

Приложение Platform Designer

Приложение Platform Designer

Часть 1

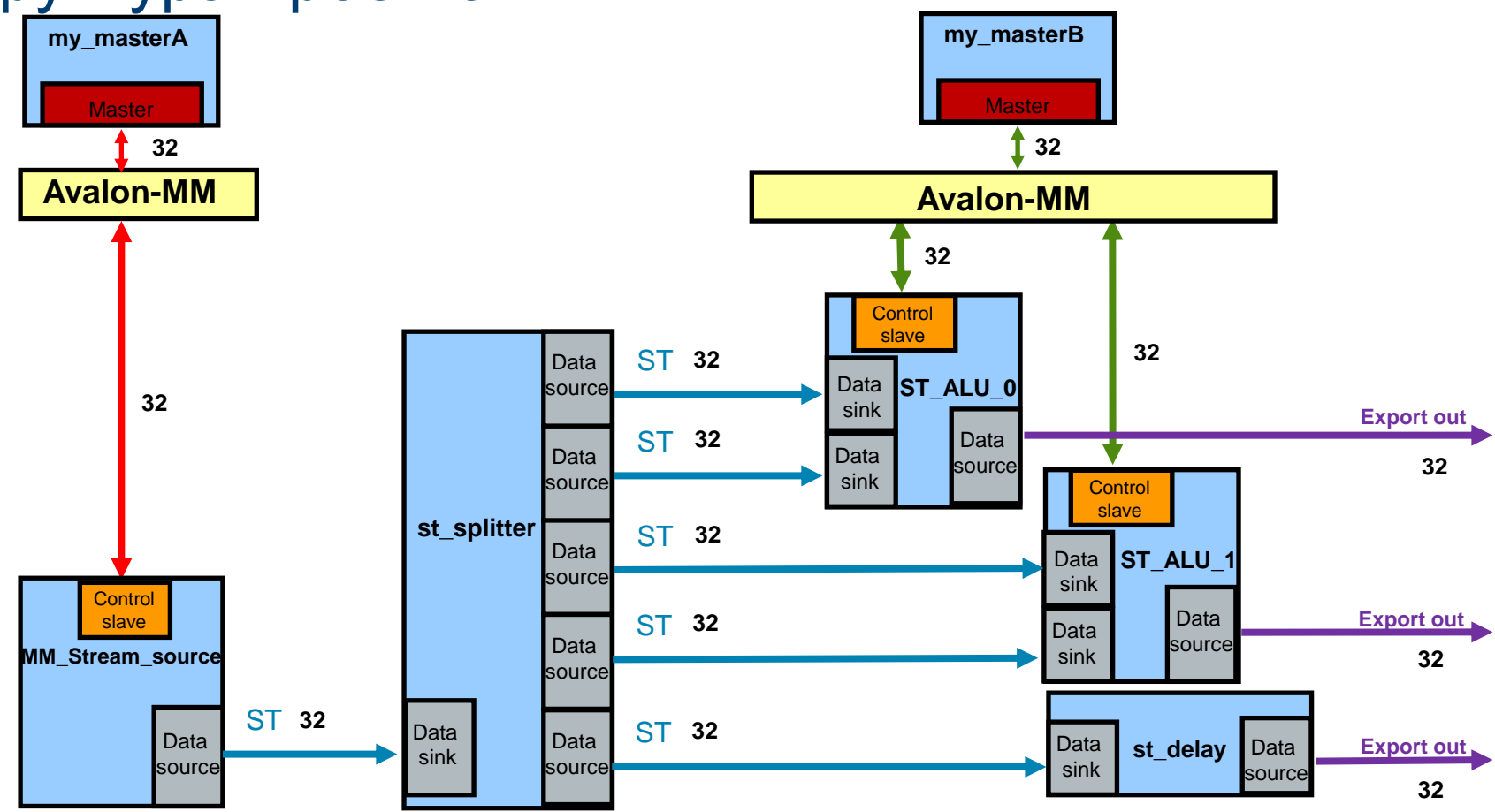
Пользовательский интерфейс



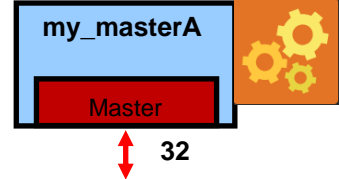
Лабораторная 1



Структура проекта



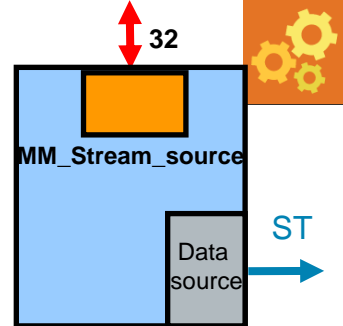
my_masterA



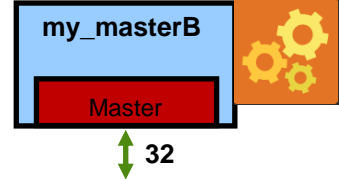
```
1  `timescale 1 ns / 1 ns
2  module my_masterA #(
3      parameter integer address = 0,
4      parameter integer data    = 100 //cnt_up = 100; cnt_down = 200
5  ) (
6      //MM master
7      input bit          csi_clk,           // MM master clk
8      input bit          rsi_reset,         // MM master reset
9      output bit [31:0]  avm_m0_address,    // MM master address
10     output bit          avm_m0_write,     // MM master write
11     output bit [31:0]  avm_m0_writedata,  // MM master writedata
12     input bit          avm_m0_waitrequest // MM master waitrequest
13 );
14 enum bit[1:0] {initSM, del1, wr1D, ended } fsm_MM;
15
16 always_ff @ (posedge csi_clk)
17 if (rsi_reset) fsm_MM <= initSM;
18 else
19     case (fsm_MM)
20         initSM : fsm_MM <= del1;
21         del1   : fsm_MM <= wr1D;
22         wr1D   : if (avm_m0_waitrequest) fsm_MM <= wr1D;
23                 else fsm_MM <= ended;
24         ended  : fsm_MM <= ended;
25     endcase
26
27 always_comb
28 begin
29     case (fsm_MM)
30         wr1D:
31             begin
32                 avm_m0_address = address;
33                 avm_m0_write   = 1'd1;
34                 avm_m0_writedata = data;
35             end
36         default
37             begin
38                 avm_m0_address = 32'd255;
39                 avm_m0_write   = 1'd0;
40                 avm_m0_writedata = 32'd255;
41             end
42     endcase
43 end
44 endmodule
```

MM_Stream_source

```
1  `timescale 1 ps / 1 ps
2  module MM_Stream_source (
3      //clk and reset
4      input bit csi_clk,           // clock clk
5      input bit rsi_reset,        // reset reset
6      //stream source
7      output bit [31:0] aso_out0_data, // ST source data
8      input bit aso_out0_ready, // ST source ready
9      output bit aso_out0_valid, // ST source valid
10     //MM slave
11     input bit [31:0] avs_s0_writedata, // MM slave writedata
12     input bit avs_s0_write, // MM slave write
13     output bit avs_s0_waitrequest // MM slave waitrequest
14 );
15
16 always_ff @(posedge csi_clk)
17     if(rsi_reset) aso_out0_data <= 32'd255;
18     else if (avs_s0_write & aso_out0_ready)
19         aso_out0_data <= avs_s0_writedata;
20
21 always_ff @(posedge csi_clk)
22     if(rsi_reset) aso_out0_valid <= 1'b0;
23     else
24         aso_out0_valid <= avs_s0_write & aso_out0_ready;
25
26 assign avs_s0_waitrequest = 1'b0;
27 endmodule
```



my_masterB



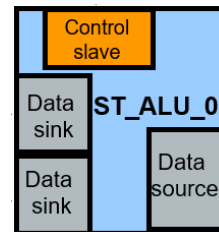
```
1  `timescale 1 ns / 1 ns
2  module my_masterB #(
3      parameter [31:0] address_1 = 0, parameter [31:0] data_1 = 111, //mult = 111  div  = 333
4      parameter [31:0] address_2 = 1, parameter [31:0] data_2 = 333  //add  = 222  sub = 444
5  ) (
6      input bit          csi_clk,           // MM master clk
7      input bit          rsi_reset,         // MM master reset
8      output bit [31:0]  avm_m0_address,    // MM master address
9      output bit         avm_m0_write,     // MM master write
10     output bit [31:0]  avm_m0_writedata,  // MM master writedata
11     input bit          avm_m0_waitrequest // MM master waitrequest
12 );
13 typedef enum bit[2:0] {initSM, del1, wr1D, del2, wr2D, ended } fsm_type;
14 fsm_type fsm_MM;
15
16 always_ff @ (posedge csi_clk)
17 if (rsi_reset) fsm_MM <= initSM;
18 else
19     case (fsm_MM)
20         initSM :          fsm_MM <= del1;
21         del1   :          fsm_MM <= wr1D;
22         wr1D   :          if (avm_m0_waitrequest) fsm_MM <= wr1D;
23                     else  fsm_MM <= del2;
24         del2   :          fsm_MM <= wr2D;
25         wr2D   :          if (avm_m0_waitrequest) fsm_MM <= wr2D;
26                     else  fsm_MM <= ended;
27         ended  :          fsm_MM <= ended;
28     endcase
29
30     always_comb
31     begin
32         case (fsm_MM)
33             wr1D:
34                 begin
35                     avm_m0_address      = address_1;
36                     avm_m0_write        = 1'd1;
37                     avm_m0_writedata    = data_1;
38                 end
39             wr2D:
40                 begin
41                     avm_m0_address      = address_2;
42                     avm_m0_write        = 1'd1;
43                     avm_m0_writedata    = data_2;
44                 end
45             default
46                 begin
47                     avm_m0_address      = 32'd255;
48                     avm_m0_write        = 1'd0;
49                     avm_m0_writedata    = 32'd255;
50                 end
51             endcase
52         end
53     endmodule
```

