МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
"Казанский (Приволжский) федеральный университет"
Институт физики

УТВЕРЖДАЮ
Проректор по образовательной деятельности КФУ
Проф. Д.А. Таюрский
_____ «___»_____20__г.

подписано электронно-цифровой подписью

Программа дисциплины

Программирование на языке Verilog Б1.В.ДВ.3

Направление подготовки: 09.03.04 - Программная инженерия
Профиль подготовки: Технология проектирования аппаратно-программных информационных систем
Квалификация выпускника: бакалавр
Форма обучения: очное
Язык обучения: русский

Автор(ы):
Акчурин А.Д.

Рецензент(ы):
Юсупов К.М.

СОГЛАСОВАНО:
Заведующий(ая) кафедрой: Акчурин А. Д.
Протокол заседания кафедры No ___ от "____" _____ 201__г
Учебно-методическая комиссия Института физики:
Протокол заседания УМК No ____ от "____" _____ 201__г

Регистрационный No 6165718

Казань
2018

ЭЛЕКТРОННЫЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННО-АНАЛИТИЧЕСКАЯ СИСТЕМА КФУ

Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре основной образовательной программы
3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля
4. Структура и содержание дисциплины/ модуля
5. Образовательные технологии, включая интерактивные формы обучения
6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов
7. Литература
8. Интернет-ресурсы
9. Материально-техническое обеспечение дисциплины/модуля согласно утвержденному учебному плану

Программу дисциплины разработал(а)(и) доцент, к.н. (доцент) Акчурин А.Д. Кафедра радиоастрономии Отделение радиофизики и информационных систем , Adel.Akchurin@kpfu.ru

1. Цели освоения дисциплины

Сформировать современные представления о проектировании логических схем в микросхемах с программируемыми логическими характеристиками, ознакомить с возможностями языкового описания проектируемых схем на примере языка Verilog HDL и научиться составлять программы на языке Verilog HDL

2. Место дисциплины в структуре основной образовательной программы высшего профессионального образования

Данная учебная дисциплина включена в раздел 'Б1.В.ДВ.3 Дисциплины (модули)' основной профессиональной образовательной программы 09.03.04 'Программная инженерия (Технология проектирования аппаратно-программных информационных систем)' и относится к дисциплинам по выбору вариативной части.

Осваивается на 3 курсе в 6 семестре.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля

В результате освоения дисциплины формируются следующие компетенции:

Шифр компетенции	Расшифровка приобретаемой компетенции
ОПК-2 (профессиональные компетенции)	владением архитектурой электронных вычислительных машин и систем
ОПК-3 (профессиональные компетенции)	готовностью применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов
ПК-19 (профессиональные компетенции)	владением навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения
ПК-2 (профессиональные компетенции)	владением навыками использования операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных
ПК-3 (профессиональные компетенции)	владением навыками использования различных технологий разработки программного обеспечения
ПК-21 (профессиональные компетенции)	владением навыками чтения, понимания и выделения главной идеи прочитанного исходного кода, документации

В результате освоения дисциплины студент:

1. должен знать:

понимать структуру и принципы работы микросхем с программируемыми логическими характеристиками, особенности работы схемогенератора, синтезирующего логическую схему на основе языкового описания аппаратуры.

2. должен уметь:

изучать структуру и базовые конструкции языка Verilog HDL, различать операторы языка, используемые в схемном синтезе и в тестировании спроектированных схем.

3. должен владеть:

навыками составления Verilog описаний цифровых устройств в среде САПР Quartus.

4. должен демонстрировать способность и готовность:

ориентироваться в синтезированных схемах на основе RTL-представления;

4. Структура и содержание дисциплины/ модуля

Общая трудоемкость дисциплины составляет 4 зачетных(ые) единиц(ы) 144 часа(ов).

Форма промежуточного контроля дисциплины экзамен в 7 семестре.

Суммарно по дисциплине можно получить 100 баллов, из них текущая работа оценивается в 50 баллов, итоговая форма контроля - в 50 баллов. Минимальное количество для допуска к зачету 28 баллов.

86 баллов и более - "отлично" (отл.);

71-85 баллов - "хорошо" (хор.);

55-70 баллов - "удовлетворительно" (удов.);

54 балла и менее - "неудовлетворительно" (неуд.).

