

ПРИЛОЖЕНИЕ 1

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ В.Ф. УТКИНА
Кафедра радиотехнических систем

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

по дисциплине (модулю)

Б1.В.ДВ.04.02 «Проектирование ЦУ на ПЛИС»

Направление подготовки

11.03.01 Радиотехника

ОПОП

«Аппаратно-программная инженерия радиолокационных и навигационных систем»

Квалификация (степень) выпускника – бакалавр

Формы обучения – очная

Рязань 2023

Оценочные материалы – это совокупность учебно-методических материалов (контрольных заданий, описаний форм и процедур), предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной профессиональной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям основной профессиональной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности общекультурных, общепрофессиональных и профессиональных компетенций, приобретаемых обучающимся в соответствии с этими требованиями.

Контроль знаний проводится в форме текущего контроля и промежуточной аттестации.

Текущий контроль успеваемости проводится с целью определения степени усвоения учебного материала, своевременного выявления и устранения недостатков в подготовке обучающихся и принятия необходимых мер по совершенствованию методики преподавания учебной дисциплины (модуля), организации работы обучающихся в ходе учебных занятий и оказания им индивидуальной помощи.

К контролю текущей успеваемости относятся проверка знаний, умений и навыков, приобретенных обучающимися в ходе выполнения индивидуальных заданий на практических занятиях и лабораторных работах. При оценивании результатов освоения практических занятий и лабораторных работ применяется шкала оценки «зачтено – не зачтено». Количество лабораторных и практических работ и их тематика определена рабочей программой дисциплины, утвержденной заведующим кафедрой.

Результат выполнения каждого индивидуального задания должен соответствовать всем критериям оценки в соответствии с компетенциями, установленными для заданного раздела дисциплины.

Промежуточный контроль по дисциплине осуществляется проведением теоретического зачета (очная форма обучения), экзамена (очная форма обучения, 4 года), теоретического зачета в весеннем семестре и экзамена в осеннем (заочная форма обучения, 5 лет).

Форма проведения теоретического зачета – выполнение тестового задания – решения задачи по разработке программы на языке VHDL для описания логики работы цифрового устройства.

Форма проведения экзамена – письменный ответ по утвержденным экзаменационным билетам, сформулированным с учетом содержания учебной дисциплины. В экзаменационный билет включается два теоретических вопроса и одна задача. После выполнения письменной работы обучаемого производится ее оценка преподавателем и, при необходимости, проводится теоретическая беседа с обучаемым для уточнения экзаменационной оценки.

Критерии оценивания уровня сформированности компетенций в процессе выполнения лабораторных работ и практических занятий:

- 41%-60% правильных ответов соответствует пороговому уровню сформированности компетенции на данном этапе ее формирования;
- 61%-80% правильных ответов соответствует продвинутому уровню сформированности компетенции на данном этапе ее формирования;
- 81%-100% правильных ответов соответствует эталонному уровню сформированности компетенции на данном этапе ее формирования.

Сформированность уровня компетенций не ниже порогового является основанием для допуска обучающегося к промежуточной аттестации по данной дисциплине.

Формой промежуточной аттестации по данной дисциплине для очной формы обучения является

для очной формы обучения – зачет, оцениваемый по принятой в ФГБОУ ВО «РГРТУ» системе: «зачтено» / «не зачтено»;

для заочной формы обучения (4 года) – экзамен, оцениваемый по принятой в ФГБОУ ВО «РГРТУ» четырехбалльной системе: «неудовлетворительно», «удовлетворительно», «хорошо» и «отлично».

для заочной формы обучения (5 лет) – зачет, оцениваемый по принятой в ФГБОУ ВО «РГРТУ» системе: «зачтено» / «не зачтено» и экзамен, оцениваемый по принятой в ФГБОУ ВО «РГРТУ» четырехбалльной системе: «неудовлетворительно», «удовлетворительно», «хорошо» и «отлично».

Критерии оценивания промежуточной аттестации представлены в таблице 1.

