



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

**"МИРЭА - Российский технологический университет"**

**РТУ МИРЭА**

---

**Институт Информационных Технологий**

**Кафедра Вычислительной Техники**

**Лабораторная работа №8**

**по дисциплине**

**«Архитектура ВМиС»**

Студент группы: ИКБО-04-20

Хан А.А.  
(Фамилия студента)

Преподаватель

Железняк Л.М.  
(Фамилия преподавателя)

Москва 2021

## Содержание

ВВЕДЕНИЕ .....	2
Цель лабораторной работы .....	3
Порядок выполнения работы .....	3
Выполнение работы .....	4
Таблица истинности .....	6
Контрольные вопросы .....	8
ВЫВОДЫ .....	20
СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ .....	21

## **ВВЕДЕНИЕ**

Нам необходимо смоделировать логическую схему 3-х разрядного мультиплексора 8:1 с использованием параметрических элементов. Исследовать работу схемы с использованием сигнального редактора.

## **Цель лабораторной работы**

Приобретение навыков использования параметрических элементов (LPM function) в САПР QUARTUS II, экспериментальное исследование счетчиков и регистров, построенных на их основе.

## **Порядок выполнения работы**

1. Изучить правила построения и принцип работы триггеров и построение на их основе логических схем.
2. Нарисовать электрическую схему по указанию преподавателя при помощи графического редактора САПР QUARTUS II. и произвести симуляцию работы схемы, зарисовать диаграммы работы и по ее результатам заполнить таблицу истинности смоделированной схемы.
3. Спроектировать эту же электрическую схему, но с использованием параметрических элементов САПР QUARTUS II, проверить ее работу в сигнальном редакторе и оценить временные задержки в схеме.
4. Спроектировать эту же электрическую схему, но и использованием готовых элементов из библиотеки примитивов САПР QUARTUS II, проверить ее работу в сигнальном редакторе и по ее результатам заполнить таблицу истинности смоделированной схемы.
5. Ответить на контрольные вопросы, оформить отчет о выполненной работе.

## Выполнение работы

**Задание варианта:** 3-х разрядный мультиплексор 8:1.

Строение мультиплексора содержит 8 информационных входов D0-D7, 3 входа выбора S0-S2 и один выход F.

Спроектировать электрическую схему, с использованием готовых элементов из библиотеки примитивов САПР QUARTUS II, проверить ее работу в сигнальном редакторе и по ее результатам заполнить таблицу истинности смоделированной схемы.

Принципиальная схема с одним мультиплексором 8к1.