ST_ALU

```

1  `timescale 1 ns / 1 ns
2  module ST_ALU (
3      input bit        csi_clk,           // clock clk
4      input bit        rsi_reset,        // reset reset
5      // stream sink0
6      input bit [31:0] asi_in0_data,     // ST sink data
7      input bit        asi_in0_valid,    // ST sink valid
8      output bit       asi_in0_ready,    // ST sink ready
9      // stream sink1
10     input bit [31:0] asi_in1_data,     // ST sink data
11     input bit        asi_in1_valid,    // ST sink valid
12     output bit       asi_in1_ready,    // ST sink ready
13     //stream source
14     output bit [31:0] aso_out0_data,    // ST source data
15     input bit        aso_out0_ready,    // ST source ready
16     output bit       aso_out0_valid,    // ST source valid
17     //MM slave
18     input bit [31:0] avs_s0_writedata,  // MM slave writedata
19     input bit        avs_s0_write,      // MM slave write
20     output bit       avs_s0_waitrequest // MM slave waitrequest
21 );
22 bit [31:0] ALU_type; //Type ALU mult = 111 add = 222 div = 333 sub = 444
23
24 always_ff @(posedge csi_clk)
25     if(rsi_reset) ALU_type <= '0;
26     else if (avs_s0_write) ALU_type <= avs_s0_writedata;
27
28 always_ff @(posedge csi_clk)
29     if(rsi_reset) begin
30         aso_out0_data <= '0;
31         aso_out0_valid <= '0;
32     end
33     else
34         if (asi_in0_valid & asi_in1_valid & aso_out0_ready)
35             case (ALU_type)
36                 'd111 : begin aso_out0_data <= asi_in0_data * asi_in1_data; aso_out0_valid <= 1'b1; end
37                 'd222 : begin aso_out0_data <= asi_in0_data + asi_in1_data; aso_out0_valid <= 1'b1; end
38                 'd333 : begin aso_out0_data <= asi_in0_data / asi_in1_data; aso_out0_valid <= 1'b1; end
39                 'd444 : begin aso_out0_data <= asi_in0_data - asi_in1_data; aso_out0_valid <= 1'b1; end
40                 default : begin aso_out0_data <= '0; aso_out0_valid <= 1'b0; end
41             endcase
42
43 assign asi_in0_ready = 1'b1;
44 assign asi_in1_ready = 1'b1;
45 assign avs_s0_waitrequest = 1'b0;
46
47 endmodule

```





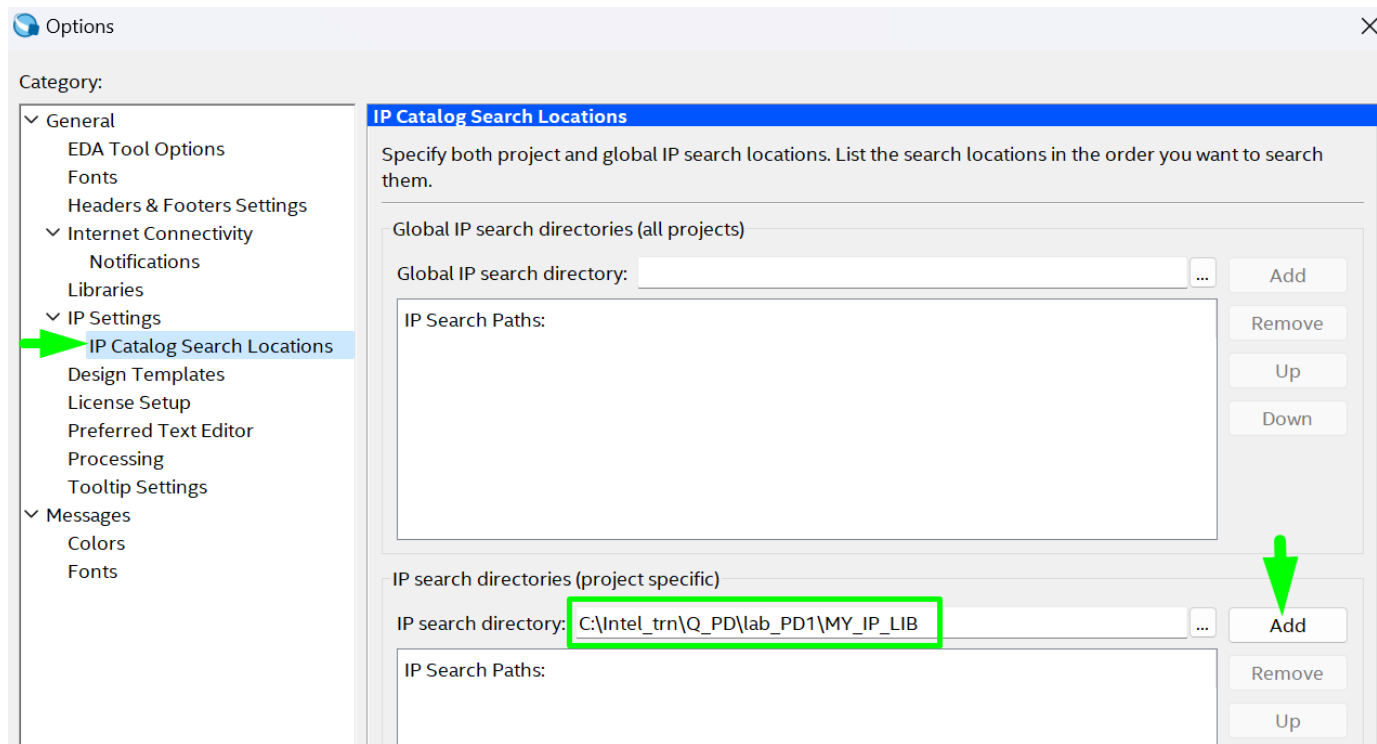
В QP создайте проект

- **Рабочая папка:** C:\Intel_trn\Q_PD\lab_PD1
- **Имя проекта:** lab_PD1
- **Модуль верхнего уровня:** lab_PD1
- **Тип проекта:** Empty Project
- **Файлы не добавляются**
- **Микросхема:** может быть любой
 - Плата DE1-SOC - 5CSEMA5F31C6N
 - Плата SoC Kit - 5CSXFC6D6F31
 - Плата MAX10_NEEK - 10M50DAF484C6G
 - Плата miniDilabCIV (**выбирается по умолчанию**) - EP4CE6E22C8
 - Плата DE0-nano - EP4CE22F17C6
- **EDA Tool Settings:** Simulation => ModelSim Altera Starter Edition=>SystemVerilog




В QR задайте путь к библиотеке IP

- Команда: Tools=>Options





В QR запустите приложение PD

- **Команда:** Tools => Platform Designer (или Qsys) или иконка 
- В PD: сохраните систему под именем lab_PD1_sys.qsys в рабочей папке проекта
- Убедитесь в том, что Ваша система выглядит так же, как показано на рисунке ниже

Qsys | lab_PD1_sys.qsys (C:\Intel_trn\Q_PD\lab_PD1\lab_PD1_sys.qsys)

File Edit System Generate View Tools Help

IP Catalog

Project

- New Component...
- System
 - lab_PD1_sys
- Training
 - MM_Stream_source_component
 - my_masterA_component
 - my_masterB_component
 - ST_ALU_component

Library

- Basic Functions

New... Edit... Add...

Hierarchy

- lab_PD1_sys [lab_PD1_sys.qsys]
 - clk
 - reset
 - clk_0

System Contents

Address Map

Interconnect Requirements

System: lab_PD1_sys

Use	Co...	Name	Description	Export	Clock	Base	End
<input checked="" type="checkbox"/>		clk_0	Clock Source				
		clk_in	Clock Input				
		clk_in_reset	Reset Input				
		clk	Clock Output				
		clk_reset	Reset Output				

Export: clk, reset

Clock: exported, clk_0

Current filter:

Messages

Type	Path	Message
------	------	---------

0 Errors, 0 Warnings

Generate HDL... Finish



Добавьте компоненты к системе

Для добавления компонента – двойной щелчок левой клавиши мыши.

