



СиФО ВМ Л.Р. №3

Стракович Андрей Иванович

БГУИР, кафедра ЭВМ

а. 508-5

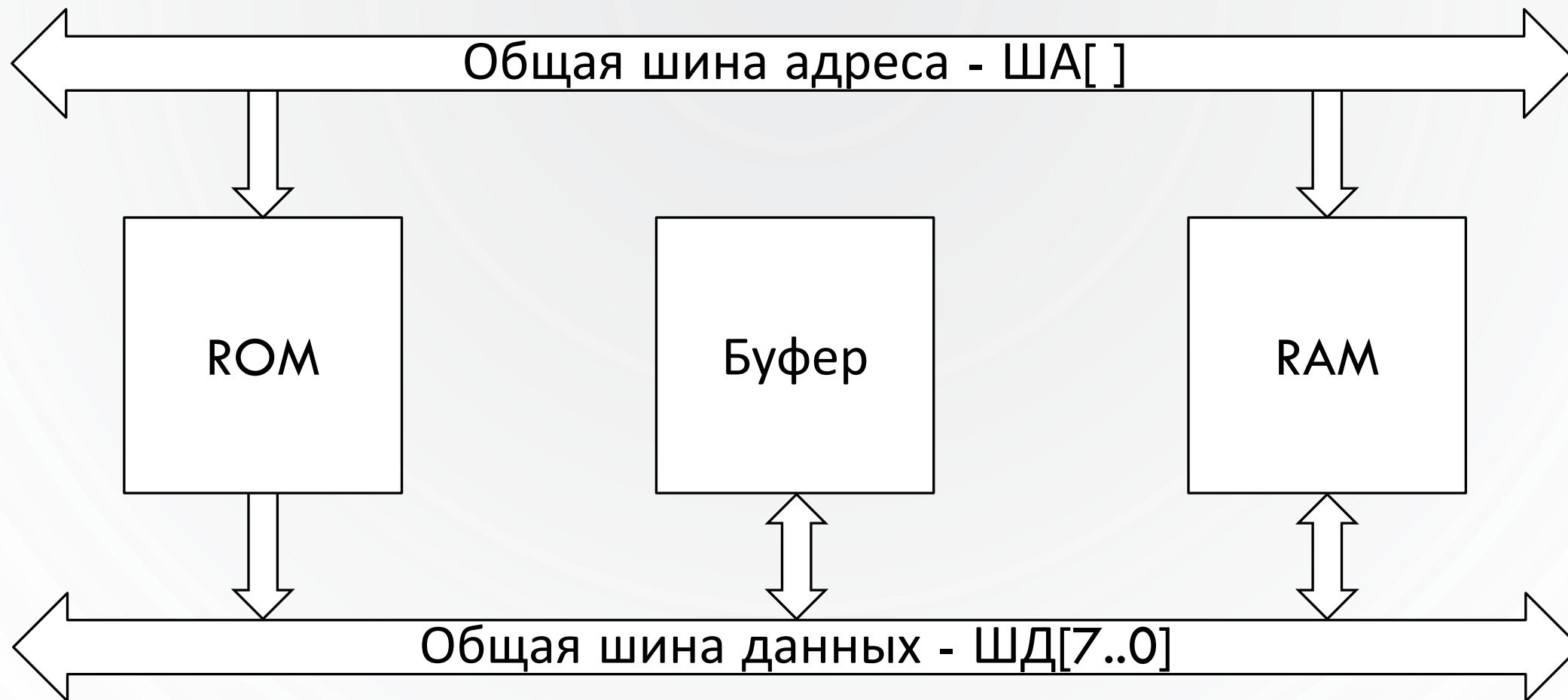
2024

The background features a light gray gradient with faint, concentric circular patterns. In the corners, there are decorative black line art elements resembling circuit traces or stylized trees, with small circles at the end of the lines.

ИСПОЛЬЗОВАНИЕ МОДУЛЕЙ ПАМЯТИ

SIFO LAB PRACTICUM PART 1 – Л.Р. 3

СТРУКТУРНАЯ СХЕМА



ЗАДАНИЕ

Разработать схему, включающую в себя буфер данных, модули ROM и RAM, подключенные к общей шине данных.

Прочитать N последовательных байт из ROM\RAM в буфер (заполнить буфер полностью). После ожидания M тактов clk этот блок данных передается из буфера в RAM. Повторить данные действия для другого источника памяти.