Таблица 1 - Критерии оценивания промежуточной аттестации (лабораторные и практические занятия, теоретический зачет, экзамен)

Шкала оценивания	Критерии оценивания
«зачтено»	студент должен: продемонстрировать общее знание изучаемого материала; знать основную рекомендуемую программой дисциплины учебную литературу; уметь строить ответ в соответствии со структурой излагаемого вопроса; показать общее владение понятийным аппаратом дисциплины; уметь устранить допущенные погрешности в ответе на теоретические вопросы и/или при выполнении практических заданий под руководством преподавателя, либо (при неправильном выполнении практического задания) по указанию преподавателя выполнить другие практические задания того же раздела дисциплины.

«не зачтено»	ставится в случае: незнания значительной части программного материала; не владения понятийным аппаратом дисциплины; существенных ошибок при изложении учебного материала; неумения строить ответ в соответствии со структурой излагаемого вопроса; неумения делать выводы по излагаемому материалу. Оценка «не зачтено» также ставится студентам, которые не выполнили и защитили лабораторные работы и практические занятия, предусмотренные рабочей программой.
«отлично»	«Отлично» заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка «отлично» выставляется студентам, усвоившим взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала.
«хорошо»	«Хорошо» заслуживает студент, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе. Как правило, оценка «хорошо» выставляется студентам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.
«удовлетворительно»	«Удовлетворительно» заслуживает студент, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой. Как правило, оценка «удовлетворительно» выставляется студентам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя.

«не удовлетворительно»	«Неудовлетворительно» выставляется студенту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.
--------------------------------------	---

Контрольные вопросы для защиты лабораторных работ

1. Классификация микросхем программируемой логики.
2. Стандартные, сложные ПЛИС и программируемые пользователем вентильные матрицы.
3. Базовые матричные кристаллы. Основные свойства микросхем программируемой логики.
4. Общие свойства ПЛИС.
5. CPLD – сложные программируемые логические устройства.
6. Схема программируемой матрицы соединений CPLD.
7. Схема передачи сигналов из матрицы соединений в функциональный блок.
8. Структура функционального блока CPLD. FPGA – программируемые пользователем вентильные матрицы.
9. Структура FPGA. Функциональные блоки FPGA.
10. ПЛИС типа «Система на кристалле».
11. Схема логической ячейки.
12. Реализация цифровых фрагментов в форме полужаказных кристаллов.
13. Проектирование микропроцессорных фрагментов систем.
14. Средства и методы отладки. JTAG – интерфейс.
15. Архитектура БИС, использующей метод граничного сканирования.
16. Транспортный механизм JTAG-интерфейса.
17. Режимы, обеспечиваемые методом граничного сканирования.
18. Структура сканирующей ячейки. Основные команды граничного сканирования.
19. Программная поддержка тестирования. Системные функции на основе JTAG-интерфейса.
20. Языки описания дискретных устройств. Основные элементы языка VHDL. HDL-программа - модель проектируемого устройства.
21. Принципы интерпретации поведения дискретных устройств средствами моделирования.
22. Язык VHDL как программная система. Алфавит моделирования
23. Основные элементы и конструкции языка. Идентификаторы.

24. Ключевые слова. Литералы. Операторы. Разделители. Комментарии.
25. Форма задания синтаксических конструкций языка VHDL.
26. Типы. Классификация типов. Основные типы данных языка VHDL.
27. Декларации. Константы. Сигналы. Переменные. Дельта-задержка.
28. Операции языка VHDL. Процессы.
29. Интерфейс и архитектура объекта.
30. Атрибуты.
31. Последовательные операторы. Оператор ожидания события wait. Оператор if. Оператор case. Оператор null.
32. Оператор цикла.
33. Оператор next . Оператор exit
34. Процедуры и функции
35. Параллельные операторы. Оператор process, оператор параллельного сообщения, оператор параллельного вызова процедуры, оператор условного назначения сигнала, оператор select выборочного назначения процедуры, оператор конкретизации (создания экземпляра) компонента, параллельный оператор генерации generate, оператор block (блок).
36. Стили программирования на VHDL. Поведенческий стиль, стиль потоков данных, структурный стиль.
37. Структура проекта на VHDL. Синтаксис описания library.
38. Пакет std_logic_1164, разрешающая функция.
39. Отладка VHDL-описаний.
40. Синтезируемое подмножество языка VHDL.
41. Схемная реализация логического выражения для типа bit.
42. Пример схемной реализации логических выражений типа bit_vector
43. Схемная реализация арифметических операторов, схемная реализация операторов управления.