*В появляющемся окне настройки каждого компонента нажмите **Finish** не изменяя настройки компонента - настройка компонентов будет осуществлена после их добавления к системе*

- my_masterA_component
- MM_stream_source_component
- my_masterB_component
- Avalon-ST Splitter (в строке поиска наберите ST)
- Avalon-ST Delay (в строке поиска наберите ST)
- ST_ALU_component (добавьте **два** компонента)

При добавлении компонентов на закладке Messages будут появляться сообщения об ошибках. На данном этапе на них можно не обращать внимание.



Проверьте систему

- Убедитесь в том, что Ваша система выглядит так же, как представленная на рисунке
- Сохраните файл.

Use	Connections	Name	Description	Export	Clock
✓		clk_0	Clock Source	clk	exported
✓		clk_in	Clock Input	clk	clk_0
✓		clk_in_reset	Reset Input	reset	
✓		clk	Clock Output	Double-click to Double-click to	
✓		clk_reset	Reset Output	Double-click to Double-click to	
✓		my_masterA_0	my_masterA_component		
✓		clock	Clock Input	Double-click to Double-click to	unconnected [clock]
✓		reset	Reset Input	Double-click to Double-click to	[clock]
✓		m0	Avalon Memory Mapped ...	Double-click to Double-click to	[clock]
✓		MM_Stream_source_0	MM_Stream_source_com...		
✓		clock	Clock Input	Double-click to Double-click to	unconnected [clock]
✓		reset	Reset Input	Double-click to Double-click to	[clock]
✓		out0	Avalon Streaming Source	Double-click to Double-click to	[clock]
✓		s0	Avalon Memory Mapped ...	Double-click to Double-click to	[clock]
✓		my_masterB_0	my_masterB_component		
✓		clock	Clock Input	Double-click to Double-click to	unconnected [clock]
✓		reset	Reset Input	Double-click to Double-click to	[clock]
✓		m0	Avalon Memory Mapped ...	Double-click to Double-click to	[clock]
✓		st_splitter_0	Avalon-ST Splitter		
✓		clk	Clock Input	Double-click to Double-click to	unconnected [clk]
✓		reset	Reset Input	Double-click to Double-click to	[clk]
✓		in	Avalon Streaming Sink	Double-click to Double-click to	[clk]
✓		out0	Avalon Streaming Source	Double-click to Double-click to	[clk]
✓		out1	Avalon Streaming Source	Double-click to Double-click to	[clk]
✓		st_delay_0	Avalon-ST Delay		
✓		in	Avalon Streaming Sink	Double-click to Double-click to	[clk]
✓		out	Avalon Streaming Sink	Double-click to Double-click to	[clk]
✓		clk	Clock Input	Double-click to Double-click to	unconnected [clk]
✓		clk_reset	Reset Input	Double-click to Double-click to	[clk]
✓		ST_ALU_0	ST_ALU_component		
✓		clock	Clock Input	Double-click to Double-click to	unconnected [clock]
✓		reset	Reset Input	Double-click to Double-click to	[clock]
✓		in0	Avalon Streaming Sink	Double-click to Double-click to	[clock]
✓		in1	Avalon Streaming Sink	Double-click to Double-click to	[clock]
✓		out0	Avalon Streaming Source	Double-click to Double-click to	[clock]
✓		s0	Avalon Memory Mapped ...	Double-click to Double-click to	[clock]
✓		ST_ALU_1	ST_ALU_component		
✓		clock	Clock Input	Double-click to Double-click to	unconnected [clock]
✓		reset	Reset Input	Double-click to Double-click to	[clock]
✓		in0	Avalon Streaming Sink	Double-click to Double-click to	[clock]
✓		in1	Avalon Streaming Sink	Double-click to Double-click to	[clock]
✓		out0	Avalon Streaming Source	Double-click to Double-click to	[clock]
✓		s0	Avalon Memory Mapped ...	Double-click to Double-click to	[clock]

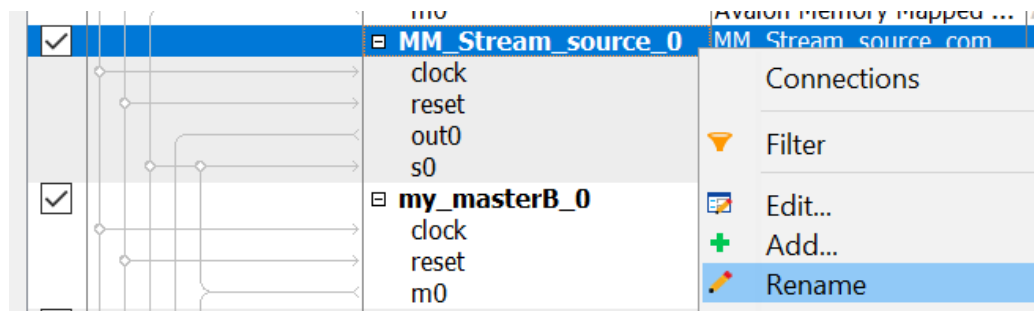
Current filter:

Type	Path	Message
✗	14 Errors	
✗	lab_PD1_sys.my_masterA_0	my_masterA_0.clock must be connected to a clock output
✗	lab_PD1_sys.MM_Stream_source_0	MM_Stream_source_0.clock must be connected to a clock output



Переименуйте компоненты

- Щелчком выделите компонент **MM_Stream_source_0**
- Нажмите правую клавишу мыши
- Выберите команду Rename
- Измените имя компонента на **MM_Stream_source**
- Повторите процедуру для компонентов:
 - my_masterA_0 => Новое имя: **my_masterA**
 - my_masterB_0 => Новое имя: **my_masterB**
 - st_splitter_0 => Новое имя: **st_splitter**
 - st_delay_0 => Новое имя: **st_delay**
 - clk_0 => Новое имя: **clk**





Проверьте систему

- Убедитесь в том, что Ваша система выглядит так же, как представленная на рисунке
- Сохраните файл.

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	Clock Output	Double-click to	clk
		clk_reset	Reset Output	Double-click to	
<input checked="" type="checkbox"/>		my_masterA	my_masterA_component		
		clock	Clock Input	Double-click to	unconnected
		reset	Reset Input	Double-click to	[clock]
		m0	Avalon Memory Mapped ...	Double-click to	[clock]
<input checked="" type="checkbox"/>		MM_Stream_source	MM_Stream_source_com...		
		clock	Clock Input	Double-click to	unconnected
		reset	Reset Input	Double-click to	[clock]
		out0	Avalon Streaming Source	Double-click to	[clock]
		s0	Avalon Memory Mapped ...	Double-click to	[clock]
<input checked="" type="checkbox"/>		my_masterB	my_masterB_component		
		clock	Clock Input	Double-click to	unconnected
		reset	Reset Input	Double-click to	[clock]
		m0	Avalon Memory Mapped ...	Double-click to	[clock]
<input checked="" type="checkbox"/>		st_splitter	Avalon-ST Splitter		
		clk	Clock Input	Double-click to	unconnected
		reset	Reset Input	Double-click to	[clk]
		in	Avalon Streaming Sink	Double-click to	[clk]
		out0	Avalon Streaming Source	Double-click to	[clk]
		out1	Avalon Streaming Source	Double-click to	[clk]
<input checked="" type="checkbox"/>		st_delay	Avalon-ST Delay		
		in	Avalon Streaming Sink	Double-click to	[clk]
		out	Avalon Streaming Sink	Double-click to	[clk]
		clk	Clock Input	Double-click to	unconnected
		clk_reset	Reset Input	Double-click to	[clk]
<input checked="" type="checkbox"/>		ST_ALU_0	ST_ALU_component		
		clock	Clock Input	Double-click to	unconnected
		reset	Reset Input	Double-click to	[clock]
		in0	Avalon Streaming Sink	Double-click to	[clock]
		in1	Avalon Streaming Sink	Double-click to	[clock]
		out0	Avalon Streaming Source	Double-click to	[clock]
		s0	Avalon Memory Mapped ...	Double-click to	[clock]
<input checked="" type="checkbox"/>		ST_ALU_1	ST_ALU_component		
		clock	Clock Input	Double-click to	unconnected
		reset	Reset Input	Double-click to	[clock]
		in0	Avalon Streaming Sink	Double-click to	[clock]
		in1	Avalon Streaming Sink	Double-click to	[clock]
		out0	Avalon Streaming Source	Double-click to	[clock]
		s0	Avalon Memory Mapped ...	Double-click to	[clock]