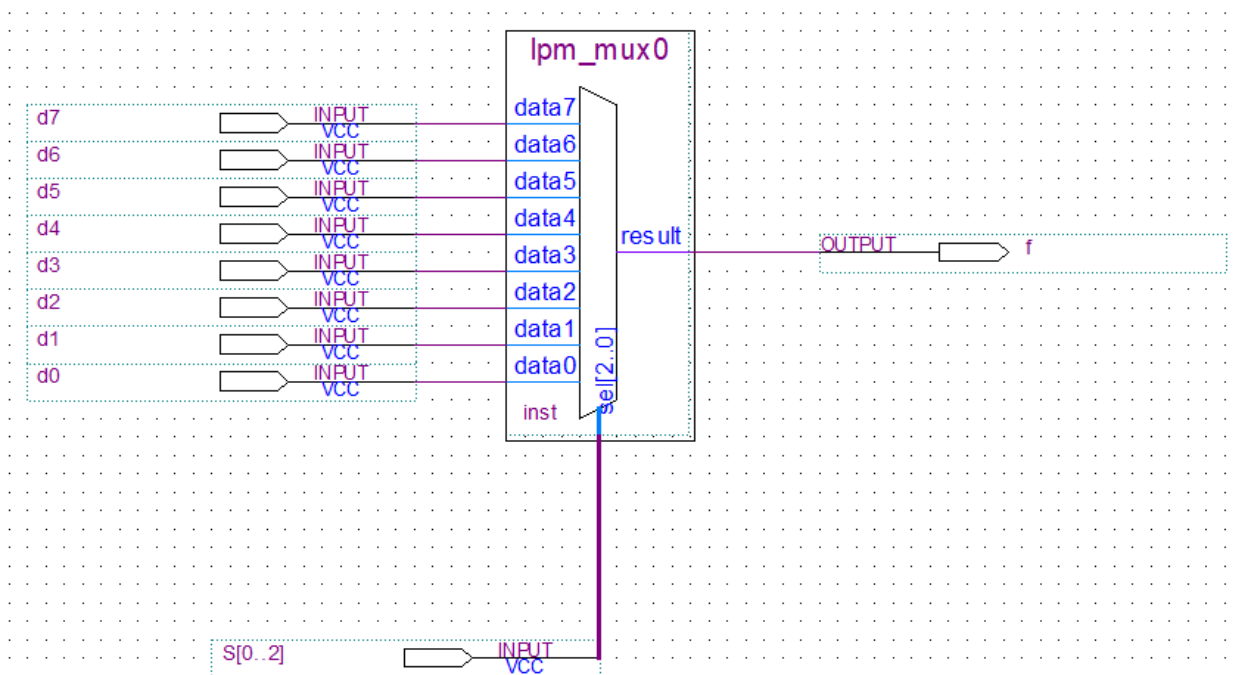


Рис. 1. – Синтезированная схема.

Диаграмма результата работы схемы.

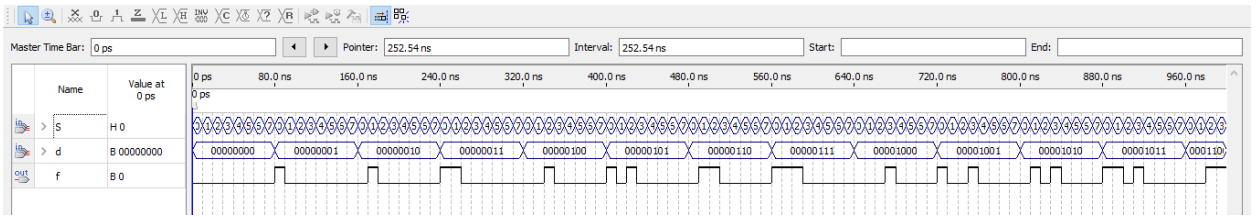


Рис. 2. – Результаты моделирования работы схемы в сигнальном редакторе.

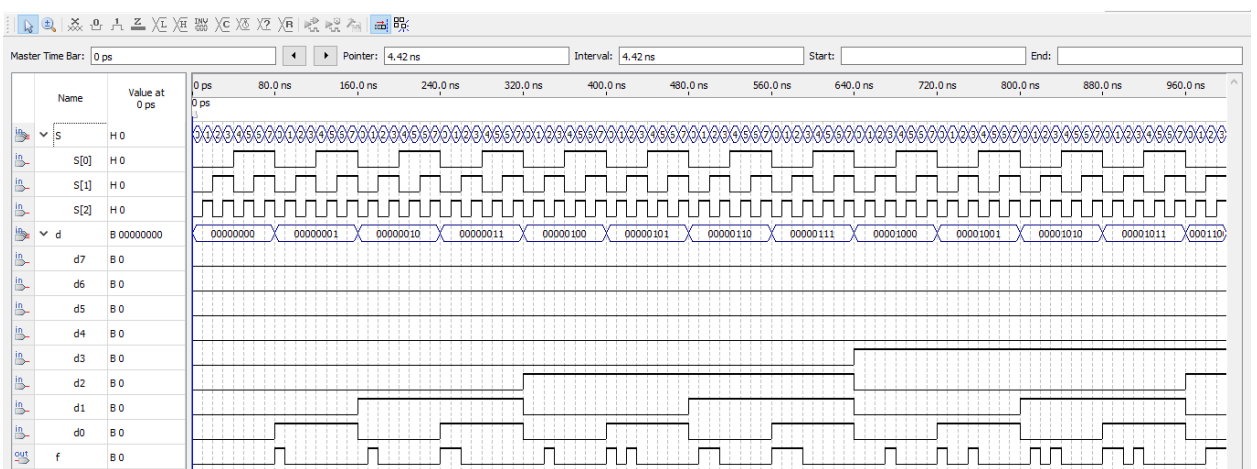


Рис. 3. - Результаты моделирования работы схемы в сигнальном редакторе.

Диаграмма результата работы оказался идентичной двум первым диаграммам, следовательно, обе схемы и программа составлены правильно.

## Таблица истинности

S2S1S0	D7D6D5D4D3D2D1D0	F
000	0000001	1
001		0
010		0
011		0
100		0
101		0
110		0
111		0
000	0000010	0
001		1
010		0
011		0
100		0
101		0
110		0
111		0
000	0000100	0
001		0
010		1
011		0
100		0
101		0
110		0
111		0
000	0001000	0
001		0
010		0
011		1
100		0
101		0
110		0
111		0

000	0010000	0
001		0
010		0
011		0
100		0
101		1
110		0
111		0
000	0100000	0
001		0
010		0
011		0
100		0
101		0
110		1
111		0
000	1000000	0
001		0
010		0
011		0
100		0
101		0
110		0
111		1



## Контрольные вопросы

1. Объясните понятие «параметрический элемент». Какие параметрические элементы доступны в САПР QUARTUS II?