Контрольные вопросы для оценки сформированности компетенций

1. Разрешенные идентификаторы в VHDL.
2. Тип std_logic.
3. Объявление сущности в VHDL.
4. Какие элементы объявляются в декларативной части архитектуры программы?
5. Какие элементы объявляются в декларативной части процесса?
6. Можно ли объявить внутри процесса сигнал? Почему?
7. Можно ли объявить внутри процесса другой процесс? Почему?
8. Поименованное и позиционное подключение портов компонента.
9. Сигналы в VHDL.
10. Процессы в VHDL.
11. Синхронные и асинхронные действия.
12. Описание триггеров и регистров на языке VHDL.
13. Настраиваемые константы в VHDL.

14.Пакеты в VHDL.

15.Записи в VHDL.

16.Цифровые автоматы Мура и Мили.

**Примеры задач для практических занятий и теоретического зачета
(очная форма обучения)**

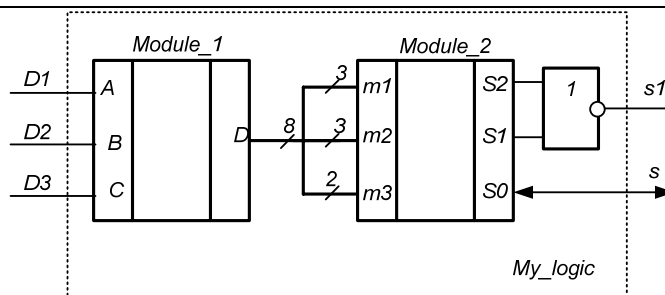
1. Составить описание на языке VHDL генератора M-последовательности с порождающим полиномом $P(x) = 1 + x^2 + x^5$ с входом разрешения асинхронной параллельной загрузки <i>Load</i> (активное значение – высокий логический уровень) стартовой комбинации <i>Start_data</i> .
2. Составить описание на языке VHDL параллельного N-разрядного регистра с асинхронным входом разрешения тактовых импульсов <i>ena</i> (активное значение – высокий логический уровень). Параметр <i>N</i> определить как глобальную константу, которая задается в пакете. Обеспечить работу регистра по заднему фронту тактового сигнала.
3. Составить описание на языке VHDL ШИМ-модулятора, позволяющего представить 128 градаций среднего за период значения выходного напряжения.
4. Составить описание на языке VHDL делителя частоты с коэффициентом деления 12000000 и скважностью выходного сигнала <i>Q</i> . Параметр <i>Q</i> определить как настроечную константу <i>Generic</i> .
5. Составить описание на языке VHDL архитектуры вычитающего счетчика с модулем счета 6, работающего по заднему фронту тактовых импульсов частотой 1 Гц, если имеется тактовый сигнал <i>clk</i> частотой 50 Гц и VHDL-описание делителя частоты, в котором определена сущность вида: <pre>entity Divider is generic (M: natural := 8); port (clk_in: in std_logic; -- входной сигнал clk_out: out std_logic -- выходной сигнал); end Divider;</pre> Компонент делителя подключить с использованием поименованного присвоения портов.

6. Составить описание на языке VHDL сторожевого таймера с модулем счета N , заданным в виде настроечной константы *Generic*.

7. Составить описание на языке VHDL вычитающего счетчика с входом асинхронной установки *Set* (активное значение – высокий логический уровень, по которому счетчик принимает максимальное значение) и входом тактовых импульсов *clk*. Модуль счета $K_{сч} = 32$.