Подключите тактовый сигнал

- На закладке System Contents щелчком выделите интерфейс **clk.clk** (интерфейс clk компонента clk)
- Выполните команду меню View=>Connections
- В появившемся окне закладки Connections выберите подключение ко всем тактовым входам
- Переключитесь на закладку System Contents
- Нажмите правую клавишу мыши
- Выберите команду Filter=>Clock and Reset Interfaces
- Убедитесь, что соединения выполнены - Ваша система выглядит так же, как представленная на рисунке

System Contents Address Map Interconnect Requirements Connections

System: lab_PD1_sys Path: clk.clk

Connected to: clk.clk

Connected	Connection	Clock Crossing
✓	clk.clk/MM_Stream_source.clk	
✓	clk.clk/ST_ALU_0.clk	
✓	clk.clk/ST_ALU_1.clk	
✓	clk.clk/my_masterA.clk	
✓	clk.clk/my_masterB.clk	
✓	clk.clk/st_delay.clk	
✓	clk.clk/st_splitter.clk	


System Contents Address Map Interconnect Requirements Connections

System: lab_PD1_sys Path: clk.clk

Use	Co...	Name	Description	Export	Clock
		clk	Clock Source		
		clk_in	Clock Input		
		clk_in_reset	Reset Input	clk reset	<i>exported</i>
		clk	Clock Output	<i>Double-click to</i>	clk
		clk_reset	Reset Output	<i>Double-click to</i>	
		my_masterA	my_masterA_component		
		clock reset	Clock Input	<i>Double-click to</i>	clk
		reset	Reset Input	<i>Double-click to</i>	[clock]
		MM_Stream_source	MM_Stream_source_com...		
		clock reset	Clock Input	<i>Double-click to</i>	clk
		reset	Reset Input	<i>Double-click to</i>	[clock]
		my_masterB	my_masterB_component		
		clock reset	Clock Input	<i>Double-click to</i>	clk
		reset	Reset Input	<i>Double-click to</i>	[clock]
		st_splitter	Avalon-ST Splitter		
		clk reset	Clock Input	<i>Double-click to</i>	clk
		reset	Reset Input	<i>Double-click to</i>	[ck]
		st_delay	Avalon-ST Delay		
		clk clock reset	Clock Input	<i>Double-click to</i>	clk
		reset	Reset Input	<i>Double-click to</i>	[ck]
		ST_ALU_0	ST_ALU_component		
		clock reset	Clock Input	<i>Double-click to</i>	clk
		reset	Reset Input	<i>Double-click to</i>	[clock]
		ST_ALU_1	ST_ALU_component		
		clock reset	Clock Input	<i>Double-click to</i>	clk
		reset	Reset Input	<i>Double-click to</i>	[clock]




Подключите сигнал Reset

- На закладке System Contents выполните команду
 - меню **System=>Create Global Reset Network**
- Убедитесь, что соединения выполнены - Ваша система выглядит так же, как представленная на рисунке
- Сбросьте фильтрацию – нажмите на иконку  в нижней части окна System Contents
- Сохраните файл

System Contents | Address Map | Interconnect Requirements | Connections

System: lab_PD1_sys Path: clk.clk_reset

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

 **Current filter:** Clock and Reset Interfaces



Подключите Avalon-MM интерфейсы

- На закладке System Contents щелчком выделите интерфейс **my_masterA.m0**
- Нажмите правую клавишу мыши
- Выберите команду **Filter=> Avalon-MM Interfaces**
- В столбце Connections выполните подключения так, как показано на рисунке
- Сохраните файл

System Contents | Address Map | Interconnect Requirements | Connections

System: lab_PD1_sys Path: my_masterA.m0

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 ...	Double-click to	clk		
<input checked="" type="checkbox"/>		MM_Stream_source	MM_Stream_source_com...	Double-click to	clk		
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped ...	Double-click to	clk	0x0000_0000	0x0000_0003
<input checked="" type="checkbox"/>		my_masterB	my_masterB_component		[clock]		
<input checked="" type="checkbox"/>		m0	Avalon Memory Mapped ...	Double-click to	clk		
<input checked="" type="checkbox"/>		ST_ALU_0	ST_ALU_component	Double-click to	clk	0x0000_0000	0x0000_0003
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped ...	Double-click to	clk	0x0000_0000	0x0000_0003
<input checked="" type="checkbox"/>		ST_ALU_1	ST_ALU_component	Double-click to	clk	0x0000_0000	0x0000_0003
<input checked="" type="checkbox"/>		s0	Avalon Memory Mapped ...	Double-click to	clk	0x0000_0000	0x0000_0003

Current filter: Avalon-MM Interfaces

Messages


Type	Path	Message
Error	1 Error	
Error	lab_PD1_sys.my_masterB.m0	ST_ALU_1.s0 (0x0..0x3) overlaps ST_ALU_0.s0 (0x0..0x3)
Warning	12 Warnings	



Назначьте базовые адреса ведомым Avalon-MM

- Выберите команду меню **System=> Assign Base Addresses**
- Убедитесь, что адреса назначены - Ваша система выглядит так же, как представленная на рисунке

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



Закладка Address Map

- Выберите команду меню **View=> Address Map**

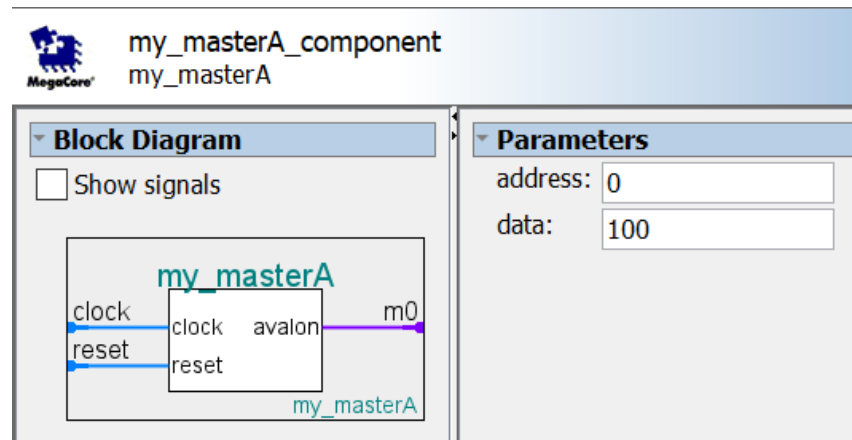
System Contents	Address Map	Interconnect Requirements	Connections
System: lab_PD1_sys Path: MM_Stream_source.s0			
	my_masterA.m0	my_masterB.m0	
MM_Stream_source.s0	0x0000 0000 - 0x0000 0003		
ST_ALU_0.s0		0x0000 0004 - 0x0000 0007	
ST_ALU_1.s0		0x0000 0000 - 0x0000 0003	

- В окне закладки изображены
 - Ведущие (Master) шины Avalon_MM – столбцы;
 - Ведомые (Slave) шины Avalon_MM – строки
- Убедитесь, что у Вас окно выглядит так же, как представлено на рисунке
- Запомните базовые адреса (они нужны для настройки Ведущих):
 - MM_Stream_source = 0
 - ST_ALU_0 = 4
 - ST_ALU_1 = 0



Настройка компонента my_masterA

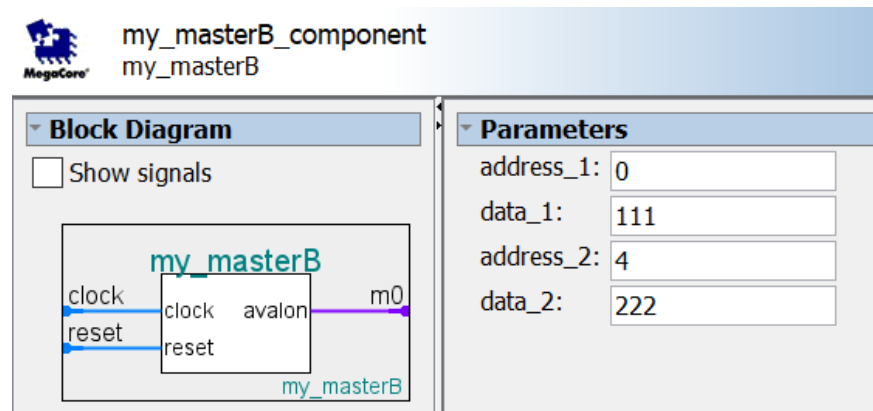
- Переключитесь на закладку System Contents
- Щелчком выберите **my_masterA**
- Нажмите правую клавишу мыши и выберите команду Edit
- В появившемся окне настройки параметров задайте
 - Адрес, по которому Ведущий **my_masterA** будет записывать данные
 - В нашей системе – это адрес **0** (это базовый адрес Ведомого MM_Stream_source)
 - Записываемые данные: **100** – счет на сложение; 200 – счет на вычитание
 - Нажмите кнопку **Finish**
- Сохраните файл