- RAM -> буфер (N байт) - (подождать M тактов clk) -> RAM
- ROM -> буфер (N байт) - (подождать M тактов clk) -> RAM

СИГНАЛЫ В СХЕМЕ

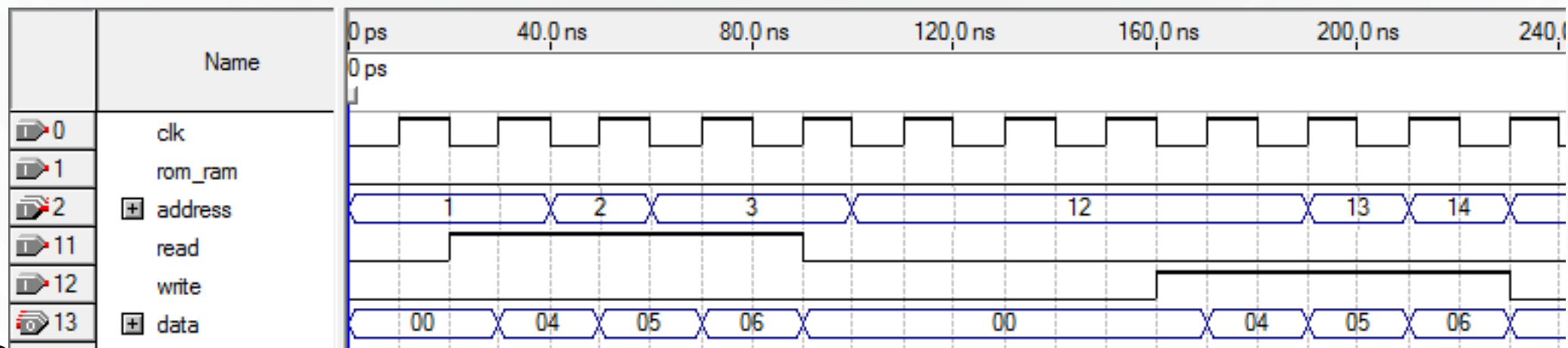
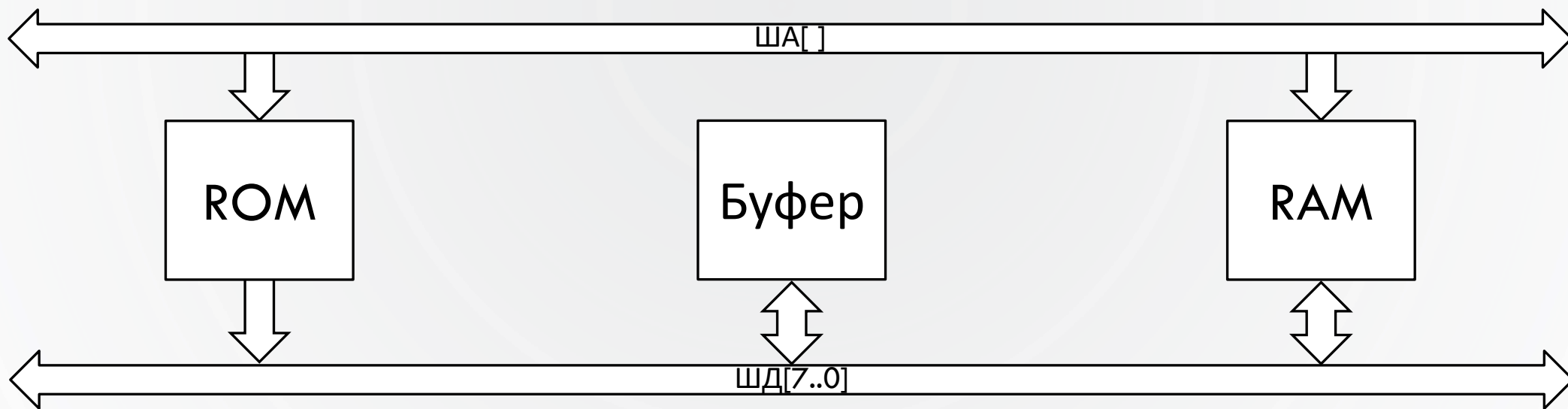
ВХОДНЫЕ

- clk
- Address[] – Общая ША
- ROM_RAM – выбор источника
- Read – чтение из памяти
- Write – запись в RAM

ВЫХОДНЫЕ

- Data[7..0] – Общая ШД

ПРИМЕР



ПРИМЕР

ram.hex

| Addr | +0 | +1 | +2 | +3 | +4 | +5 | +6 | +7 |
|------|----|----|----|----|----|----|----|----|
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 08 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 10 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 18 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 20 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 28 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 30 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 38 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 48 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 50 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 58 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 60 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 68 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 70 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 78 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |

Simulation Report - |lab3|lpm_ram_io:inst1|altr...

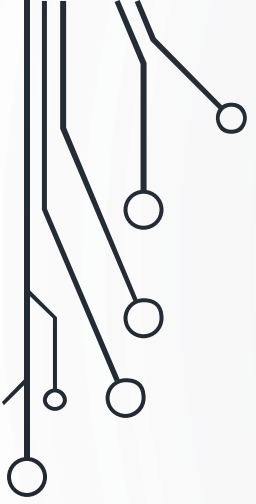
| |
|-----------------------|
| Simulation Report |
| Legal Notice |
| Flow Summary |
| Flow Settings |
| Simulator |
| Summary |
| Settings |
| Simulation Waveforms |
| Logical Memories |
| lab3 lpm_rom0:inst1 |
| lab3 lpm_ram_io:inst1 |
| Simulation Coverage |
| INI Usage |
| Messages |

| Addr | +0 | +1 | +2 | +3 | +4 | +5 | +6 | +7 |
|------|----|----|----|----|----|----|----|----|
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 08 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 10 | 00 | 00 | 04 | 05 | 06 | 00 | 00 | 00 |
| 18 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 20 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 28 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 30 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 38 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 48 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 50 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 58 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 60 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 68 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 70 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 78 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |

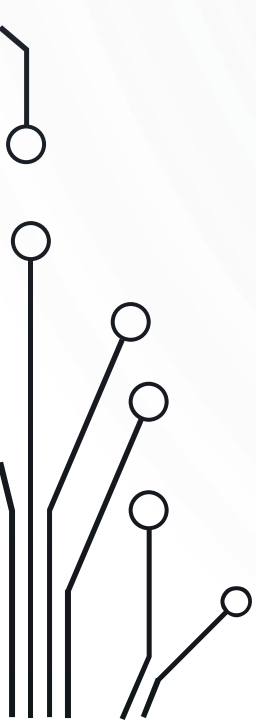
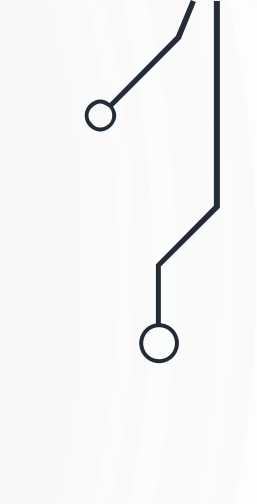
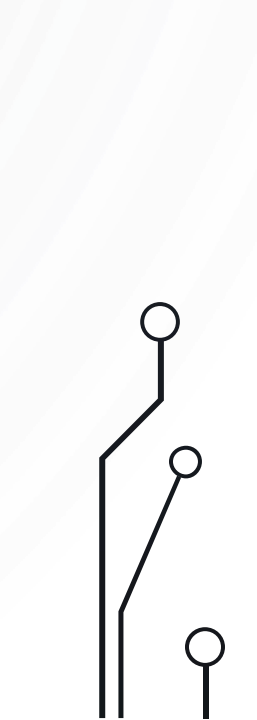
- Буфер делать только через регистры (например `lpm_dff`)
- Вход clock на ROM и RAM обязательно разделять на inclock и outclock
- Кол-во тактов чтения\записи неограничено
- Чтение не должно начинаться с 0 адреса
- Запись не должна начинаться с 0 адреса
- Прочитанные данные из ROM и RAM должны отличаться между собой
- Вид шин (ША, ШД) на моделировании и дампы памяти привести к одному виду (DEC или HEX)

Л.Р. НЕ ПРИНИМАЕТСЯ

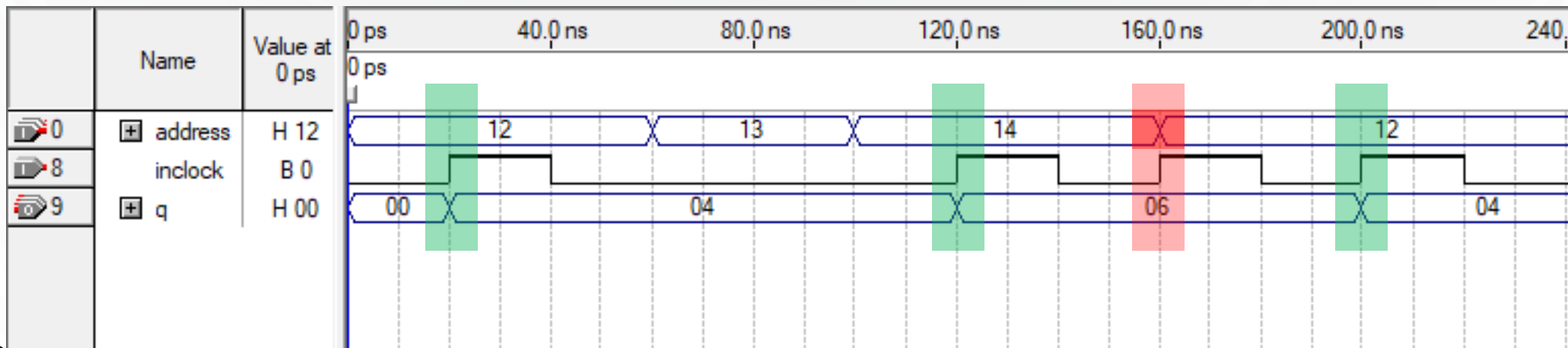
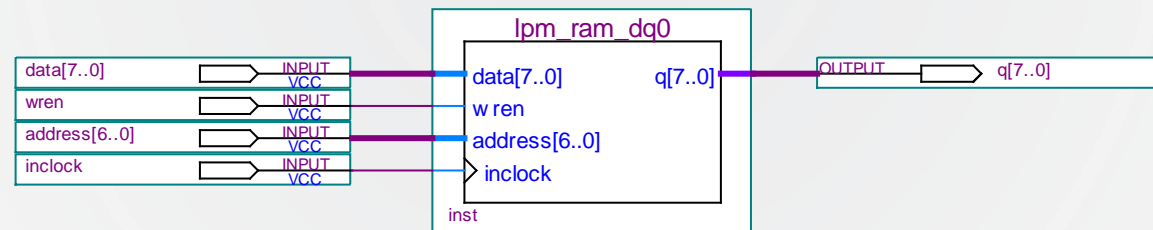
- Работа схемы начата с 0ns
- Добавлены другие входные сигналы
- Ожидание M тактов clk сделано путем последовательного добавления регистров в буфере
- Не привязан файл инициализации памяти
- Чтение данных из ROM\RAM происходит до появления сигнала Read
- Запись данных в RAM происходит до появления сигнала Write
- Данные на шину поступают через мультиплексор