4.1 Структура и содержание аудиторной работы по дисциплине/ модулю

Тематический план дисциплины/модуля

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практические занятия	Лабораторные работы	
1.	Тема 1. Введение. Цели и назначение курса.	7		2	0	0	Устный опрос
2.	Тема 2. Структура программы на языке Verilog. Базовая единица языка - модуль. Элементы проектного модуля - декларации и операторы.	7		2	0	4	Устный опрос
3.	Тема 3. Типы данных. Регистры и цепи, шины. Элементы памяти (массива). Параметры. Операторы языка Verilog. Бинарные операции. Арифметические операции. Операции отношения. Операторы сравнения.	7		2	0	4	Устный опрос

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практические занятия	Лабораторные работы	
4.	Тема 4. Организация комбинаторной логики от простейших вентильных схем до разнообразных мультиплексоров и шифраторов/дешифраторов. Блоки always, управляемые списком чувствительности	7		2	0	6	Устный опрос
5.	Тема 5. Организация последовательностной (регистровой) логики от регистров до счетчиков. Блоки always, управляемые фронтом тактового сигнала	7		2	0	6	Устный опрос
6.	Тема 6. Сдвиговые регистры и линии задержки с заданными ответвлениями	7		2	0	6	Контрольная работа
7.	Тема 7. Реализация проекта, управляющего индикаторными ЖК- и OLED-модулями, на отладочных комплектах MAX II и DE2, DE0-CV (на ПЛИС MAX II, Cyclone II и Cyclone V). Способы тестирования спроектированных схем.	7		6	0	10	Отчет
.	Тема . Итоговая форма контроля	7		0	0	0	Экзамен
	Итого			18	0	36	

4.2 Содержание дисциплины

Тема 1. Введение. Цели и назначение курса.

лекционное занятие (2 часа(ов)):

Структура и принципы работы микросхем с программируемыми логическими характеристиками (ПЛИС). Способы программирования ПЛИС и роль (возможности) языков описания аппаратуры (HDL-языков) для этих целей. Особенности языка Verilog HDL. Ознакомление со средой разработки Quartus II.

Тема 2. Структура программы на языке Verilog. Базовая единица языка - модуль. Элементы проектного модуля - декларации и операторы.

лекционное занятие (2 часа(ов)):

Создание с помощью САПР Quartus II рабочего проекта логической схемы в ПЛИС, расположенной на предложенной преподавателем обучающей плате. Создание проекта включает написание на языке Verilog простейшей схемы, позволяющей по кнопке включение/выключение светодиодов. Назначение выводов ПЛИС (связывание выводов ПЛИС с сигналами, использованными в проекте на Verilog HDL). Конфигурирование (заливка файла конфигурации в ПЛИС).

лабораторная работа (4 часа(ов)):

Создать работоспособную схему, описанную Verilog-кодом. Примеры заданий: 1. Создайте схему, включающую светодиодный индикатор на оценочной плате по состоянию движкового переключателя. 2. Создайте схему, включающую светодиодный индикатор на оценочной плате по состоянию двух движковых переключателей объединенных функцией И. 3. Создайте схему, включающую светодиодный индикатор на оценочной плате по состоянию двух движковых переключателей объединенных функцией ИЛИ. 4. Создайте схему, включающую светодиодный индикатор на оценочной плате по состоянию двух движковых переключателей объединенных функцией ИсклИЛИ.

Тема 3. Типы данных. Регистры и цепи, шины. Элементы памяти (массива). Параметры. Операторы языка Verilog. Бинарные операции. Арифметические операции. Операции отношения. Операторы сравнения.

лекционное занятие (2 часа(ов)):

Создание проекта на Veriog, реализующего различные комбинационные схемы на основе операторов непрерывного присвоения с использованием различных бинарных (побитовые, редуцирующие) и арифметических операций. В Veriog программе также должны использоваться операции отношения и сравнения, а также встроенные в среду разработки примитивы логических элементов. Конкатенация. Комбинационный перемножитель (перемножение "столбиком").