Настройка компонента my_masterB

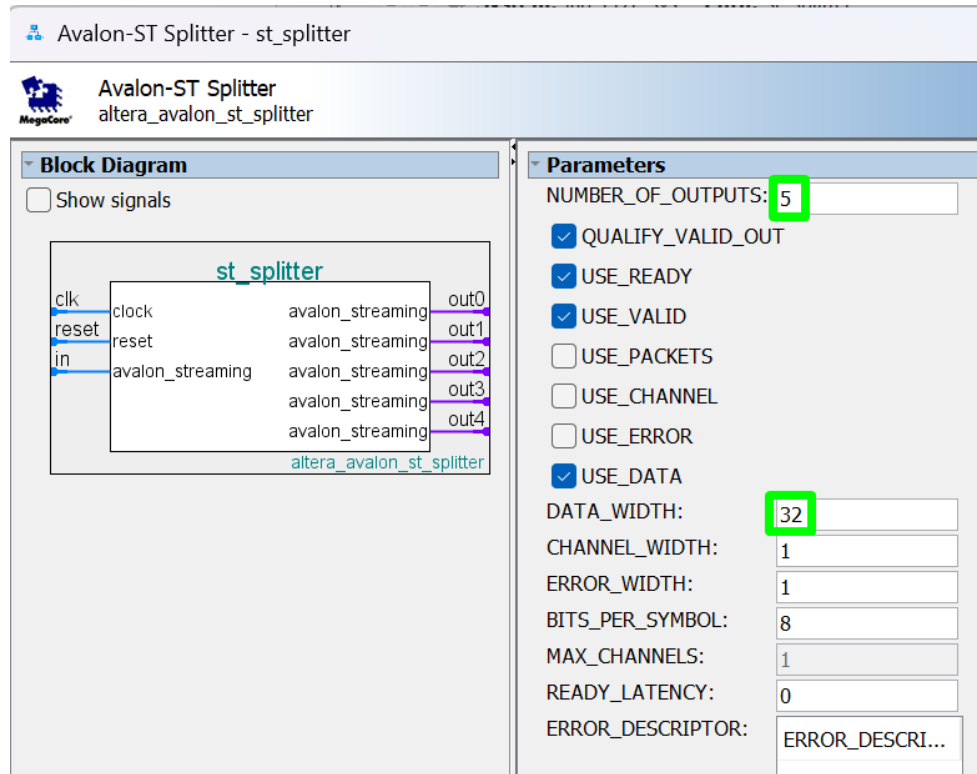
- Щелчком выберите **my_masterB**
- Нажмите правую клавишу мыши и выберите команду Edit
- В появившемся окне задайте
 - Адреса, по которому Ведущий **my_masterB** будет записывать данные
 - 0** (это базовый адрес Ведомого ST_ALU_1)
 - 4** (это базовый адрес Ведомого ST_ALU_0)
 - Записываемые данные:
 - 111** – сложение (тип операции ST_ALU_1)
 - 222** – умножение (тип операции ST_ALU_0) ;
 - Нажмите кнопку **Finish**
- Сохраните файл





Настройка компонента `st_splitter`

- Сбросьте фильтрацию – нажмите на иконку  в нижней части окна System Contents
- На закладке System Contents щелчком выделите компонент `st_splitter`
- Нажмите правую клавишу мыши
- Выберите команду **Edit**
- Установите
 - `NUMBER_OF_OUTPUTS` = 5
 - `DATA_WIDTH` = 32
- Нажмите кнопку Finish
- Сохраните файл



The screenshot shows the configuration window for the 'Avalon-ST Splitter' component. The window is titled 'Avalon-ST Splitter - st_splitter'. It features a 'Block Diagram' tab on the left and a 'Parameters' tab on the right.

Block Diagram: The diagram shows the 'st_splitter' component with inputs 'clk' (clock), 'reset', and 'in' (avalon_streaming). It has five outputs labeled 'out0' through 'out4', each connected to an 'avalon_streaming' signal. The component is identified as 'altera_avalon_st_splitter'.

Parameters: The 'Parameters' tab is active, showing the following settings:


- `NUMBER_OF_OUTPUTS`: 5 (highlighted with a green box)
- `QUALIFY_VALID_OUT`: ☒
- `USE_READY`: ☒
- `USE_VALID`: ☒
- `USE_PACKETS`: ☐
- `USE_CHANNEL`: ☐
- `USE_ERROR`: ☐
- `USE_DATA`: ☒
- `DATA_WIDTH`: 32 (highlighted with a green box)
- `CHANNEL_WIDTH`: 1
- `ERROR_WIDTH`: 1
- `BITS_PER_SYMBOL`: 8
- `MAX_CHANNELS`: 1
- `READY_LATENCY`: 0
- `ERROR_DESCRIPTOR`: ERROR_DESCRI...



Настройка компонента `st_delay`


- На закладке System Contents щелчком выделите компонент `st_delay_0`
- Нажмите правую клавишу мыши
- Выберите команду **Edit**
- Установите
 - `DATA_WIDTH = 32`
- Нажмите кнопку Finish
- Сохраните файл

Avalon-ST Delay - st_delay

 Avalon-ST Delay
altera_avalon_st_delay

Block Diagram

☐ Show signals



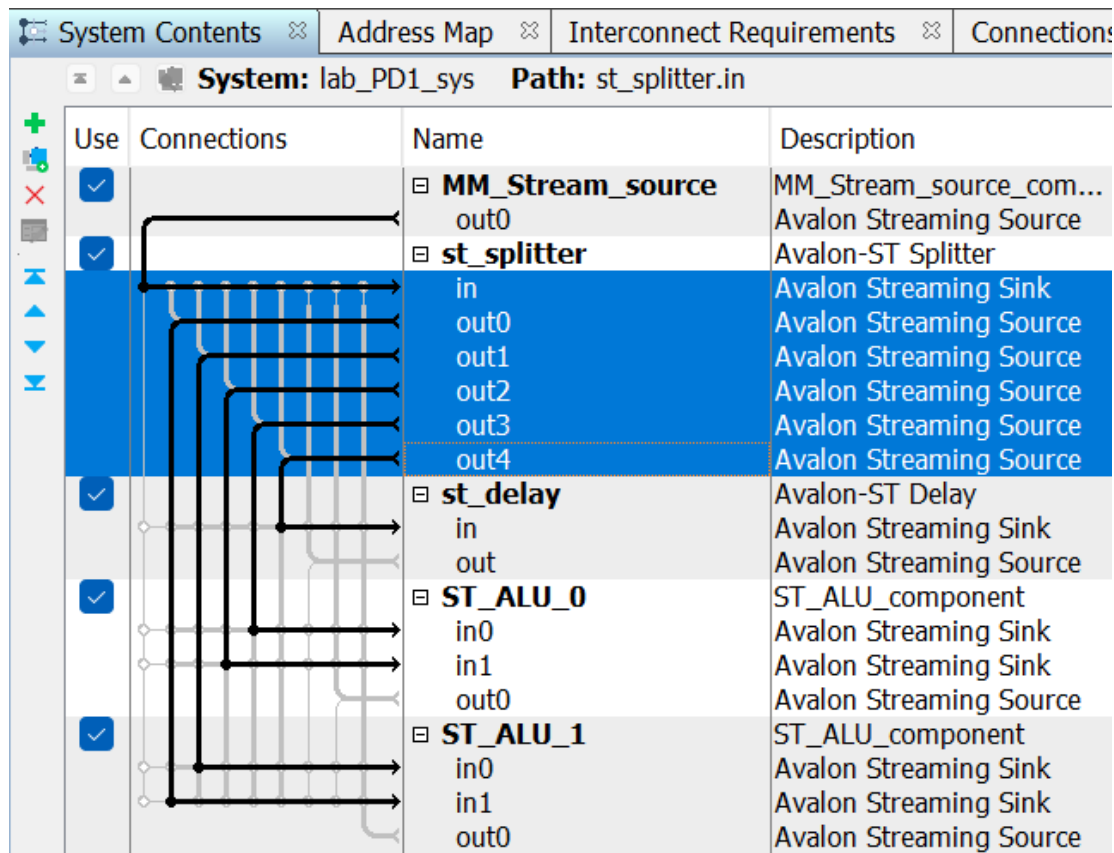
Parameters

NUMBER_OF_DELAY_CLOCKS:	1
DATA_WIDTH:	32
BITS_PER_SYMBOL:	8
<input type="checkbox"/> USE_PACKETS	
<input type="checkbox"/> USE_CHANNEL	
CHANNEL_WIDTH:	1
MAX_CHANNELS:	1
<input type="checkbox"/> USE_ERROR	
ERROR_WIDTH:	1



Подключите Avalon-ST интерфейсы

- На закладке System Contents щелчком выделите интерфейс **MM_Stream_source.out0**
- Нажмите правую клавишу мыши
- Выберите команду **Filter=> Avalon-ST Interfaces**
- В столбце Connections выполните подключения так, как показано на рисунке
- Сохраните файл