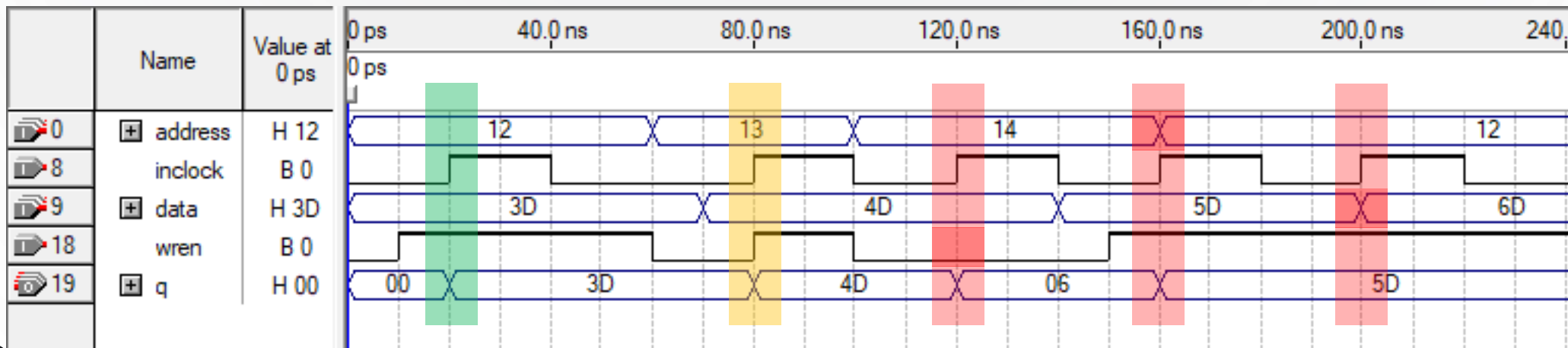
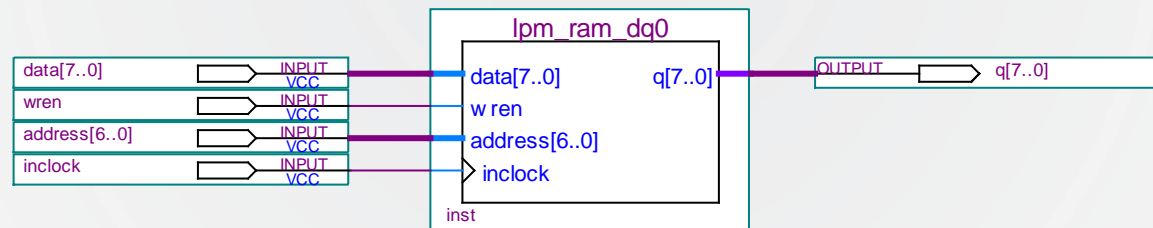
БАЛЛЫ ЗА Л.Р.