лабораторная работа (4 часа(ов)):

Создать работоспособную схему, описанную Verilog-кодом. Примеры заданий (с реализацией на оценочной плате): 1. Создайте схему, определяющую поразрядную функцию И для двух разрядных шин. Кроме того, схема должна вычислять редуцирующую функцию И с каждой шиной в отдельности. 2. Создайте схему, определяющую поразрядную функцию ИЛИ для двух разрядных шин. Кроме того, схема должна вычислять редуцирующую функцию ИЛИ с каждой шиной в отдельности. 2. Создайте схему, определяющую поразрядную функцию ИсклИЛИ для двух разрядных шин. Кроме того, схема должна вычислять редуцирующую функцию ИсклИЛИ с каждой шиной в отдельности (так называемый бит четности). 3. Создайте модуль компаратора двух 4-разрядных чисел a и b. Модуль компаратора будет генерировать два выходных бита f и q. Один бит f равен 0, если a > b, и равен 1, если a <b. Второй бит q равен 1, если a = b, и равен 0 в противном случае. 4. Создайте 2-битный модуль АЛУ (ALU). Входами модуля являются два 2-битных операнда a и b, 3-битный вход f вида операции и 1-битовый вход переноса из младших разрядов ci. Выходами являются 2-битовый результат D и 1-битовый выход переноса в следующие разряды. Алгоритм работы задается значением f: если f = 1, d = a + b; если f = 2, d = a + b + ci; если f = 3, d = a - b; Если f = 4, d = a - b - ci; если f = 5, d = a OR b; Если f = 6, d = a AND b; если f = 7, d = a XOR b. 5. Создайте схему, реализующей алгоритм определения числа дней в месяце по номеру месяца

Тема 4. Организация комбинаторной логики от простейших вентильных схем до разнообразных мультиплексоров и шифраторов/дешифраторов. Блоки always, управляемые списком чувствительности

лекционное занятие (2 часа(ов)):

Создание проекта на Verilog, реализующего различные комбинационные схемы на основе операторов процедурного (последовательностного) присвоения - блоков always, управляемые списком чувствительности. В Veriog программе реализуются схемы разнообразных мультиплексоров и шифраторов/дешифраторов. Результат отображается светодиодами и семисегментными индикаторами.

лабораторная работа (6 часа(ов)):

ЭЛЕКТРОННЫЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННО-АНАЛИТИЧЕСКАЯ СИСТЕМА КФУ

Создать работоспособную схему, описанную Verilog-кодом. Примеры заданий (с реализацией на оценочной плате): 1. Создайте модуль преобразования 4-битного числа в его дополнение до единицы (ones' complement) и в дополнение до двух (two's complement). 2. Создайте модуль для сложения двух двоично-десятичных чисел. 3. Создайте модуль для вычитания двух двоично-десятичных чисел. 4. Создайте модуль преобразования 8-разрядного числа в соответствующее двоично-десятичное (BCD - binary-decimal code) представление. 5. Создайте модуль преобразования пары двоично-десятичных чисел (BCD) в соответствующий байт. 6. Создайте модуль преобразования 4-битного набора в коде Грея в эквивалентное двоичное число. 7. Создайте модуль преобразования 8-битного набора в коде Грея в эквивалентное двоичное число. 8. Создайте модуль кодировки (прорисовки) одной из первых 6 цифр "левой половинки" штрих-кода UPC-A с шаблоном SLLLLLLMRRRRRRE согласно таблице (где 1 - соответствует черному цвету, 0 - белому) 0 0001101 1 0011001 2 0010011 3 0111101 4 0100011 5 0110001 6 0101111 7 0111011 8 0110111 9 0001011 9. Создайте модуль преобразования 4-битного набора X в семисегментный код (g-f-e-d-c-b-a) по таблице истинности. X g f e d c b a 0 0 1 1 1 1 1 1 1 0 0 0 0 1 1 0 2 1 0 1 1 0 1 1 3 1 0 0 1 1 1 1 4 1 1 0 0 1 1 0 5 1 1 0 1 1 0 1 6 1 1 1 1 1 0 1 7 0 0 0 0 1 1 1 8 1 1 1 1 1 1 1 9 1 1 0 1 1 1 1 A 1 1 1 0 1 1 1 B 1 1 1 1 1 0 0 C 0 1 1 1 0 0 1 D 1 0 1 1 1 1 0 F 1 1 1 1 0 0 1 G 1 1 1 0 0 0 1 10. Адресное пространство памяти 64 КБ разделено на восемь сегментов шириной по 8 КБ. Используя Verilog, опишите дешифратор адреса, который распознает сегменты по 16-битовому адресу. 11. Адресное пространство от предыдущего примера разделено на семь сегментов равной длины (по 8 КБ), и самый верхний восьмой сегмент разделен на четыре части с размером в 2 КБ. Используя Verilog, опишите дешифратор адреса, который распознает сегменты по 16-битовому адресу. Опишите дешифратор, используя блок always и различные процедурные утверждения. Сделайте, по крайней мере, два различных описания дешифратора.