Экспортируйте выводы

- На закладке System Contents щелчком выделите интерфейс **st_delay.out0**
- Дважды щелкните в поле Export и задайте имя delay_out
- Щелчком выделите интерфейс **ST_ALU_0.out0**
- Дважды щелкните в поле Export и задайте имя alu0_out
- Щелчком выделите интерфейс **ST_ALU_1.out0**
- Дважды щелкните в поле Export и задайте имя alu1_out
- Сбросьте фильтрацию – нажмите на иконку  в нижней части окна System Contents
- Сохраните файл

st_delay_0	Avalon-ST Delay	
in	Avalon Streaming Sink	Double-click to
out	Avalon Streaming Source	delay_out

ST_ALU_0	ST_ALU_component	
in0	Avalon Streaming Sink	Double-click to
in1	Avalon Streaming Sink	Double-click to
out0	Avalon Streaming Source	alu0_out

ST_ALU_1	ST_ALU_component	
in0	Avalon Streaming Sink	Double-click to
in1	Avalon Streaming Sink	Double-click to
out0	Avalon Streaming Source	alu1_out

Проверьте систему

- Убедитесь в том, что:
 - Ваша система выглядит так же, как представленная на рисунке
 - Закладка сообщений (Messages) содержит только одно информационное сообщение.

System: lab_PD1_sys Path: clk

Use	Connections	Name	Description	Export	Clock	Base	End
		clk	Clock Source		exported		
		clk_in	Clock Input	clk			
		clk_in_reset	Reset Input	reset			
		clk_out	Clock Output	clk			
		clk_reset	Reset Output	reset			
		my_masterA	my_masterA_component				
		clock	Clock Input	clk			
		reset	Reset Input	reset			
		m0	Avalon Memory Mapped ...				
		MM_Stream_source	MM_Stream_source_com...				
		clock	Clock Input	clk			
		reset	Reset Input	reset			
		out0	Avalon Streaming Source				
		s0	Avalon Memory Mapped ...				
		my_masterB	my_masterB_component				
		clock	Clock Input	clk			
		reset	Reset Input	reset			
		m0	Avalon Memory Mapped ...				
		st_splitter	Avalon-ST Splitter				
		clk	Clock Input	clk			
		reset	Reset Input	reset			
		in	Avalon Streaming Sink				
		out0	Avalon Streaming Source				
		out1	Avalon Streaming Source				
		out2	Avalon Streaming Source				
		out3	Avalon Streaming Source				
		out4	Avalon Streaming Source				
		st_delay	Avalon-ST Delay				
		in	Avalon Streaming Sink				
		out	Avalon Streaming Source				
		clk	Clock Input	clk			
		clk_reset	Reset Input	reset			
		ST_ALU_0	ST_ALU_component				
		clock	Clock Input	clk			
		reset	Reset Input	reset			
		in0	Avalon Streaming Sink				
		in1	Avalon Streaming Sink				
		out0	Avalon Streaming Source				
		s0	Avalon Memory Mapped ...				
		ST_ALU_1	ST_ALU_component				
		clock	Clock Input	clk			
		reset	Reset Input	reset			
		in0	Avalon Streaming Sink				
		in1	Avalon Streaming Sink				
		out0	Avalon Streaming Source				
		s0	Avalon Memory Mapped ...				

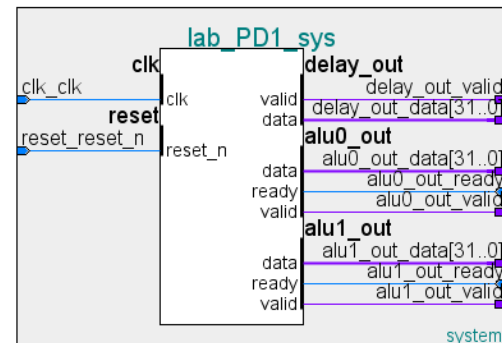
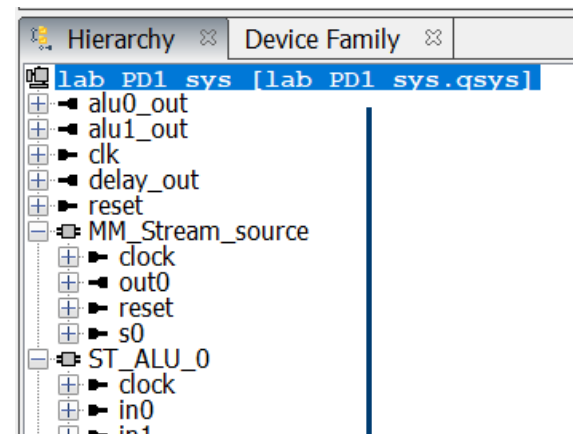
Current filter: All Interfaces

Messages

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.

Анализ системы

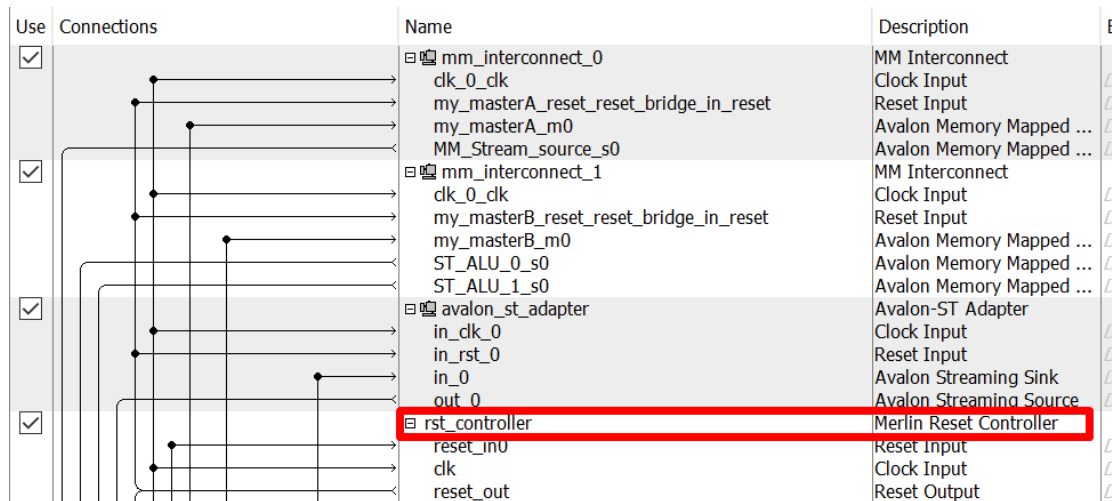
- Выполните команду: меню View=>Hierarchy
- В окне закладки Hierarchy выделите систему lab_PD1_sys [lab_PD1_sys.qsys]
- Выполните команду: меню View=>Block Symbol
- Убедитесь, что Ваш символ системы соответствует представленному на рисунке





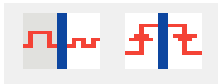
Анализ системы

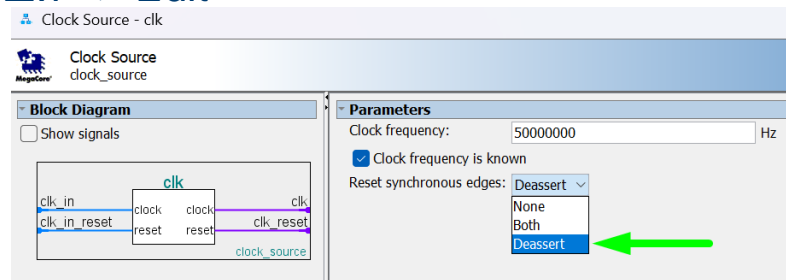
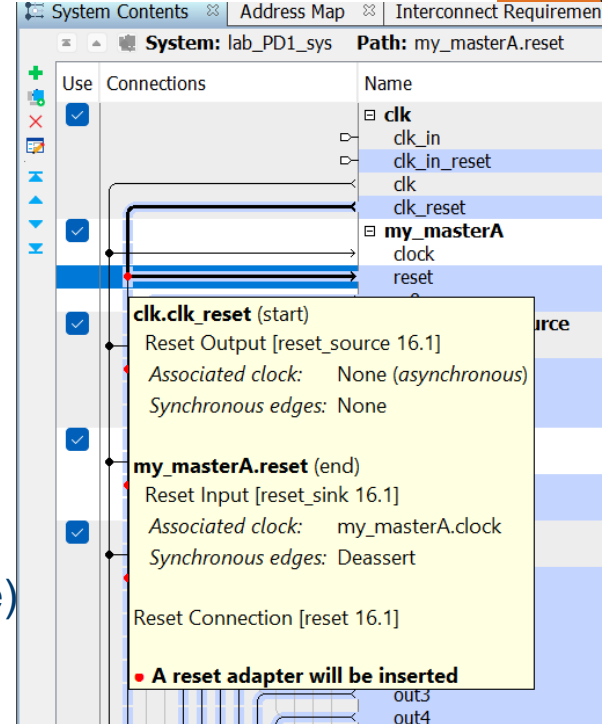
- Выполните команду: меню **System => Show System with Platform Designer Interconnect** (**Show System with QSYS Interconnect**)
- сравните созданную Вами систему и систему с модулями вставленными PD:
 - Какие модули были добавлены? Зафиксируйте их имена.
 - Обратите внимание на наличие модуля `rst_controller` – это адаптер сигнала Reset
- Закройте окно закладки