- 4-8 – за выполнение.
 - 1 – за скорость сдачи.
- 
- 
- 

ЧТЕНИЕ



ЗАПИСЬ



СИНХРОННЫЙ \ АСИНХРОННЫЙ ВЫВОД

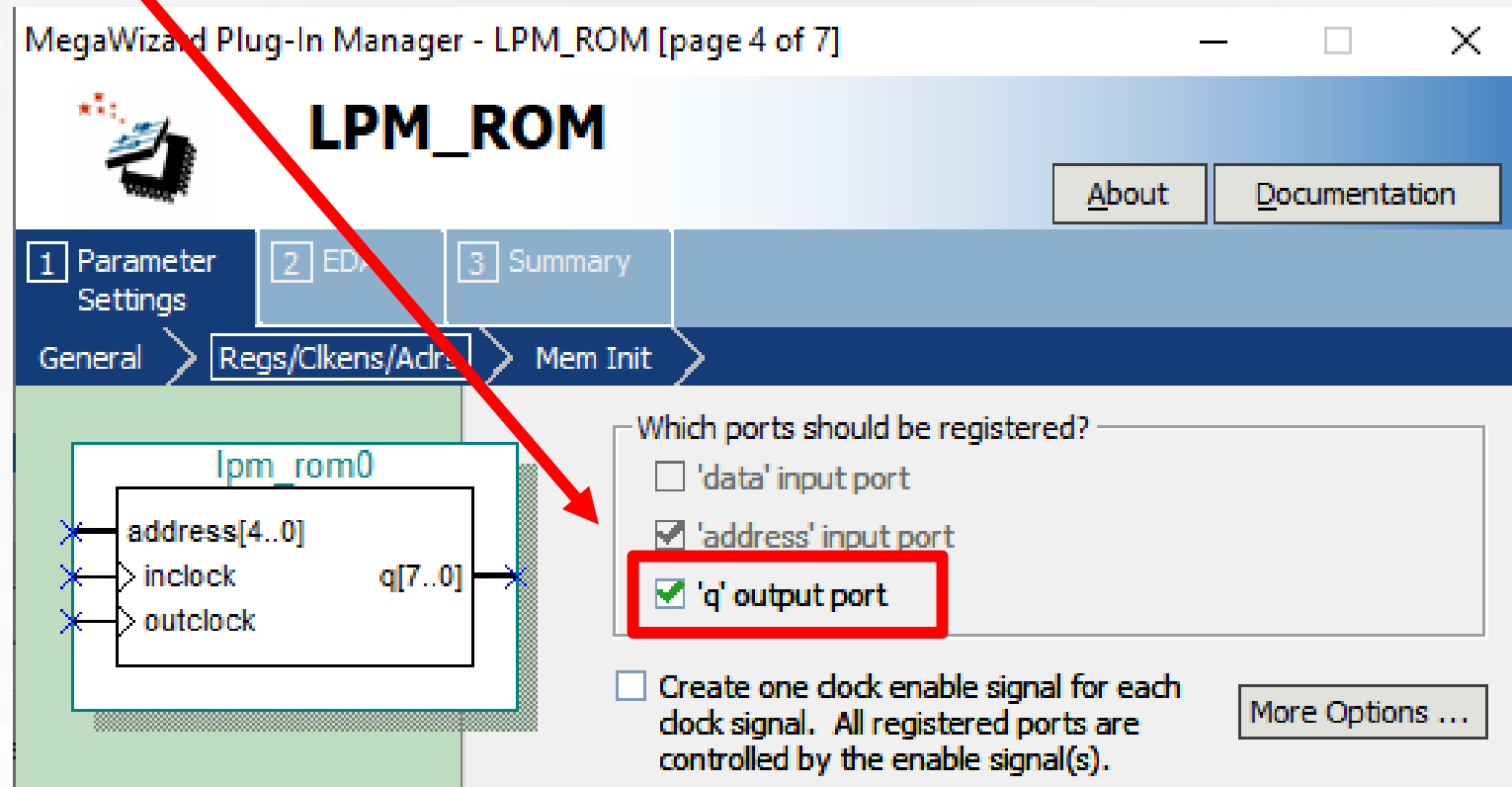
- Синхронный \ асинхронный вывод памяти (ROM или RAM) заключается в наличии \ отсутствии входа `outclock` у соответствующего модуля

Table 3–1. lpm_rom Input Ports (User Guide)

| Name | Required | Description | Comment |
|-----------|----------|-----------------------------|---|
| address[] | Yes | Address input to the memory | Input port LPM_WIDTHHAD wide. |
| inclock | No | Clock for input registers. | The address[] port is synchronous (registered) when inclock port is connected, and is asynchronous (unregistered) when the inclock port is not connected. |
| outclock | No | Clock for output registers. | Addressed memory content-to-q[] response is synchronous when outclock port is connected, and is asynchronous when it is not connected. |
| memenab | No | Memory enable input. | High = data output on q[], Low = high-impedance outputs. |

СИНХРОННЫЙ \ АСИНХРОННЫЙ ВЫВОД

- Синхронный \ асинхронный вывод памяти (ROM или RAM) заключается в наличии \ отсутствии входа outclock у соответствующего модуля



ВЫБОР СИНХРОНИЗАЦИИ ПАМЯТИ

MegaWizard Plug-In Manager - LPM_ROM [page 3 of 7]

LPM_ROM

About Documentation

1 Parameter Settings 2 EDA 3 Summary

General Regs/Clkens/Acds Mem Init

lpm_rom0

address[7..0] q[7..0]

clock

Currently selected device family: Stratix II

☒ Match project/default

Family supports LPM_ROM only in backward-compatibility mode. Altera recommends using ALTSYNCRAM wizard.

How wide should the 'q' output bus be? 8 bits

How many 8-bit words of memory? 256 words

What should the memory block type be?

☒ Auto ☐ M512 ☐ M4K

☐ M-RAM ☐ LCs Options...

Set the maximum block depth to Auto words

What clocking method would you like to use?

☒ Single clock

☐ Dual clock: use separate 'input' and 'output' clocks

Resource Usage

2048 ram_bits (AUTO)

Cancel < Back Next > Finish

MegaWizard Plug-In Manager - LPM_ROM [page 3 of 7]

LPM_ROM

About Documentation

1 Parameter Settings 2 EDA 3 Summary

General Regs/Clkens/Acds Mem Init

lpm_rom0

address[7..0] q[7..0]

inclock outclock

Currently selected device family: Stratix II

☒ Match project/default

Family supports LPM_ROM only in backward-compatibility mode. Altera recommends using ALTSYNCRAM wizard.

How wide should the 'q' output bus be? 8 bits

How many 8-bit words of memory? 256 words

What should the memory block type be?

☒ Auto ☐ M512 ☐ M4K

☐ M-RAM ☐ LCs Options...

Set the maximum block depth to Auto words

What clocking method would you like to use?