Тема 5. Организация последовательностной (регистровой) логики от регистров до счетчиков. Блоки always, управляемые фронтом тактового сигнала

лекционное занятие (2 часа(ов)):

Создание проекта на Verilog, реализующего различные последовательностные схемы на основе операторов процедурного (последовательностного) присвоения - блоков always, чувствительных к фронту сигнала. В Veriog программе реализуются схемы от регистров до счетчиков с различными типами архитектуры. Делители тактовых сигналов. Генерация импульсов с разной скважностью. Блоки always, нечувствительные к фронту сигнала, с одноальтернативной конструкцией if - триггер, управляемый уровнем управляющего сигнала.

лабораторная работа (6 часа(ов)):

Создать работоспособную схему, описанную Verilog-кодом. Примеры заданий (с реализацией на оценочной плате): 1. Опишите JK и T триггеры, используя блок always в Verilog. 2. Опишите на языке Verilog общий синхронный n-разрядный суммирующий/вычитающий счетчик, что считает до p, когда суммирует, и считает до q, когда вычитает. Используя эту модель, породите экземпляр 8-битового счетчика, доходящего до единицы при суммировании, до двух при вычитании. 3. Опишите асинхронный счётчик со сквозным переносом (ripple counter), который делит входные такты на 32. Для каскадов сквозного переноса (ripple stages) счетчик использует триггер D-типа, выход которого связан обратной связью с его D входом так, что каждый каскад (stage) делит его входные такты на два. Для описания используйте моделирование поведенческого стиля. Как вы будете изменять счетчик для деления входных тактов на число, которое находится между 17 и 31, и оно не может быть выражено как 2k (k - целое число)? 4. Спроектируйте параметризованный делитель частоты, который делит частоту входных тактов на N, и обеспечивает коэффициент заполнения произведенных тактов длительностью M (M<N-1) входных тактов. 5. Создайте счетчик, однократно проходящий все свои состояния от нуля до модуля счета по внешнему сигналу запуска от кнопки (значения счетчика во время его работы создадут положительно растущий ?цифровой зуб пилы?), несколькими способами с помощью: цифрового автомата (FSM-машины), конструкции if else, а также с помощью различных примитивов защелок и триггеров (latch, flip-flop). Сравните полученные RTL-схемы. 6. Создайте счетчик, однократно проходящий все свои состояния от нуля до модуля счета и обратно до нуля по внешнему сигналу от кнопки (теперь значения счетчика будут образовывать положительный и отрицательный цифровой зуб пилы), несколькими способами с помощью: цифрового автомата (FSM-машины), конструкции if else, а также с помощью различных примитивов защелок и триггеров (latch, flip-flop). Сравните полученные RTL-схемы. 7. Создайте счетчик, заданное количество раз проходящий все свои состояния от нуля до модуля счета (несколько положительных цифровых зубов пилы) по внешнему сигналу запуска (например, по нажатию кнопки), несколькими способами с помощью: цифрового автомата (FSM-машины), конструкции if else, а также с помощью различных примитивов защелок и триггеров (latch, flip-flop). Сравните полученные RTL-схемы. 8. Создайте счетчик, заданное количество раз проходящий все свои состояния от нуля до модуля счета и обратно до нуля (несколько положительных и отрицательных цифровых зубов пилы) по внешнему сигналу (например, по нажатию кнопки), несколькими способами с помощью: цифрового автомата (FSM-машины), конструкции if else, а также с помощью различных примитивов защелок и триггеров (latch, flip-flop). Сравните полученные RTL-схемы. 9. Создайте счетчик, заданное количество раз проходящий все свои состояния от нуля до модуля счета, после чего он должен пройти заданное число раз вниз от модуля счета до нуля (сначала несколько положительных цифровых зубов пилы, после которых должны проследовать несколько отрицательных цифровых зубов пилы) по внешнему сигналу, несколькими способами с помощью: цифрового автомата (FSM-машины), конструкции if else, а также с помощью различных примитивов защелок и триггеров (latch, flip-flop). Сравните полученные RTL-схемы. 10. Создайте счетчик, заданное количество раз проходящий все свои состояния от нуля до модуля счета, но с разной скоростью роста (с разным инкрементом счета). При первом проходе инкремент равен 1 (обычный), при втором проходе инкремент равен 2 и так далее (несколько положительных цифровых зубов пилы с разным наклоном) по внешнему сигналу (например, по нажатию кнопки). Реализовать несколькими способами с помощью: цифрового автомата (FSM-машины), конструкции if else, а также с помощью различных примитивов защелок и триггеров (latch, flip-flop). Сравните полученные RTL-схемы.