**Параметрический элемент** — это один или несколько конструктивных элементов, которые можно сохранять и повторно использовать в других проектах.

### *Параметрические элементы САПР QUARTUSII*

#### **Counter**

<i>Входные выводы</i>	
Имя вывода	Описание
data [ ]	Параллельный вход данных счетчика
clock	Вход счетных импульсов
clk_en	Разрешение синхронизации.
cnt_en	Разрешение счета
updown	Управление направлением счета (1 = сложение, 0 = вычитание)
aclr	Асинхронный сброс входов
aset	Асинхронная установка входов
aload	Асинхронная загрузка входов. Установка счетчика в значение data[ ].
sclr	Синхронный сброс входов. Сброс счетчика следующим тактовым импульсом
sset	Синхронная установка входов. Установка счета следующим тактовым импульсом.

sload	Синхронная загрузка входов. Загрузка в счетчик значения data[ ] следующим тактовым импульсом.
-------	---

<i>Выходные выводы</i>	
Имя вывода	Описание
q [ ]	Выход счетчика
eq [15..0]	Декодированный выход счетчика. Высокий активный уровень появляется в момент, когда счетчик достигает заданного значения.
cout	Перенос в старший разряд
<i>Параметры</i>	
Параметр	Описание
LPM_WIDTH	Разрядность счетчика или входных значений data[ ] и выходных q[ ].
LPM_DIRECTION	Может принимать значения “UP”, “DOWN” или “UNUSED”. Если этот параметр используется, то вход updown не должен быть подключен. Если вход updown не подключен, то значение LPM_DIRECTION по умолчанию – “UP”
LPM_MODULUS	Максимальный счет, плюс один. Число уникальных состояний в цикле счетчика. Если введенное значение больше, чем LPM_MODULUS параметр, поведение счетчика не определено.
LPM_AVALUE	Постоянное значение, которое загружается, когда aset высок. Если введенное значение больше чем <modulus>, поведение счетчика - неопределенный (X) логический уровень, где

	<modulus> - LPM_MODULUS. Параметр ограничен значением в 32 бита.
LPM_SVALUE	Постоянное значение, которое загружается по переднему фронту тактовых импульсов, когда sset или sconst высок. Должен использоваться, если sconst используется.
LPM_HINT	Позволяет определять специфические Altera-параметры в файлах проекта VHDL.
LPM_TYPE	Идентифицирует LPM имя файлах проекта VHDL

### Multiplier

Входные выводы	
Имя вывода	Описание
dataa[]	Множимое
datab[]	Множитель
sum[]	Частичная сумма
clock	Вход тактовых импульсов
clken	Разрешение использования тактового входа
aclr	Асинхронный сброс
Выходные выводы	
Имя вывода	Описание
result[]	result = dataa [] * datab [] + sum. The product LSB is aligned with the sum LSB.

Параметры	
Параметр	Описание

LPM_WIDTHA	Разрядность dataa[ ].
LPM_WIDTHB	Разрядность datab[ ].
LPM_WIDTHP	Разрядность result[ ].
LPM_WIDTHS	Разрядность sum [ ]. Обязателен, даже если порт суммы не используется.
LPM_REPRESENTATION	Тип выполняемого сравнения “SIGNED”, “UNSIGNED”, “UNUSED”. Если значение не указано, то по умолчанию устанавливается “UNSIGNED”
LPM_HINT	Позволяет определять специфические Altera-параметры в файлах проекта VHDL.
LPM_TYPE	Идентифицирует LPM имя файлах проекта VHDL
INPUT_A_IS_CONSTANT	Altera параметр. Принимает значения "YES", "NO", и "UNUSED". Если dataa [ ] связан с постоянным значением, устанавливая INPUT_A_IS_CONSTANT "YES" оптимизирует <i>multiplier</i> по использованию ресурсов и скорости. Если опущено, значение по умолчанию - "NO".
INPUT_B_IS_CONSTANT	Altera параметр. Принимает значения "YES", "NO", и "UNUSED". Если datab [ ] связан с постоянным значением, устанавливая INPUT_B_IS_CONSTANT "YES" оптимизирует <i>multiplier</i> по использованию ресурсов и скорости. Значение по умолчанию - "NO".