☐ Single clock

☒ Dual clock: use separate 'input' and 'output' clocks

Resource Usage

2048 ram_bits (AUTO)

Cancel < Back Next > Finish

ВЫБОР синхронизации ПАМЯТИ

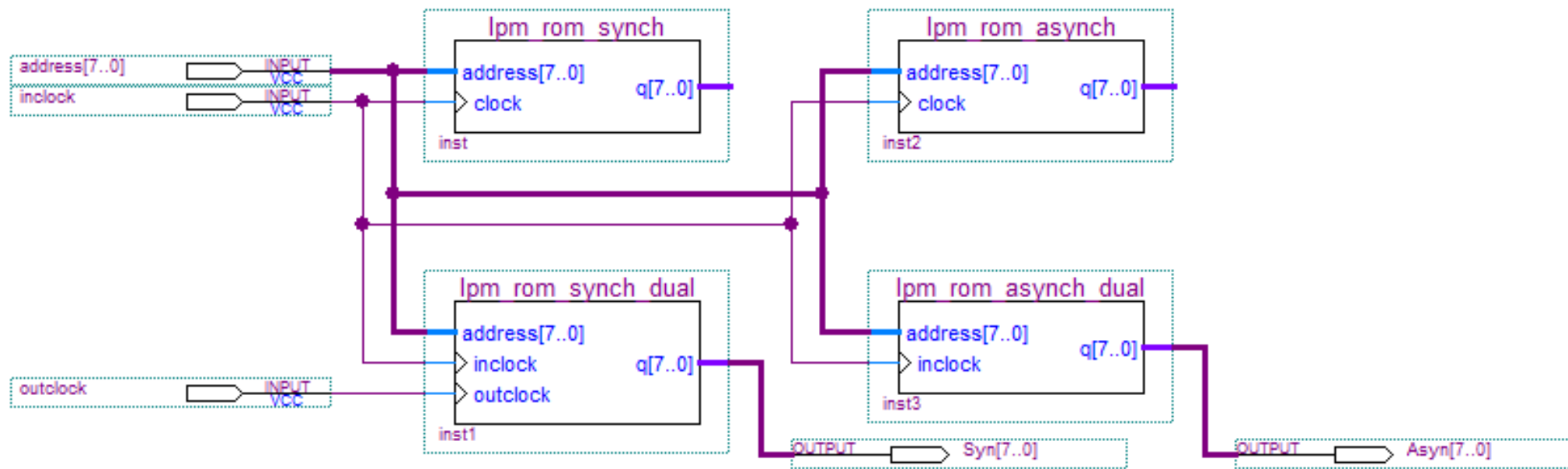
синхронный вывод

- Без разделения данные будут получены на 2 такте
- Раздельная синхронизация позволяет более гибко построить схему