Use	Connections	Name	Description	E
<input checked="" type="checkbox"/>		mm_interconnect_0	MM Interconnect	
		clk_0_clk	Clock Input	
		my_masterA_reset_reset_bridge_in_reset	Reset Input	
		my_masterA_m0	Avalon Memory Mapped ...	
		MM_Stream_source_s0	Avalon Memory Mapped ...	
<input checked="" type="checkbox"/>		mm_interconnect_1	MM Interconnect	
		clk_0_clk	Clock Input	
		my_masterB_reset_reset_bridge_in_reset	Reset Input	
		my_masterB_m0	Avalon Memory Mapped ...	
		ST_ALU_0_s0	Avalon Memory Mapped ...	
		ST_ALU_1_s0	Avalon Memory Mapped ...	
<input checked="" type="checkbox"/>		avalon_st_adapter	Avalon-ST Adapter	
		in_clk_0	Clock Input	
		in_rst_0	Reset Input	
		in_0	Avalon Streaming Sink	
		out_0	Avalon Streaming Source	
<input checked="" type="checkbox"/>		rst_controller	Merlin Reset Controller	
		reset_in0	Reset Input	
		clk	Clock Input	
		reset_out	Reset Output	

Анализ системы

- Выполните команду: меню View => Clock domains Beta
- Выберите режим отображения **Reset**

Clocks **Resets**
- В столбце Connections отображаются красные точки => есть проблемы подключения:
 - Выход clk.clk_reset асинхронный
 - Входы clk_reset всех модулей – синхронны.=> Поставлен адаптер Reset (видели на предыдущем слайде)
- Следует внести исправление в модуль clk:
 - Надо выбрать модуль => правая клавиша мыши => Edit
 - Установить Deassert
 - Нажать Finish
- Красные точки исчезнут



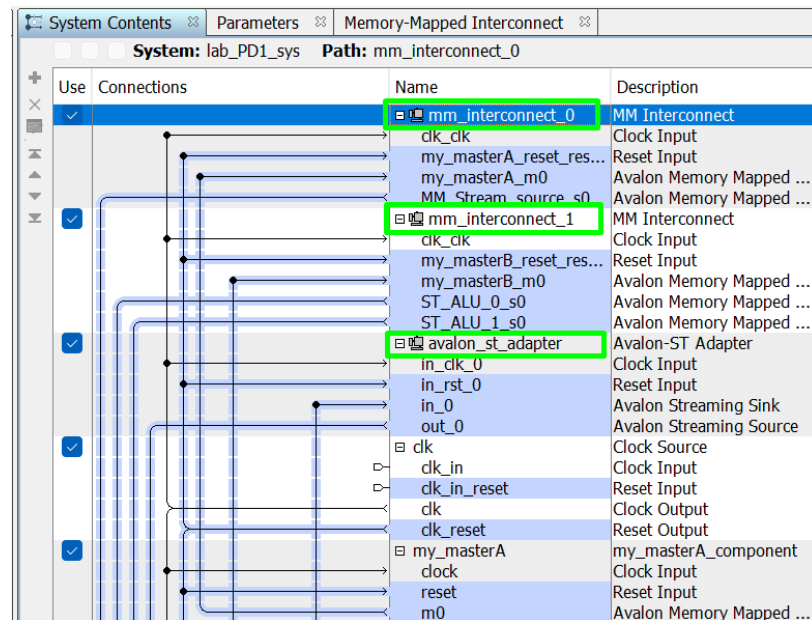


Анализ системы

- Еще раз выполните команду: меню **System => Show System with Platform Designer Interconnect**

(Show System with QSYS Interconnect)

- Убедитесь в том, что адаптера Reset нет в системе.
- Назначение добавленных модулей
 - mm_interconnect_0 система межсоединений для Ведущего my_masterA
 - mm_interconnect_1 система межсоединений для Ведущего my_master
 - 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

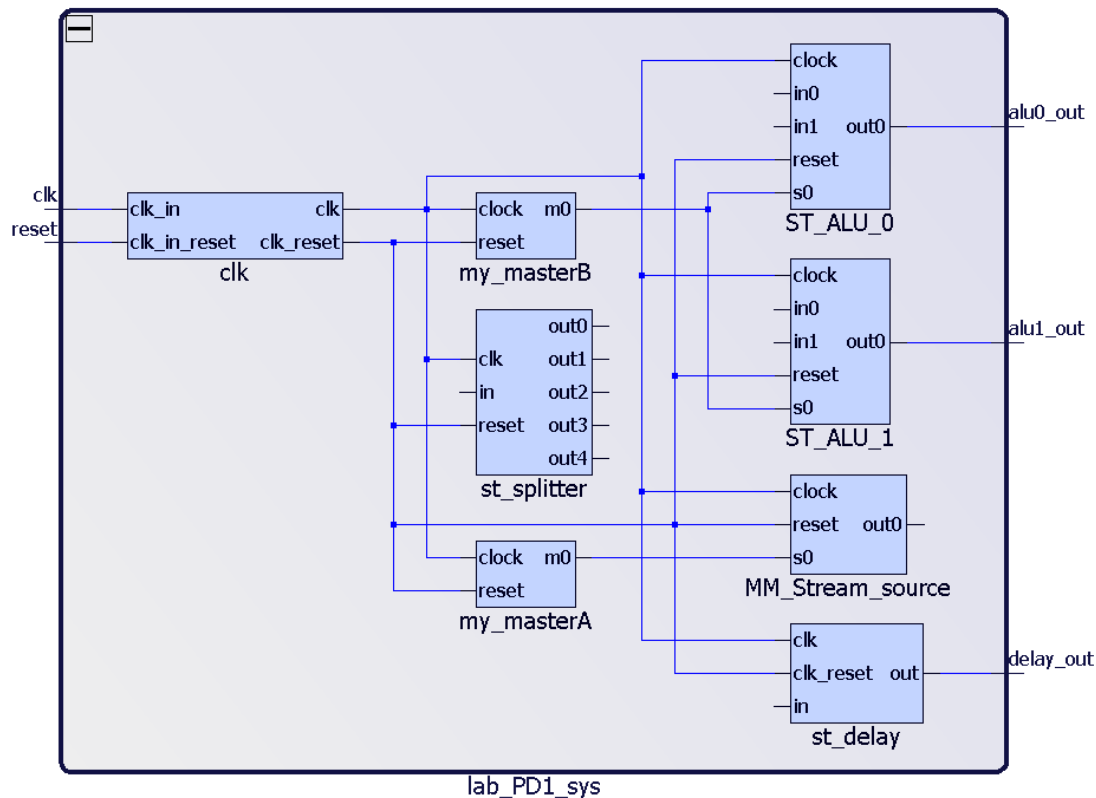


Messages		
Type	Path	Message
1 Info Message		
lab_PD1_sys.st_splitter.out4/st_delay.in		The source has a ready signal of 1 bits, but the sink does not. Avalon-ST Adapter will be inserted.