Тема 6. Сдвиговые регистры и линии задержки с заданными ответвлениями

лекционное занятие (2 часа(ов)):

Создание проекта на Verilog, реализующего различные последовательностные схемы на основе операторов процедурного (последовательностного) присвоения - блоков always, чувствительных к фронту сигнала, с блочным и внеблочным присвоением. Сдвиговые регистры и линии задержки с заданными ответвлениями. Иерархические структуры. Описание памяти с произвольным доступом. Реализация последовательного управления группой светодиодов по принципу бегущей волны или строки.

лабораторная работа (6 часа(ов)):

1. Создайте иллюминацию "бегущие огни" в одном направлении на линейке светодиодов (т.е. выглядящую как светящая точка, всегда смещающаяся в одном направлении) двумя способами с помощью сдвигового регистра и счетчика с дешифратором. Сравните полученные RTL-схемы. 2. Создайте иллюминацию "бегущие огни" со сменяющимся направлением движения на линейке светодиодов (т.е. выглядящую как светящая точка, сначала смещающаяся в одном направлении, а затем в обратном) двумя способами с помощью сдвигового регистра и счетчика с дешифратором. Сравните полученные RTL-схемы. 3. Создайте иллюминацию "бегущие огни" в одном направлении на светодиодах, но с растущей скоростью пробега (т.е. выглядящую как светящая точка, всегда смещающаяся в одном направлении, но после каждого прохода наращивающая скорость пробега) двумя способами с помощью сдвигового регистра и счетчика с дешифратором. Сравните полученные RTL-схемы. 4. Создайте иллюминацию "бегущие огни" со сменяющимся направлением движения на светодиодах и с растущей скоростью пробега (т.е. выглядящую как светящая точка, сначала смещающаяся в одном направлении, а затем в обратном и после каждого прохода наращивающая скорость пробега) двумя способами с помощью сдвигового регистра и счетчика с дешифратором. Сравните полученные RTL-схемы.

Тема 7. Реализация проекта, управляющего индикаторными ЖК- и OLED-модулями, на отладочных комплектах MAX II и DE2, DE0-CV (на ПЛИС MAX II, Cyclone II и Cyclone V). Способы тестирования спроектированных схем.

лекционное занятие (6 часа(ов)):

Цифровые автоматы с конечным числом состояний (Finite State Machines, FSMs). Типовой контроллер ЖК/OLED-модулей HD44780 (Hitachi). Инициалицация ЖК/OLED-модуля (CFAH1602B или WEH001602) и высвечивание на нем состояний сдвигового регистра с линейными обратными связями (LFSR - Linear Feedback Shift Register). Способы тестирования спроектированных схем (компьютерная симуляция, внутрисхемный логический анализатор).

лабораторная работа (10 часа(ов)):

Обучающимся предоставляются отладочные/образовательные платы DK-MAXII-1270N, DE2 и DE0-CV с размещенными на них микросхемами ПЛИС разной степени интеграции (EPM1270F256C5N, EP2C35F672C6 и 5CEBA4F23C7N соответственно). На первых двух платах размещен алфавитно-цифровой ЖК-модуль (CFAH1602B). Более современные символьно-графические OLED-модули (WEH001602 и WEG010016) могут подключаться к отладочным/образовательным платам DE2 и DE0-CV через разъем расширения. Так исторически сложилось, что процедура инициализации ЖК/OLED-модуль достаточно сложная (соответствующая интерфейсу и системе команд контроллера HD44780), и если не пользоваться готовыми схемными/программными решениями, и это позволяет экспериментально на какие узлы разрабатываемой схемы "тратится" наибольшая часть ресурсов ПЛИС. Последнее позволяет яснее понять область применимости ПЛИС и область, где ее применение наиболее эффективно. Для реализации разрабатываемой схемы, помещаемой в ПЛИС, обучающимся необходимо изучить техническую документацию на индикаторные алфавитно-цифровые ЖК/OLED-модули (CFAH1602B/WEH001602). Если не принимать в расчет особенности индикации модулей (путем отражения или свечения, питающие напряжения), то изучение модулей равносильно изучению интерфейса и системы команд контроллера HD44780 (от Hitachi). Опираясь на это описание, необходимо реализовать инициализацию индикаторного модуля. Для тестирования работы различных узлов и наблюдением за их работоспособностью в состав проекта необходимо включать компоненты внутрикристальной отладки. Последняя включает в себя создание на кристалле ПЛИС (через соответствующую мегафункцию) обвязки отлаживаемых блоков проектируемой схемы ячейками, аналогичными ячейкам граничного сканирования, и использование свободной памяти на кристалле для хранения защелкиваемых данных. Вторым путем отладки проекта является возможность быстрого соединения между внутренними сигналами и внешними контактами ввода/вывода без оказания воздействия на основной проект. Естественного в этом случае процесс отладки осуществляется с помощью внешнего анализатора/осциллографа. На последней стадии разработки схемы (когда инициализации прошла успешно) производится синтез и анализ схем, выводящих на индикаторный модуль символов, порожденных LSFR-сдвиговым регистром с подсчетом CRC-циклических контрольных сумм. Работоспособность схем также должна демонстрироваться отладочными средствами. Для наиболее подготовленных студентов предлагается работа с символьно-графическими OLED-модули (WEG010016).