асинхронный вывод

- В любом случае данные будут получены на 1 такте

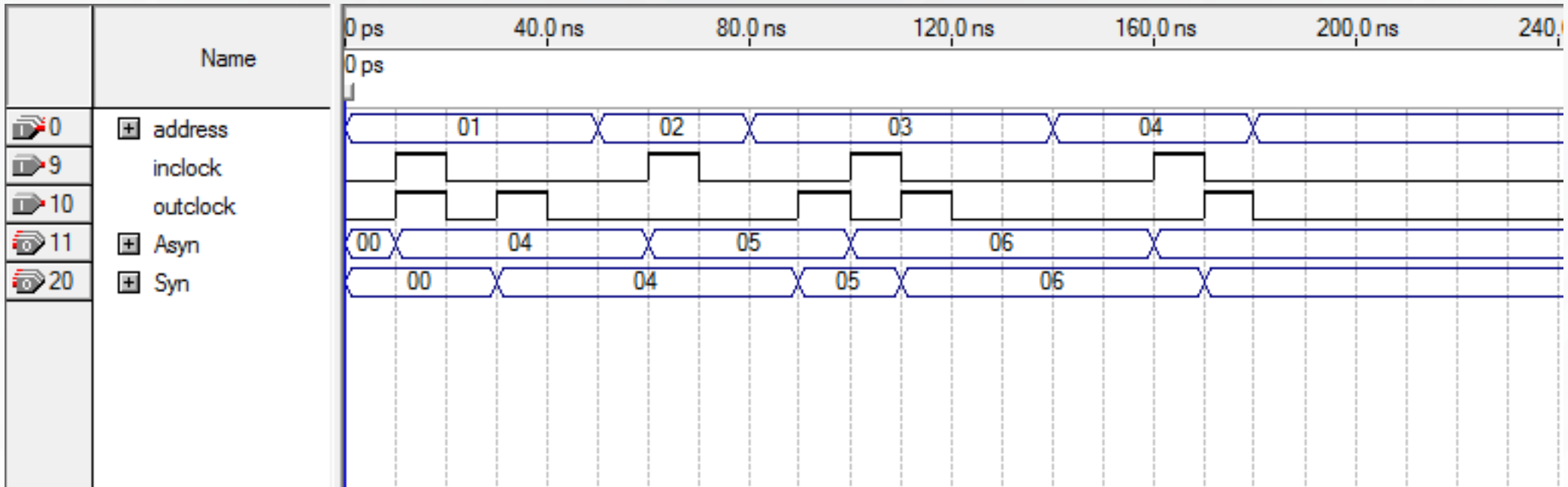
СИНХРОННЫЙ\АСИНХРОННЫЙ ВЫВОД



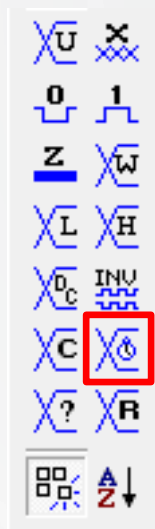
Синхронный вывод

Асинхронный вывод

СИНХРОННЫЙ\АСИНХРОННЫЙ ВЫВОД



НАСТРОЙКА СИНХРОСИГНАЛА



Base waveform on

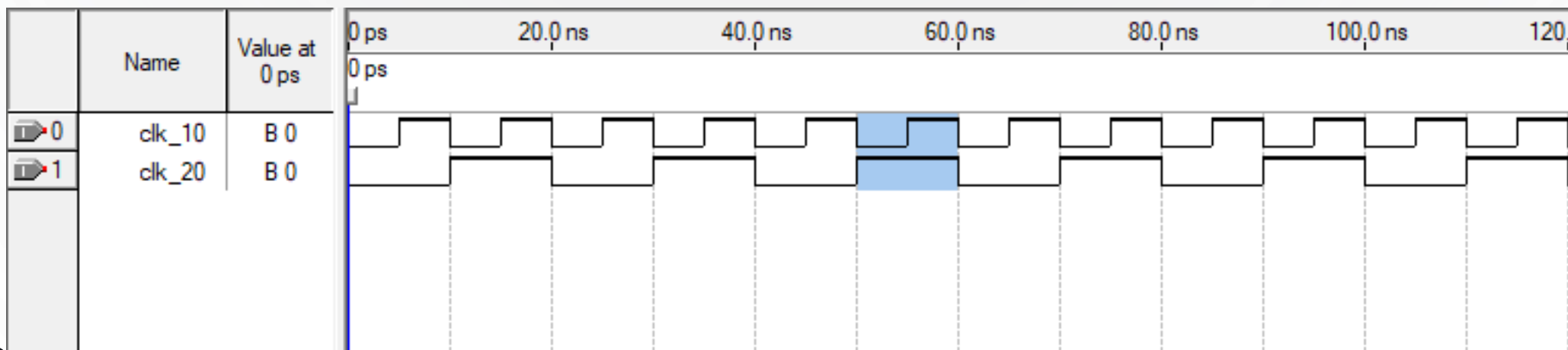
☐ Clock settings:

☒ Time period:

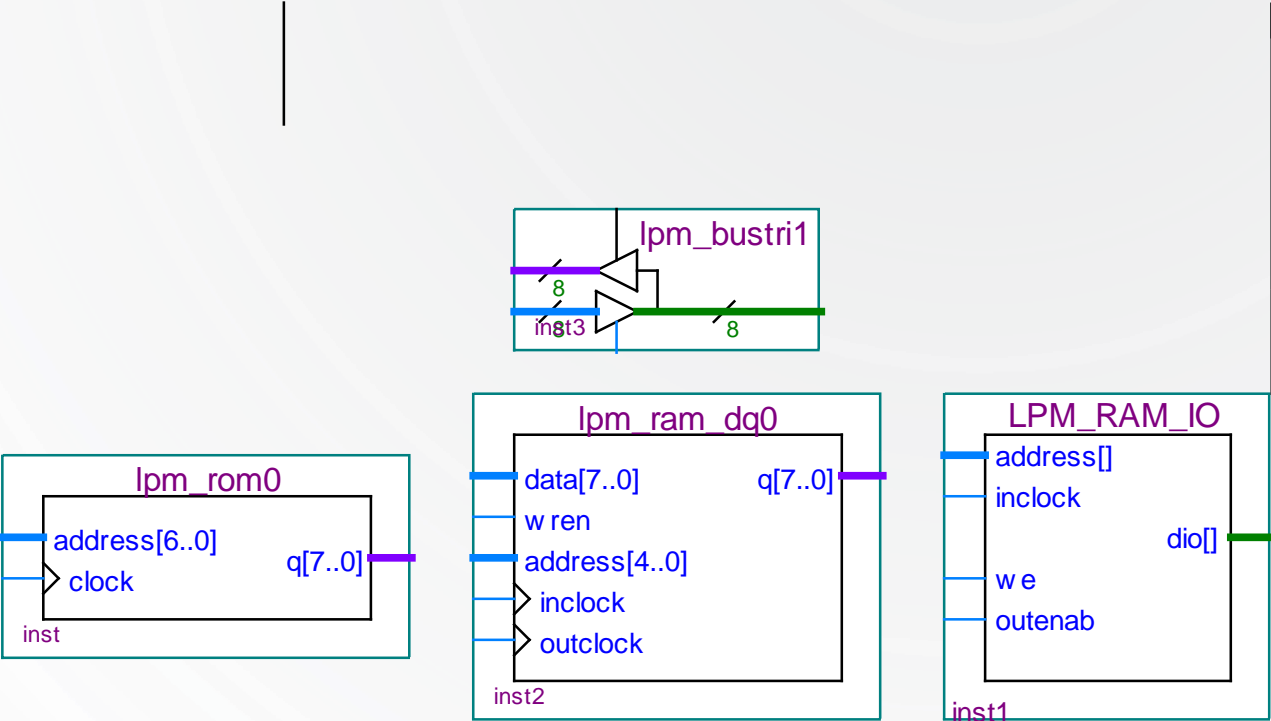
Period: 20.0 ns

Offset: 0.0 ns

Duty cycle (%): 50



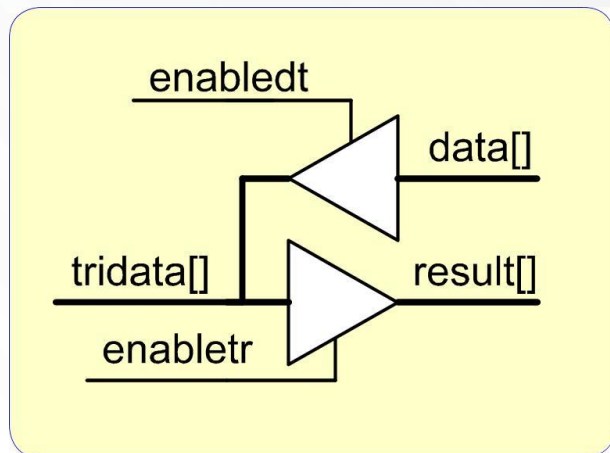
НЕОБХОДИМЫЕ ЭЛЕМЕНТЫ



| Parameter | Value | Type |
|---------------------|--------------|--------|
| LPM_ADDRESS_CONTROL | REGISTERED | String |
| LPM_FILE | ram.hex | |
| LPM_INDATA | REGISTERED | String |
| LPM_NUMWORDS | 128 | |
| LPM_OUTDATA | UNREGISTERED | String |
| LPM_WIDTH | 8 | |
| LPM_WIDTHAD | 7 | |

LPM_BUSTRI

- Двухнаправленный буфер с тремя состояниями
- Примитивы TRI, которые управляют портами OUTPUT или BIDIR, имеют вход разрешения выхода (Output Enable), который переводит выход в высокоимпедансное состояние



| № | Inputs | | Bidirectional tridata[LPM_WIDTH-1..0] | Output result[LPM_WIDTH-1..0] |
|---|----------|----------|--|----------------------------------|
| | enabledt | enabletr | | |
| 0 | 0 | 0 | Z (input) | Z |
| 1 | 0 | 1 | Z (input) | tridata[LPM_WIDTH-1..0] |
| 2 | 1 | 0 | data[LPM_WIDTH-1..0] | Z |
| 3 | 1 | 1 | data[LPM_WIDTH-1..0] | data[LPM_WIDTH-1..0] |

Master Time Bar: 70.0 ns Pointer: 0 ps Interval: -70.0 ns Start:

| Name | 0 ps | 20.0 ns | 40.0 ns | 60.0 ns | 80.0 ns | 100.0 ns | 120.0 ns | 140.0 ns | | | | | | | | | |
|------------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|--|--|
| 0 Reg_ascii | [0] | [1] | [2] | [9] | [10] | [11] | [31] | [] | ! | " | 2 | 3 | = | K | L | | |
| 9 Reg_decimal | 0 | 1 | 2 | 9 | 10 | 11 | 31 | 32 | 33 | 34 | 50 | 51 | 61 | 75 | 76 | | |
| 18 Reg_hex | 00 | 01 | 02 | 09 | 0A | 0B | 1F | 20 | 21 | 22 | 32 | 33 | 3D | 4B | 4C | | |
| 27 Reg_binary | 00000000 | 00000001 | 00000010 | 00010001 | 00010100 | 00010111 | 00111111 | 01000000 | 01000001 | 01000010 | 01100010 | 01100011 | 01111011 | 10010111 | 10011111 | | |
| 28 Reg_binary[6] | | | | | | | | | | | | | | | | | |
| 29 Reg_binary[5] | | | | | | | | | | | | | | | | | |
| 30 Reg_binary[4] | | | | | | | | | | | | | | | | | |
| 31 Reg_binary[3] | | | | | | | | | | | | | | | | | |
| 32 Reg_binary[2] | | | | | | | | | | | | | | | | | |
| 33 Reg_binary[1] | | | | | | | | | | | | | | | | | |
| 34 Reg_binary[0] | | | | | | | | | | | | | | | | | |

QUARTUS

- Увеличить время моделирования: Edit -> End Time...
- Кнопка “Generate Functional Simulation Netlist” нажимается 1 раз после каждой компиляции
- Создание символа: SIFO lab practicum part 1 (стр 16)
- Схему в виде блока пересохранять нужно только при изменении пинов

Горячие клавиши:

- Ctrl + Space - увеличить масштаб
- Ctrl + Shift + Space - уменьшить масштаб
- Ctrl + Alt + Space - полноэкранный режим