

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Воронежский государственный технический университет»

УТВЕРЖДАЮ

Декан факультета

радиотехники и электроники

Небольсин В.А.

«31» августа 2020 г.



РАБОЧАЯ ПРОГРАММА

дисциплины

«Проектирование больших интегральных схем на системном уровне»

Направление подготовки 11.04.04 Электроника и нанoeлектроника

Профиль Интегральные системы и устройства в микро- и нанoeлектронике

Квалификация выпускника магистр

Нормативный период обучения 2 года/2 года 3 месяца

Форма обучения очная/заочная

Год начала подготовки 2020

Автор программы

Строгонов А.В.

Заведующий кафедрой
полупроводниковой электроники
и нанoeлектроники

Рембеза С.И.

Руководитель ОПОП

Рембеза С.И.

Воронеж 2020

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Цели дисциплины – обеспечение основ проектирования БИС, БИС программируемой логики и БИС типа «система на кристалле» (SoC) с использованием системного уровня проектирования.

1.2. Задачи освоения дисциплины

- изучение основных современных архитектур БИС, ПЛИС и СБИС типа SoC;
- изучение маршрутов проектирования БИС (ПЛИС) с использованием инструментов системного уровня проектирования и языка SystemC/C++;
- изучение проблем, связанных с проектированием БИС по субмикронным проектным нормам и методы их решения;
- получение практических навыков работы с системой визуально-имитационного моделирования Matlab/Simulink для разработки имитационных моделей на уровне системы с последующим созданием функциональных моделей на языке VHDL;
- освоение языка VHDL для написания кода основных функциональных цифровых блоков БИС;
- освоение языка Verilog-A для написания кода поведенческих моделей аналого-цифровых блоков БИС;
- получение практических навыков работы с САПР БИС Tanner для разработки электрических схем и топологии функциональных блоков по масштабируемой субмикронной КМОП-технологии

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина Б1.В.ДВ.02.02 «Проектирование больших интегральных схем на системном уровне» относится к дисциплинам части блока Б1 учебного плана, формируемой участниками образовательных отношений.

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Проектирование больших интегральных схем на системном уровне» направлен на формирование следующих компетенций:

ПК-5: способность разрабатывать эффективные алгоритмы решения сформулированных задач с использованием современных языков программирования и обеспечивать их программную реализацию;

ПК-7: способность к восприятию, разработке и критической оценке новых способов проектирования твердотельных приборов и устройств.

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ПК-5	знать основные узлы вычислительных устройств цифровой обработки сигналов; основы высокоуровневого языка описания аппа-

	ратных средств (VHDL); уметь строить имитационные модели сложно-функциональных цифровых устройств обработки сигналов в системе Matlab/Simulink с использованием языка М-файла, fi-объектов и графического представления цифровых автоматов (StateFlow); функциональные модели цифровых устройств обработки сигналов с применением языка VHDL и мегафункций в САПР ПЛИС Altera Quartus II и Xilinx ISE; владеть навыками работы с САПР ПЛИС Altera Quartus II или Xilinx ISE (Vivado).
ПК-7	знать теорию цифровых кодов; форматы представления чисел с фиксированной и плавающей запятой; методы обработки цифровых сигналов; методы расчета аналоговых и дискретных фильтров; основные структуры цифровых фильтров; уметь работать с пакетом Signal Processing среды FDATATool системы Matlab для расчета параметров цифровых фильтров; владеть навыками работы с системой визуально-имитационного моделирования Matlab/Simulink и приложением по извлечению кода языка VHDL из моделей устройств цифровой обработки сигналов; проектирования цифровых фильтров в базе ПЛИС.

4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Проектирование больших интегральных схем на системном уровне» составляет 6 зачетных единиц.

Распределение трудоемкости дисциплины по видам занятий

очная форма обучения

Виды учебной работы	Всего часов	Семестры
		3
Аудиторные занятия (всего)	66	66
В том числе:		
Лекции	16	16
Практические занятия (ПЗ)	16	16
Лабораторные работы (ЛР)	34	34
Самостоятельная работа	123	123
Курсовая работа	+	+
Часы на контроль	27	27
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость академические часы	216	216
з.е.	6	6

заочная форма обучения

Виды учебной работы	Всего часов	Семестры
		4

Аудиторные занятия (всего)	16	16
В том числе:		
Лекции	4	4
Практические занятия (ПЗ)	4	4
Лабораторные работы (ЛР)	8	8
Самостоятельная работа	191	191
Курсовая работа	+	+
Часы на контроль	9	9
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость академические часы з.е.	216 6	216 6