4.3 Структура и содержание самостоятельной работы дисциплины (модуля)

N	Раздел Дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
1.	Тема 1. Введение. Цели и назначение курса.	7		подготовка к устному опросу	4	Устный опрос
2.	Тема 2. Структура программы на языке Verilog. Базовая единица языка - модуль. Элементы проектного модуля - декларации и операторы.	7		подготовка к устному опросу	4	Устный опрос

N	Раздел Дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
3.	Тема 3. Типы данных. Регистры и цепи, шины. Элементы памяти (массива). Параметры. Операторы языка Verilog. Бинарные операции. Арифметические операции. Операции отношения. Операторы сравнения.	7		подготовка к устному опросу	2	Устный опрос
4.	Тема 4. Организация комбинаторной логики от простейших вентильных схем до разнообразных мультиплексоров и шифраторов/дешифраторов. Блоки always, управляемые списком чувствительности	7		подготовка к устному опросу	4	Устный опрос
5.	Тема 5. Организация последовательностной (регистровой) логики от регистров до счетчиков. Блоки always, управляемые фронтом тактового сигнала	7		подготовка к устному опросу	6	Устный опрос
6.	Тема 6. Сдвиговые регистры и линии задержки с заданными ответвлениями	7		подготовка к контрольной работе	10	Контрольная работа
7.	Тема 7. Реализация проекта, управляющего индикаторными ЖК- и OLED-модулями, на отладочных комплектах MAX II и DE2, DE0-CV (на ПЛИС MAX II, Cyclone II и Cyclone V). Способы тестирования спроектированных схем.	7		подготовка к отчету	24	Отчет
	Итого				54	

5. Образовательные технологии, включая интерактивные формы обучения

Активные и интерактивные формы проведения занятий (работа с современным реальными аппаратными и программными средствами системного программирования, выполнение и защита заданий лабораторных работ, разбор конкретных ситуаций, объяснение результатов работы конкретной компьютерной системы)

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

Тема 1. Введение. Цели и назначение курса.

Устный опрос , примерные вопросы:

Ознакомление со средой разработки Quartus II. Порядок создания рабочего проекта логической схемы в ПЛИС, расположенной в на предложенной преподавателем обучающей плате. Способы задания схем в среде Quartus II, отличные от текстового с использованием языка Verilog.

Тема 2. Структура программы на языке Verilog. Базовая единица языка - модуль. Элементы проектного модуля - декларации и операторы.

Устный опрос , примерные вопросы:

Простейшие комбинационные схемы с использованием различных бинарных (побитовые, редуцирующие) и арифметических операций. Встроенные в среду разработки примитивы логических элементов. Буферные элемент. Вход для тактовых частот.

Тема 3. Типы данных. Регистры и цепи, шины. Элементы памяти (массива). Параметры. Операторы языка Verilog. Бинарные операции. Арифметические операции. Операции отношения. Операторы сравнения.

Устный опрос , примерные вопросы:

Возможные способы реализации схем разнообразных мультиплексоров и шифраторов/дешифраторов. Назначение графических средств визуализации схемы на уровне регистровых передач (RTL-viewer).