Анализ системы

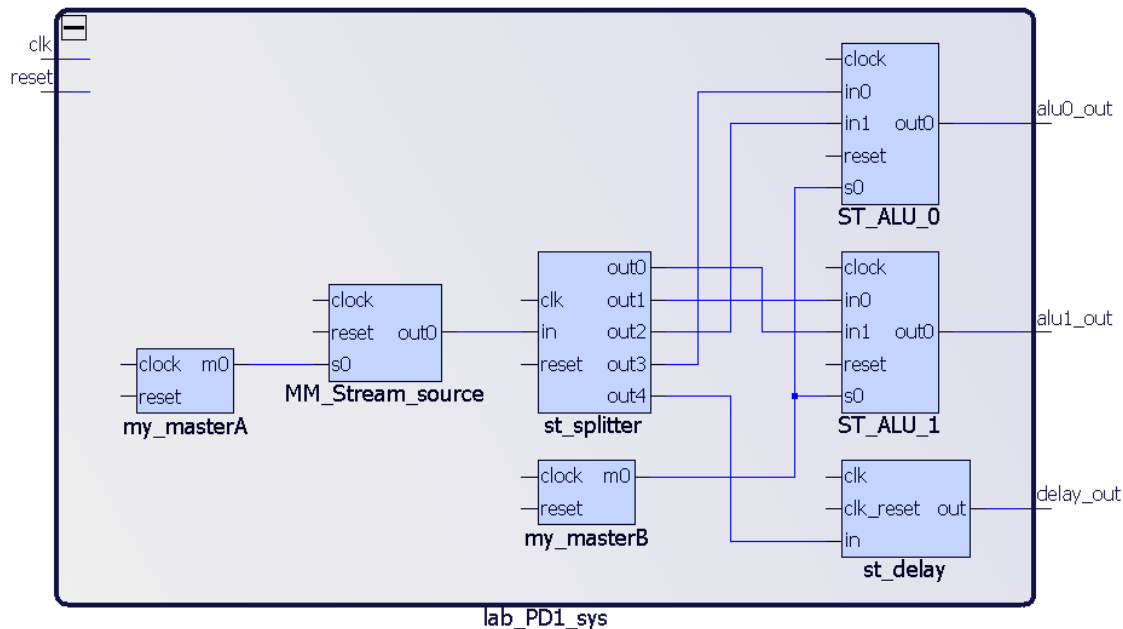
- Выполните команду: меню **View=>Schematic**
- В поле фильтра введите
 - in
 - **Нажмите Enter**
- Убедитесь в том, что система синхронизации и каналы ST Вашей системы подключены так же, как это изображено на рисунке





Анализ системы

- В поле фильтра введите
 - **clk**
 - **Нажмите Enter**
- Убедитесь в том, что шины Avalon MM в Вашей системе подключены так же, как рисунок
- Закройте окно закладки Schematic



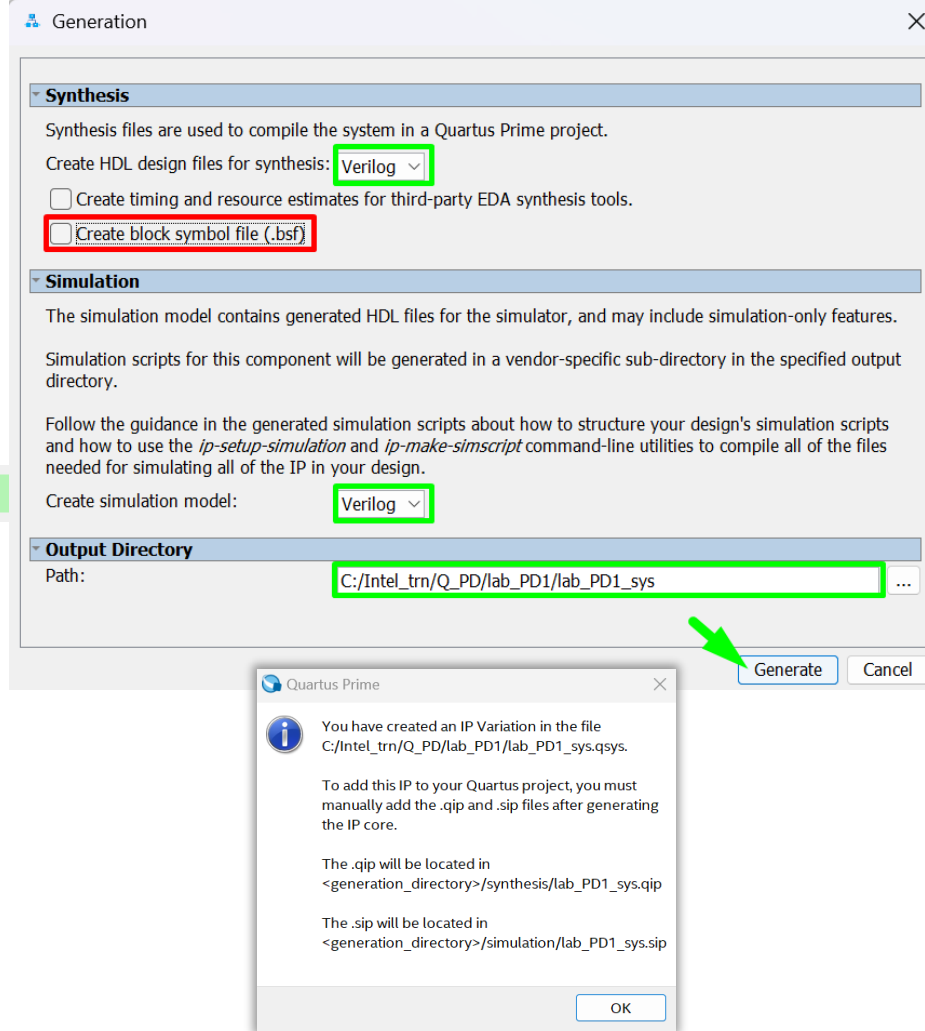
Генерация системы

- В окне PD нажмите кнопку Generate HDL... (правый нижний угол окна)
- Установите режимы так, как показано на рисунке
- Нажмите кнопку Generate
- По окончании процедуры появится сообщение

✔ Generate: completed successfully.

– Нажмите кнопку Close

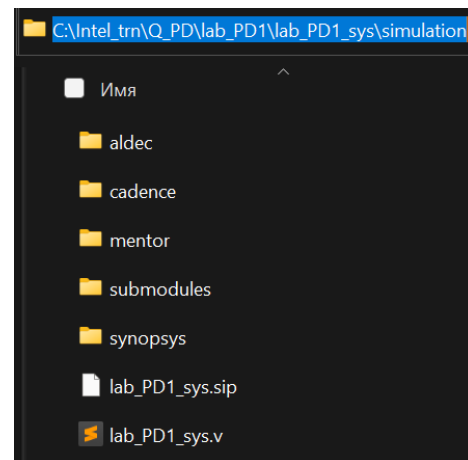
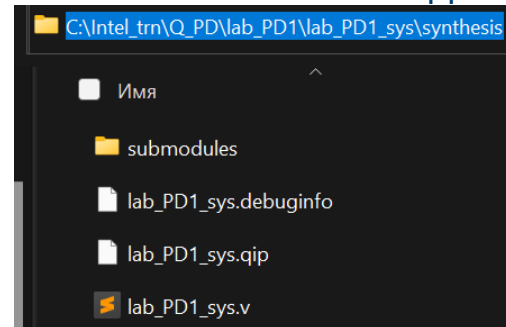
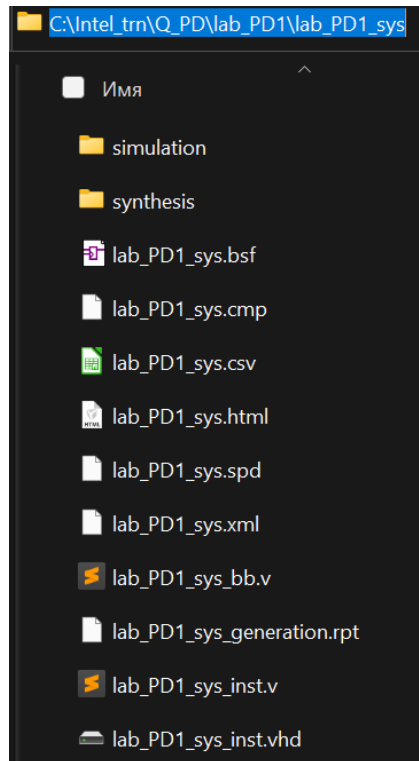
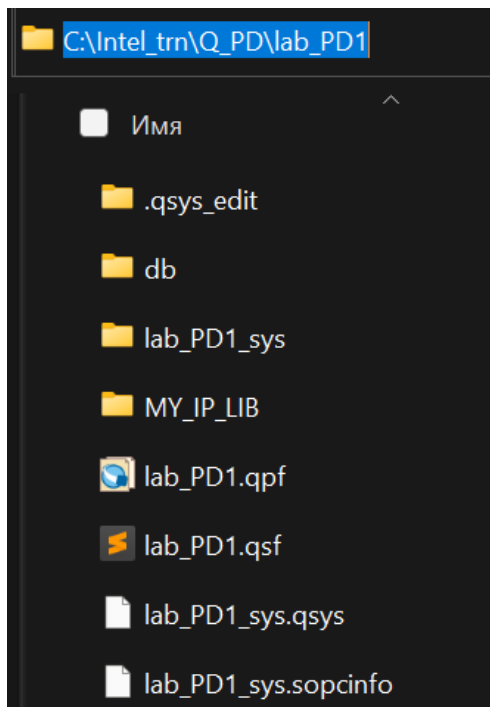
- В окне PD нажмите кнопку Finish (правый нижний угол окна)
- Появится напоминание о необходимости подключить файлы к проекту пакета QP
- Нажмите кнопку OK.





Анализ рабочей папки проекта

- Откройте рабочую папку проекта и проведите анализ файлов и папок.
 - Сравните с файлами в рабочей папке проекта на Вашем ПК – они должны совпадать





Лабораторная 1
ЗАВЕРШЕНА!