USE_EAB	Altera параметр. Принимает значения "ON", "OFF", и "UNUSED". Устанавливая параметр USE_EAB "ON" позволяет QUARTUSII использовать блоки дополнительных атрибутов, чтобы использовать 4 x 4 или (8 x значение константы) стандартные блоки в ACEX1K и FLEX10K устройствах.
LATENCY	Altera параметр. То же, что и LPM_PIPELINE. Параметр обеспечивает совместимости с QUARTUSII проектами версии ниже 7.0. Для всех новых проектов, используется параметр LPM_PIPELINE
MAXIMIZE_SPEED	Altera параметр. Возможные значения от 0 до 10. Если параметр используется, то QUARTUSII пытается оптимизировать данную функцию lpm_mult для скорости, а не для уменьшения занимаемой области, и отменяет установку опции Optimize в диалоговом окне Global Project Logic Synthesis (меню Assign). Если MAXIMIZE_SPEED не использован, значение опции Optimize используется вместо него. Если установлено MAXIMIZE_SPEED - 6 или выше, компилятор оптимизирует мегафункции lpm_mult для более высокой скорости; если установлено - 5 или меньше, компилятор оптимизирует для уменьшения занимаемой области.

### Comparator

Входные выводы	
Имя вывода	Описание

dataa[]	datab[] сравнивается с этим значением
datab[]	Значение с которым сравнивается dataa[]
clock	Вход тактовых импульсов
clken	Разрешение использования тактового входа
aclr	Асинхронный сброс
<i>Выходные выводы</i>	
Имя вывода	Описание
alb	“High” (1) если dataa[] < datab[]
aeb	“High” (1) если dataa[] == datab[]
agb	“High” (1) если dataa[] > datab[]
ageb	“High” (1) если dataa[] >= datab[]
aneb	“High” (1) если dataa[] != datab[]
aleb	“High” (1) если dataa[] <= datab[]

<i>Параметры</i>	
Параметр	Описание
LPM_WIDTH	Разрядность входов dataa[] и datab[]
LPM_REPRESENTATION	Тип выполняемого сравнения “SIGNED”, “UNSIGNED”, “UNUSED”. Если значение не указано, то по умолчанию устанавливается “UNSIGNED”
LPM_PIPELINE	
LPM_HINT	Позволяет определять специфические Altera-параметры в файлах проекта VHDL.
LPM_TYPE	Идентифицирует LPM имя файла проекта VHDL

CHAIN_SIZE	
ONE_INPUT_IS_CONSTANT	Специфический Altera - параметр. Принимает значения "YES", "NO", или "UNUSED". Обеспечивает большую оптимизацию, если один из входов постоянен. По умолчанию - "NO".

### Adder Subtractor

Входные выходы	
Имя вывода	Описание
dataa[]	Первое слагаемое/ Уменьшаемое
datab[]	Слагаемое/ Вычитаемое
add_sub	Если "1" (high), операция = dataa [] +datab [] +cin. Если "0" (low), операция = dataa[]- datab[] +cin-1
clock	Вход тактовых импульсов
clken	Разрешение использования тактового входа
aclr	Асинхронный сброс

Выходные выходы	
Имя вывода	Описание
result[]	dataa [] +datab [] +cin или dataa[] -datab[] +cin-1.
cout	Обнаруживает переполнения в операциях "UNSIGNED".
overflow	Результат превышает доступную точность

Параметры
-----------

Параметр	Описание
LPM_WIDTH	Разрядность входов dataa[], datab[], result[]
LPM_DIRECTION	Значения - "ADD", "SUB", и "UNUSED". Если не указано, значение по умолчанию "DEFAULT", в этом случае используется значение add_sub порта. Add_sub порт не может использоваться, если используется LPM_DIRECTION.
LPM_REPRESENTATION	Тип выполняемого сравнения “SIGNED”, “UNSIGNED”, “UNUSED”. Если значение не указано, то по умолчанию устанавливается “UNSIGNED”
LPM_HINT	Позволяет определять специфические Altera-параметры в файлах проекта VHDL.
LPM_TYPE	Идентифицирует LPM имя файла проекта VHDL
ONE_INPUT_IS_CONSTANT	Altera параметр. Принимает значения "YES", "NO", и "UNUSED". Обеспечивает большую оптимизацию, если один вход постоянный. Если не указано, значение по умолчанию - "NO"
MAXIMIZE_SPEED	Altera параметр. Возможные значения от 0 до 10. Если параметр используется, то QUARTUSII пытается оптимизировать данную функцию lpm_mult для скорости, а не для уменьшения занимаемой области, и отменяет установку опции Optimize в диалоговом окне Global Project Logic Synthesis (меню Assign). Если MAXIMIZE_SPEED не использован, значение опции Optimize используется вместо него. Если установлено MAXIMIZE_SPEED – 6 или выше,