Тема 4. Организация комбинаторной логики от простейших вентильных схем до разнообразных мультиплексоров и шифраторов/дешифраторов. Блоки always, управляемые списком чувствительности

Устный опрос , примерные вопросы:

Назначение циклов for, repeat, forever и др. Комбинационный перемножитель (перемножение "столбиком"). Побитное сравнение.

Тема 5. Организация последовательностной (регистровой) логики от регистров до счетчиков. Блоки always, управляемые фронтом тактового сигнала

Устный опрос , примерные вопросы:

Реализация последовательностных схем от регистров до счетчиков с различными типами архитектуры. Параллельный перенос. Последовательный перенос. Сквозной перенос.

Тема 6. Сдвиговые регистры и линии задержки с заданными ответвлениями

Контрольная работа , примерные вопросы:

Сдвиговые регистры и линии задержки с заданными ответвлениями. Иерархические структуры. SPI-интерфейс на примере микросхем ЦАП и АЦП - AD5452/AD5453 и AD7356/AD7357.

Тема 7. Реализация проекта, управляющего индикаторными ЖК- и OLED-модулями, на отладочных комплектах MAX II и DE2, DE0-CV (на ПЛИС MAX II, Cyclone II и Cyclone V). Способы тестирования спроектированных схем.

Отчет , примерные вопросы:

Цифровые автоматы с конечным числом состояний (Finite State Machines, FSMs). Типовой LCD-контроллер для высвечивания результатов сдвигового регистра с линейными обратными связями (LFSR - Linear Feedback Shift Register).

Итоговая форма контроля

экзамен

Примерные вопросы к экзамену:

1. Структура и принципы работы микросхем с программируемыми логическими характеристиками (ПЛИС).
2. Структура программы на языке Verilog. Модуль. Декларации и операторы.
3. Операторы непрерывного присвоения с использованием различных бинарных (побитовые, редуцирующие) и арифметических операций.
4. Операторы процедурного (последовательностного) присвоения - блоки always, управляемые списком чувствительности.
5. Операторы процедурного (последовательностного) присвоения - блоки always, чувствительных к фронту сигнала.
6. Цифровые автоматы с конечным числом состояний (Finite State Machines, FSMs).
7. Условный оператор if, оператор выбора case и операторы повторения. Блочное и внеблочное присваивание.
8. Типы данных. Регистры и цепи, шины. Элементы памяти (массива). Параметры. Вещественные числа.
9. Операторы языка Verilog. Бинарные операции (побитовые, редуцирующие). Арифметические операции. Операции отношения. Операторы сравнения. Конкатенация. Встроенные примитивы.
10. Графические средств визуализации разрабатываемой схемы на уровне регистровых передач (RTL-viewer)
11. Организация последовательностной (регистровой) логики. Назначение циклов for, repeat, forever и др.
12. Сдвиговые регистры и линии задержки с заданными ответвлениями.
13. Цифровые автоматы с конечным числом состояний (Finite State Machines, FSMs).
14. Сдвиговый регистр с линейными обратными связями (LFSR - Linear Feedback Shift Register).

РЕГЛАМЕНТ БАЛЛЬНО-РЕЙТИНГОВОЙ СИСТЕМЫ ПО ДИСЦИПЛИНЕ:

Выполнение 6 лабораторных работ (первые 5 - по 5 баллов, последняя - по 15 баллов)

Активность и результативность устного опроса в течение семестра - 10 баллов (макс.)

Итого: за работу в семестре 50 баллов (макс.)

К зачету допускаются студенты, набравшие не менее 27,5 баллов за семестр.

Максимальная оценка за экзамен - 50 баллов.

Итоговый регламент по дисциплине: 100 баллов (макс.).

7.1. Основная литература:

Максфилд, К. Проектирование на ПЛИС. Курс молодого бойца [Электронный ресурс] : учеб. пособие - Электрон. дан. - Москва : ДМК Пресс, 2010. - 407 с. - Режим доступа: https://e.lanbook.com/book/60987.

Наваби, З. Проектирование встраиваемых систем на ПЛИС [Электронный ресурс] - Электрон. дан. - Москва : ДМК Пресс, 2016. - 464 с. - Режим доступа: https://e.lanbook.com/book/73058.

7.2. Дополнительная литература:

Стешенко, В.Б. ПЛИС фирмы Altera: элементная база, система проектирования и языки описания аппаратуры [Электронный ресурс] - Электрон. дан. - Москва : ДМК Пресс, 2010. - 573 с. - Режим доступа: https://e.lanbook.com/book/60976.