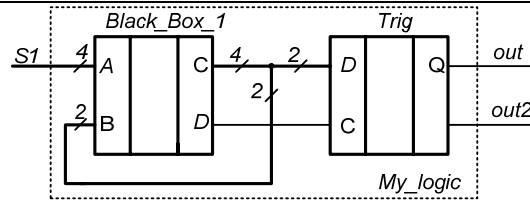
8. Составить описание на языке VHDL без использования оператора *case* дешифратора 5x32 – цифрового устройства, которое ставит входному 5-разрядному вектору X выходной 32-разрядный вектор Y , в котором на позиции с индексом $i = X_{(10)}$ находится логическая единица, а все остальные разряды равны нулю.

9. Составить описание на языке VHDL архитектуры цифрового устройства с помощью структурного стиля программирования. При подключении компонентов в содержательной части архитектуры применить и позиционное, и поименованное подключение портов.



10. Составить описание на языке VHDL параллельного N -разрядного регистра с входом синхронной установки *Set* (активное значение – высокий логический уровень, по которому все разряды принимают единичное состояние). Параметр N определить как глобальную константу, которая задается в пакете.

11. Составить описание на языке VHDL архитектуры цифрового устройства с помощью структурного стиля программирования. При подключении компонентов в содержательной части архитектуры применить и позиционное, и поименованное подключение портов. При наличии на структурной схеме запрещенных идентификаторов заменить их на разрешенные.

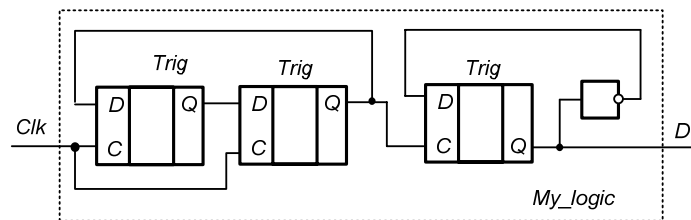


12. Составить описание на языке VHDL D-триггера с входами асинхронной установки *Set* и синхронного сброса *Reset*. Обеспечить работу триггера по заднему фронту тактового сигнала.

13. Составить описание на языке VHDL делителя частоты с коэффициентом деления N и скважностью выходного сигнала 4. Параметр N определить как глобальную константу, которая задается в пакете. Выходной сигнал делителя должен начинаться с низкого логического уровня.

14. Составить описание на языке VHDL D-триггера с входами асинхронной установки *Set* и асинхронного сброса *Reset*. Реализовать запрет одновременной комбинации активных уровней сигналов сброса и установки и не реагировать на нее.

15. Составить описание на языке VHDL архитектуры цифрового устройства с помощью структурного стиля программирования. При подключении компонентов в содержательной части архитектуры применить и позиционное, и поименованное подключение портов. При наличии на структурной схеме запрещенных идентификаторов заменить их на разрешенные.



16. Составить описание на языке VHDL делителя частоты с коэффициентом деления N и скважностью выходного сигнала 2 с асинхронным входом разрешения выдачи выходного сигнала ena . При неактивном логическом уровне ena выдается логический ноль. Параметр N определить как настроечную константу *Generic*.

17. Составить описание на языке VHDL последовательного N -разрядного регистра с входом асинхронного сброса *Reset* (активное значение – высокий логический уровень). Параметр N определить как настроечную константу *Generic*.

В качестве выходных портов определить параллельный код (содержимое всего) регистра и бит из последнего триггера регистра.

18. Составить описание на языке VHDL делителя частоты с коэффициентом деления N и скважностью выходного сигнала 4. Параметр N определить как глобальную константу, которая задается в пакете. Выходной сигнал делителя должен начинаться с высокого логического уровня.

19. Составить описание на языке VHDL шифратора 8х3 – цифрового устройства, которое ставит входному 8-разрядному вектору Y , в котором на позиции с индексом $i = 0 \dots 7$ находится логическая единица, а все остальные разряды равны нулю, выходной 3-разрядный вектор $X = i_{(2)}$.

