.st_splitter.out4/st_delay.in	The source has a ready signal of 1 bits, but the sink does not. Avalon-ST Adapter will be inserted.

Рис. 25 – Проверка отсутствия ошибок в системе

Анализ с помощью Schematic

Выполним View → Schematic, в качестве фильтра введём in и убедимся в том, что система синхронизации и каналы ST системы подключены верно:

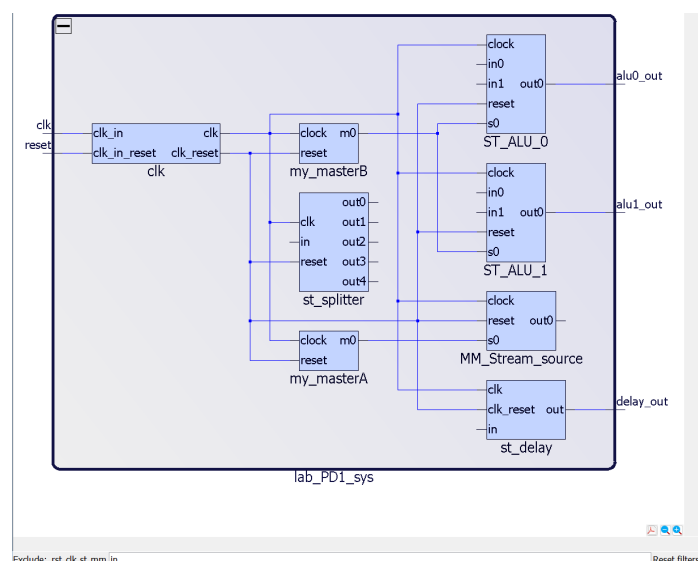


Рис. 26 – Schematic (фильтр по in)

Теперь введём в качестве фильтра clk, чтобы проверить, что шины Avalon MM подключены верно:

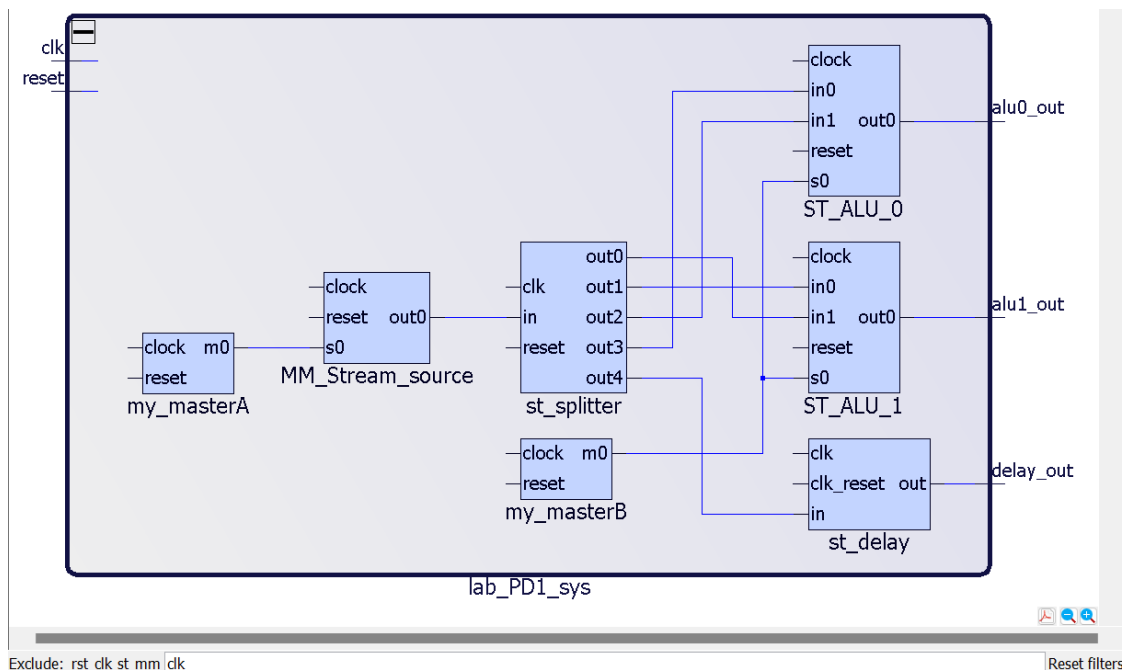


Рис. 27 – Schematic (фильтр по clk)

## 2.7. Генерация системы

Выполним PD → Generate HDL и укажем следующие предустановки для генерации:

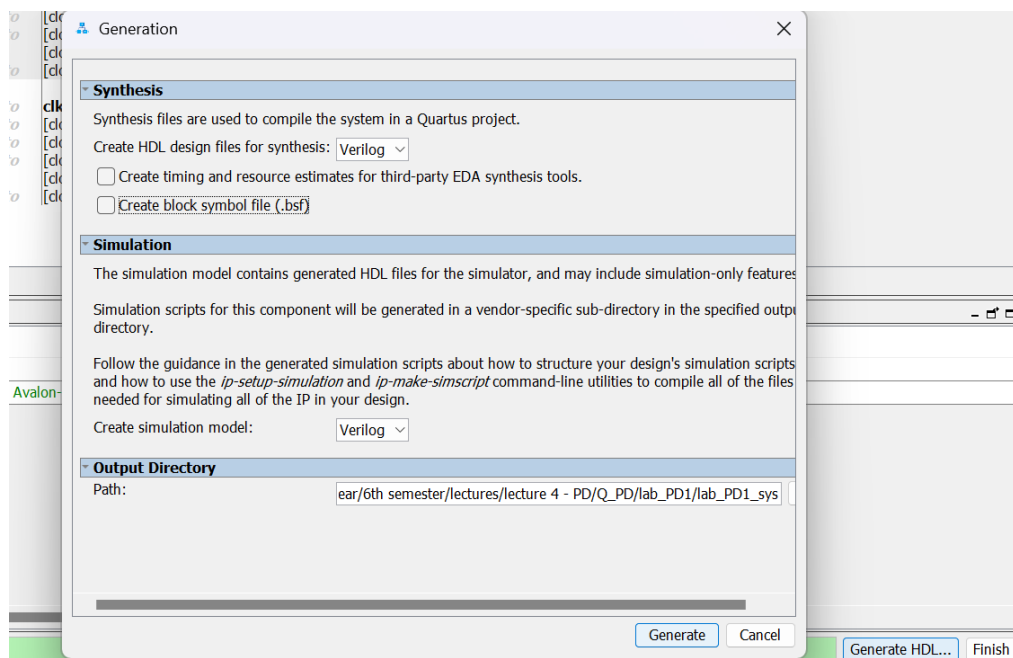


Рис. 28 – Предустановки окна Generation

Удостоверимся в том, что генерация прошла успешно:

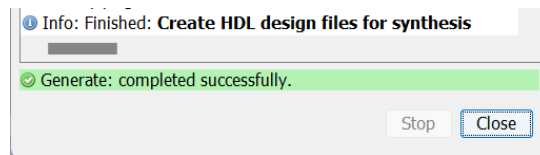


Рис. 29 – Проверка успешности генерации HDL

## 2.8. Конец работы

Закончим работу в PD, нажав кнопку Finish справа снизу, после чего выведется сообщение, где говорится о том, что при необходимости можно подключить файлы к проекту пакета QR:

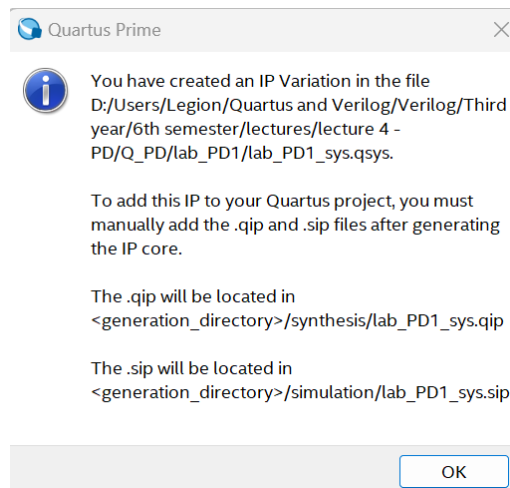


Рис. 30 – Сообщение с указанием пакетов, которые можно подключить

Проверим, что в системных файлах есть все необходимые файлы:

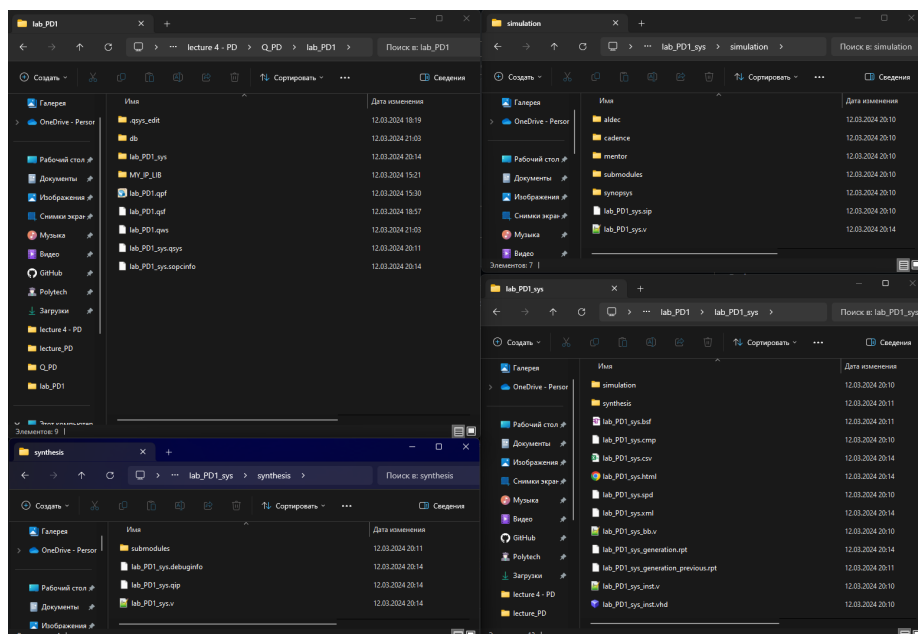


Рис. 31 – Анализ рабочей папки проекта



### 3. 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.