Книшев, Д.А. ПЛИС фирмы 'Xilink': описание структуры основных семейств [Электронный ресурс] / Д.А. Книшев, М.О. Кузелин. - Электрон. дан. - Москва : ДМК Пресс, 2010. - 230 с. - Режим доступа: https://e.lanbook.com/book/60951.

Левицкий, А. А. Проектирование микросистем. Программные средства обеспечения САПР [Электронный ресурс] : Учеб. пособие / А. А. Левицкий, П. С. Маринушкин. - Красноярск : Сиб. федер. ун-т, 2010. - 156 с. - ISBN 978-5-7638-2111-6. URL: http://znanium.com/catalog.php?bookinfo=442124

7.3. Интернет-ресурсы:

Verilog. Базовый курс. - http://digitrode.ru/computing-devices/fpga/80-verilog-bazovyy-kurs-chast-i.html

Verilog Tutorial - http://www.asic-world.com/verilog/veritut.html

Акчурин А.Д., Юсупов К.М. Программирование на языке Verilog. Учебное пособие. ? Казань, 2016. ? 90 с. - http://kpfu.ru/portal/docs/F2032376552/Verilog_Examples_D.pdf

Краткий курс HDL - http://www.kit-e.ru/articles/circuit/2008_3_161.php

Обзор литературы по FPGA программированию - Junior FPGA Design Engineer: как стать? - https://habrahabr.ru/post/281525/

Основы HDL Verilog как средства проектирования цифровых устройств - http://emirs.miet.ru/oroks-miet/upload/normal/003q8u0xcty3w3/POPOVA.pdf

Раздел "ModelSim по-русски" - http://www.naliwator.narod.ru/modelsim.html

Учебник Verilog - http://dev.visuale.ru/visuale/verilog

8. Материально-техническое обеспечение дисциплины(модуля)

Освоение дисциплины "Программирование на языке Verilog" предполагает использование следующего материально-технического обеспечения:

Компьютерный класс, представляющий собой рабочее место преподавателя и не менее 15 рабочих мест студентов, включающих компьютерный стол, стул, персональный компьютер, лицензионное программное обеспечение. Каждый компьютер имеет широкополосный доступ в сеть Интернет. Все компьютеры подключены к корпоративной компьютерной сети КФУ и находятся в едином домене.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе "ZNANIUM.COM", доступ к которой предоставлен студентам. ЭБС "ZNANIUM.COM" содержит произведения крупнейших российских учёных, руководителей государственных органов, преподавателей ведущих вузов страны, высококвалифицированных специалистов в различных сферах бизнеса. Фонд библиотеки сформирован с учетом всех изменений образовательных стандартов и включает учебники, учебные пособия, УМК, монографии, авторефераты, диссертации, энциклопедии, словари и справочники, законодательно-нормативные документы, специальные периодические издания и издания, выпускаемые издательствами вузов. В настоящее время ЭБС ZNANIUM.COM соответствует всем требованиям федеральных государственных образовательных стандартов высшего профессионального образования (ФГОС ВПО) нового поколения.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе Издательства "Лань" , доступ к которой предоставлен студентам. ЭБС Издательства "Лань" включает в себя электронные версии книг издательства "Лань" и других ведущих издательств учебной литературы, а также электронные версии периодических изданий по естественным, техническим и гуманитарным наукам. ЭБС Издательства "Лань" обеспечивает доступ к научной, учебной литературе и научным периодическим изданиям по максимальному количеству профильных направлений с соблюдением всех авторских и смежных прав.

Для выполнения практических и лабораторных работ студентам предоставляются оценочные комплекты (платы для обучения) с различными ПЛИС, произведенных фирмой Altera. Это комплекты DK-MAXII-1270N и DE2 (Develpment and Educatin Bard для Cyclne II), DE2-115 (Develpment and Educatin Bard для Cyclne IV) и DE0-CV (Develpment and Educatin Bard для Cyclne V)

Программа составлена в соответствии с требованиями ФГОС ВПО и учебным планом по направлению 09.03.04 "Программная инженерия" и профилю подготовки Технология проектирования аппаратно-программных информационных систем .

Автор(ы):
Акчурин А.Д. _____
"__" _____ 201 __ г.

Рецензент(ы):
Юсупов К.М. _____
"__" _____ 201 __ г.