20. С применением параллельного оператора *GENERATE* составить описание на языке VHDL **архитектуры** делителя частоты с коэффициентом деления 1024, используя **только** существующий модуль делителя частоты с фиксированным коэффициентом деления 2, описание которого имеет вид:

```
entity Div_freq is
port
(clock: in std_logic; -- входной сигнал
clk: out std_logic -- выходной сигнал
);
end Div_freq;
```

21. Составить описание на языке VHDL последовательного N -разрядного регистра с входом синхронного сброса *Reset* (активное значение – низкий логический уровень). Параметр N определить как настроечную константу *Generic*.

Выходной порт – бит из последнего триггера регистра.

22. Составить описание на языке VHDL последовательного N -разрядного регистра с входом разрешения синхронной параллельной загрузки *Load* (активное значение – высокий логический уровень). Параметр N определить как настроечную константу *Generic*.

В качестве выходных портов определить параллельный код (содержимое всего) регистра и бит из последнего триггера регистра.

23. Составить описание на языке VHDL без использования оператора case дешифратора 6х64 – цифрового устройства, которое ставит входному 6-разрядному вектору X выходной 64-разрядный вектор Y, в котором на позиции с индексом $i = X_{(10)}$ находится логическая единица, а все остальные разряды равны нулю.
24. С применением параллельного оператора GENERATE составить описание на языке VHDL архитектуры последовательного регистра для хранения одного байта данных, используя только существующий модуль D-триггера с управляющим входом сброса <i>Reset</i>, тактовым входом <i>clk</i> и информационным входом <i>D</i>, описание которого имеет вид: <pre>entity D_trig is port (clk, reset, D: in std_logic; Q: out std_logic); end Div freq;</pre>
25. Составить описание на языке VHDL ШИМ-модулятора, позволяющего представить 16 градаций среднего за период значения выходного напряжения.
26. Составить описание на языке VHDL параллельного N-разрядного регистра с входом асинхронного сброса <i>Reset</i> (активное значение – низкий логический уровень). Параметр N определить как настроечную константу <i>Generic</i>.
27. Составить описание на языке VHDL суммирующего счетчика с входом асинхронного сброса <i>Reset</i> (активное значение – высокий логический уровень) и входом тактовых импульсов <i>clk</i>. Модуль счета $K_{сч} = 12$. Обеспечить работу счетчика по заднему фронту тактового сигнала.

28. Составить описание на языке VHDL **архитектуры** суммирующего счетчика с модулем счета 10, работающего по переднему фронту тактовых импульсов частотой 1 Гц, если имеется высокочастотный тактовый сигнал *clk32* частотой 32,768 кГц и VHDL-описание делителя частоты, в котором определена сущность вида:

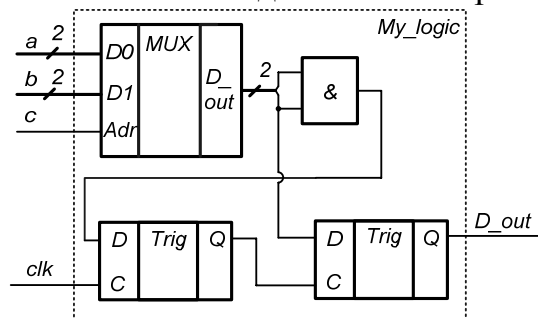
```
entity Div is
generic (M: natural := 8);
port
(clk: in std_logic; -- входной сигнал
clk_out: out std_logic -- выходной сигнал
);
end Div;
```

Компонент делителя подключить с использованием **поименованного** присвоения портов.

29. Составить описание на языке VHDL последовательного *N*-разрядного регистра с входом разрешения асинхронной параллельной загрузки *Load* (активное значение – высокий логический уровень). Параметр *N* определить как настроечную константу *Generic*.

Выходной порт – бит из последнего триггера регистра.

30. Составить описание на языке VHDL **архитектуры** цифрового устройства с помощью структурного стиля программирования. При подключении компонентов в содержательной части архитектуры применить и позиционное, и поименованное подключение портов.