	компилятор оптимизирует мегафункции <code>lpm_mult</code> для более высокой скорости; если установлено - 5 или меньше, компилятор оптимизирует для уменьшения занимаемой области.
--	---

### Absolute Value

Входные выводы	
Имя вывода	Описание
<code>data []</code>	Число со знаком

<i>Выходные выводы</i>	
Имя вывода	Описание
<code>result[]</code>	Абсолютное значение <code>data []</code> .
<code>overflow</code>	

<i>Параметры</i>	
Параметр	Описание
<code>LPM_WIDTHNA</code>	Разрядность <code>data []</code> и <code>result[]</code>
<code>LPM_HINT</code>	Позволяет определять специфические Altera-параметры в файлах проекта VHDL.
<code>LPM_TYPE</code>	Идентифицирует LPM имя файлах проекта VHDL

### Divider

Входные выводы	
Имя вывода	Имя вывода
numer[]	Числитель
denom[]	Знаменатель
clock	Вход тактовых импульсов
clken	Разрешение использования тактового входа
aclr	Асинхронный сброс
Выходные выводы	
Имя вывода	Описание
quotient[]	Частное
remain[]	Остаток

Параметры	
Параметр	Описание
LPM_WIDTHN	Разрядность numer[] и quotient[].
LPM_WIDTHD	Разрядность denom[] и remain[].
LPM_NREPRESENTATION	Определяет параметр числителя “SIGNED” или “UNSIGNED” Сейчас поддерживается только “UNSIGNED”.
LPM_DREPRESENTATION	Определяет параметр знаменателя “SIGNED” или “UNSIGNED” Сейчас поддерживается только “UNSIGNED”
LPM_HINT	Позволяет определять специфические Altera-параметры в файлах проекта VHDL.
LPM_TYPE	Идентифицирует LPM имя файлах проекта VHDL

2. Объясните принцип работы счетчика, построенного на триггерах. Какие типы счетчиков существуют?

**Счетчик** - это устройство, которое служит для отслеживания количества каких-либо событий.

**Счетчик** - это автомат, служащий для учета количества событий.

Счётчик на D-триггерах, 1 элемент меняет на противоположное значение, а на остальные последующие подаётся отрицание предыдущего D-триггера.

Счетчики классифицируются по следующим параметрам:

1. по разрядности
2.
  - суммирующие
  - вычитающие
  - реверсивные
  - с произвольным порядком пересчета
3.
  - синхронные
  - асинхронные
4. по типу формирования переноса внутри счетчика
  - с последовательным
  - с параллельным
  - с комбинированным
5.
  - с функцией установки произвольного числа
  - с установкой в ноль

Счетчик называют полным, если количество устойчивых состояний на выходе равно  $2^n$ , где n-число выходов счетчика

3. Объясните назначение пунктов меню Edit Ports/Parameters.

Для редактирования параметров и входов/выходов схемы необходимо Properties. Во вкладке Ports можно выбрать необходимые входы/выходы комбинационной схемы,

4. Чем ограничивается максимальная скорость работы счетчика? Какова максимальная частота работы счетчика разработанного в ходе выполнения лабораторной работы?

В зависимости как реализовано соединение триггеров, последовательно или параллельно, параллельно — минимально, последовательно — максимально.

Я разработала параллельную схему, т.е. 1 такт нужен.

## **ВЫВОДЫ**

В данной лабораторной работе я приобрела навыки использования параметрических элементов (LPM function) в САПР QUARTUS II, экспериментально исследовал счетчики и регистры, построенных на их основе.

## СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ

1. Головков А., Пивоваров И., Кузнецов И. Компьютерное моделирование и проектирование радиоэлектронных средств. Учебник для вузов. Стандарт третьего поколения.- СПб.: 2015. – 208 с.
2. Соловьев В.В., Климович А. Логическое проектирование цифровых систем на основе программируемых логических интегральных схем. – М.: Горячая линия - Телеком, 20011. – 376 с.
3. Стешенко В. ПЛИС фирмы ALTERA: элементная база, система проектирования и языки описания аппаратуры - М.: Додека, 2010. – 576 с.
4. Антонов А.П. Язык описания цифровых устройств AlteraHDL: Практический курс. – М.: ИП «Радиософт», 2013. – 224 с.
5. Ефремов Н.В. Введение в систему автоматизированного проектирования Quartus II. Учебное пособие. – М.: ГОУ ВПО МГУЛ, 2011. – 147 с.