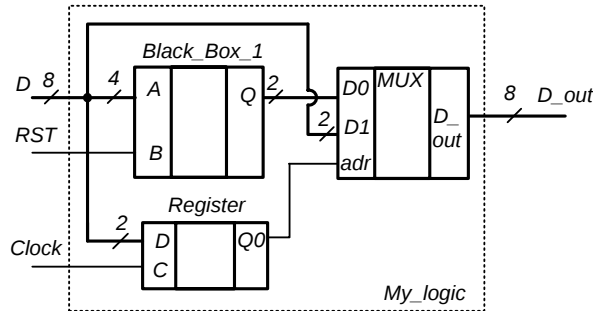


31. Составить описание на языке VHDL генератора *M*-последовательности с порождающим полиномом $P(x) = 1 + x^2 + x^3 + x^4 + x^5$ с входом разрешения асинхронной параллельной загрузки *Load* (активное значение – высокий

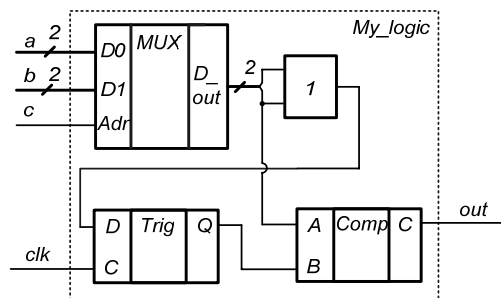
логический уровень) стартовой комбинации <i>Start_data</i> .
32. Составить описание на языке VHDL генератора М-последовательности с порождающим полиномом $P(x) = 1 + x^3 + x^5$ с входом разрешения синхронной параллельной загрузки <i>Load</i> (активное значение – высокий логический уровень) стартовой комбинации <i>Start_data</i> .
33. Составить описание на языке VHDL генератора М-последовательности с порождающим полиномом $P(x) = 1 + x^2 + x^6$ с входом разрешения асинхронной параллельной загрузки <i>Load</i> (активное значение – высокий логический уровень) стартовой комбинации <i>Start_data</i> .
34. Составить описание на языке VHDL вычитающего счетчика с входом синхронного сброса <i>Reset</i> (активное значение – высокий логический уровень) и входом тактовых импульсов <i>clk</i> . Модуль счета $K_{сч} = 14$. Обеспечить работу счетчика по заднему фронту тактового сигнала.
35. Составить описание на языке VHDL архитектуры суммирующего счетчика с модулем счета 20, работающего по заднему фронту тактовых импульсов частотой 12 Гц, если имеется высокочастотный тактовый сигнал <i>clock</i> частотой 24 МГц и VHDL-описание делителя частоты, в котором определена сущность вида: <pre>entity Divider is generic (K: natural := 8); port (clk_in: in std_logic; -- входной сигнал clk_out: out std_logic -- выходной сигнал); end Divider;</pre> Компонент делителя подключить с использованием позиционного присвоения портов.
36. Составить описание на языке VHDL реверсивного счетчика с модулем счета 12, тактовым сигналом <i>clock</i> и входом управления направлением

счета *plus_minus*.

37. Составить описание на языке VHDL **архитектуры** цифрового устройства с помощью структурного стиля программирования. При подключении компонентов в содержательной части архитектуры применить и позиционное, и поименованное подключение портов.



38. Составить описание на языке VHDL **архитектуры** цифрового устройства с помощью структурного стиля программирования. При подключении компонентов в содержательной части архитектуры применить и позиционное, и поименованное подключение портов. При запрещенных идентификаторах заменить их на разрешенные.



39. С применением параллельного оператора *GENERATE* составить описание на языке VHDL **архитектуры** параллельного регистра для хранения одного байта данных, используя **только** существующий модуль D-триггера с управляющим входом установки *Set*, тактовым входом *clk* и информационным входом *D*, описание которого имеет вид:

```
entity D_trig is
port
(clk, set, D: in std_logic;
Q: out std_logic
);
end Div_freq;
```

40. Составить описание на языке VHDL генератора M-последовательности с порождающим полиномом $P(x) = 1 + x + x^3 + x^4 + x^5$ с входом разрешения синхронной параллельной загрузки <i>Load</i> (активное значение – высокий логический уровень) стартовой комбинации <i>Start_data</i> .
41. Составить описание на языке VHDL последовательного N-разрядного регистра с входом разрешения асинхронной параллельной загрузки <i>Load</i> (активное значение – высокий логический уровень). Параметр <i>N</i> определить как настроечную константу <i>Generic</i> .
42. Составить описание на языке VHDL генератора M-последовательности с порождающим полиномом $P(x) = 1 + x + x^2 + x^3 + x^5$ с входом разрешения синхронной параллельной загрузки <i>Load</i> (активное значение – высокий логический уровень) стартовой комбинации <i>Start_data</i> .
43. Составить описание на языке VHDL D-триггера с входами синхронной установки <i>Set</i> и синхронного сброса <i>Reset</i> . Реализовать запрет одновременной комбинации активных уровней сигналов сброса и установки и не реагировать на нее.
44. Составить описание на языке VHDL генератора M-последовательности с порождающим полиномом $P(x) = 1 + x + x^2 + x^4 + x^5$ с входом разрешения асинхронной параллельной загрузки <i>Load</i> (активное значение – высокий логический уровень) стартовой комбинации <i>Start_data</i> .
45. Составить описание на языке VHDL генератора M-последовательности с порождающим полиномом $P(x) = 1 + x^2 + x^6$ с входом разрешения синхронной параллельной загрузки <i>Load</i> (активное значение – высокий логический уровень) стартовой комбинации <i>Start_data</i> .

46. Составить описание на языке VHDL **архитектуры** суммирующего счетчика с модулем счета 20, работающего по заднему фронту тактовых импульсов частотой 12 Гц, если имеется высокочастотный тактовый сигнал *clock* частотой 24 МГц и VHDL-описание делителя частоты, в котором определена сущность вида:

```
entity Divider is
generic (K: natural := 8);
port
(clk_in: in std_logic; -- входной сигнал
clk_out: out std_logic -- выходной сигнал
);
end Divider;
```

Компонент делителя подключить с использованием позиционного присвоения портов.

47. Составить описание на языке VHDL параллельного *N*-разрядного регистра с входом асинхронного сброса *Reset* (активное значение – высокий логический уровень). Параметр *N* определить как настроечную константу *Generic*.

48. Составить описание на языке VHDL делителя частоты с коэффициентом деления *N* и скважностью выходного сигнала 4. Параметр *N* определить как глобальную константу, которая задается в пакете.

Вопросы к экзамену (заочная форма обучения)

1. Архитектура ПЛИС типов CPLD и FPGA. Преимущества и недостатки.
2. Лексические элементы языка VHDL.
3. Типы данных в VHDL. Стандартные типы. Объявление типа и подтипа.
4. Тип `std_logic`.
5. Объекты в VHDL. Описание сущности.
6. Архитектура объекта (сущности).
7. Сигналы и процессы.
8. Структурный стиль описания объектов. Подключение проекта к проекту верхнего уровня. Позиционное и поименованное присвоение.
9. Операции в языке VHDL. Приоритет операций. Особенности сравнения сигналов и переменных разной разрядности.
10. Переменные в VHDL. Разница между переменной и сигналом.
11. Агрегат. Квалифицированное выражение. Имя с индексом и имя-вырезка.
12. Настраечная константа `Generic`. Описание сущности с константой `Generic`. Подключение компонента с настроечной константой к проекту верхнего уровня.
13. Константы и пакеты в VHDL. Подключение пакета к проекту.
14. Функции преобразования типов.
15. Операторы `If` и `Case`.
16. Оператор `With ... Select`. Оператор условного назначения сигнала (`Conditional Signal Assignment`).
17. Оператор генерации `Generate`.
18. Операторы цикла `For ... Loop` и `While`.
19. Записи.
20. Цифровые конечные автоматы (ЦКА). Описание ЦКА с использованием операторов процесса по заданной диаграмме состояний.
21. Интерфейс JTAG.
22. ШИМ-модулятор.
23. Генератор M-последовательности.
24. Сторожевой таймер.
25. Череспериодный компенсатор.

Составил

к.т.н., доцент кафедры РТС

/ И.С. Холопов /

Зав. кафедрой РТС, д.т.н., профессор

/ В.И. Кошелев /