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий

очная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Прак зан.	Лаб. зан.	СРС	Всего, час
1	Маршрут проектирования БИС СнК и технологические платформы	Основные этапы проектирования БИС и систем на кристалле. Системное проектирование и верификация. Архитектура программного обеспечения. Системная интеграция. Функциональное проектирование и верификация. Физическое проектирование и верификация топологии.	2	-	4	11	17
2	Особенности СнК-проектирования	Язык SystemC в проектировании на системном уровне. Точность моделирования с использованием SystemC. Построение модели функционального виртуального прототипа (FVP) на основе ARM. Создание встроенных программ. Разработка кода SW-приложения. Функционально-архитектурное совместное проектирование (HW-SW). Средства алгоритмического проектирования, анализа и реализации (SPW). Средства совместного проектирования (CoDesign).	2	-	4	11	17
3	Аналоговые и цифровые сложно-функциональные блоки БИС СнК	Сложно-функциональные цифровые и аналоговые блоки. Архитектурное планирование кристалла. Языки описания поведенческих моделей аналого-цифровых блоков: Verilog-A, Verilog-AMS, VHDL-AMS. Симуляторы SPICE, Spectre-Verilog и AMS Designer фирмы Cadence. Основы языка Verilog-A.	2	2	4	11	19
4	Разработка моделей вычислительных устройств на языке Verilog-A	Моделирование аналого-цифровых систем с использованием языка Verilog-A. Основные конструкции языка Verilog-A. Описание комбинационных и последовательностных блоков. Модели цифровых устройств на языке Verilog-A.	2	2	4	13	21
5	Программные средства для разработки встраиваемых систем на ПЛИС СнК	Характеристики микропроцессорных ядер PicoBlaze, MicroBlazer, PowerPC фирмы Xilinx для реализации в базе ПЛИС. Архитектура, функциональные модули и набор инструкций микропроцессорного ядра Nios II фирмы Altera для реализации в базе	2	2	4	13	21

		ПЛИС. Компоненты SOPC Builder. Интерфейс Avalon.					
6	Модельно-ориентированное проектирование устройств ЦОС на ПЛИС в среде Xilinx System Generator	Разработка имитационных моделей цифровых устройств обработки сигналов на верхнем уровне с использованием стандартных блоков DSP System Toolbox системы Matlab/Simulink. Разработка имитационных моделей цифровых устройств обработки сигналов на верхнем уровне с использованием блоков среды Xilinx System Generator и системы Matlab/Simulink.	2	2	4	13	21
7	Программные инструменты для Verilog-синтеза в базис ПЛИС/БИС	Использование программных инструментов с открытым кодом Yosys/ABC (синтез и оптимизация), Graywolf (глобальное размещение), Qrouter (детальная трассировка), Magic (топологический редактор) для автоматического синтеза топологии	2	2	4	13	21
8	Промышленные форматы САПР заказных БИС	Ознакомление с промышленными форматами САПР БИС — LEF (Library Exchange Format, формат библио-течного обмена), DEF (Design Exchange Format, формат конструктивного обмена), GDSII (для топологии)	2	2	2	13	19
9	Использование системы визуально-имитационного моделирования MATLAB/SIMULINK для проектирования цифровых фильтров в САПР ISE DESIGN SUITE с применением среды System Generator	Система Xilinx System Generator САПР ISE Design Suite. Создание проектов КИХ-фильтров.	-	2	2	13	17
10	Использование системы визуально-имитационного моделирования MATLAB/SIMULINK для проектирования цифровых фильтров в САПР QUARTUS II с применением среды Altera DSP Builder	Система визуально-имитационного моделирования Matlab/Simulink с использованием Altera DSP Builder. Создание проектов КИХ-фильтров	-	2	2	12	16
Всего			16	16	34	123	189
Контроль							27
Итого							216

заочная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Прак зан.	Лаб. зан.	СРС	Всего, час
1	Маршрут проектирования БИС СнК и технологические платформы	Основные этапы проектирования БИС и систем на кристалле. Системное проектирование и верификация. Архитектура программного обеспечения. Системная интеграция. Функциональное проектирование и верификация. Физическое проектирование и верификация топологии.	2	-	2	18	22
2	Особенности СнК-проектирования	Язык SystemC в проектировании на системном уровне. Точность моделирования с использованием SystemC. Построение модели функционального виртуального прототипа (FVP) на основе ARM. Создание встроенных программ. Разработка кода SW-приложения. Функционально-архитектурное совместное проектирование (HW-SW). Средства алгоритмического проектирования, анализа и реализации (SPW). Средства совместного проектирования	2	-	2	18	22

		(CoDesign).					
3	Аналоговые и цифровые сложно-функциональные блоки БИС СнК	Сложно-функциональные цифровые и аналоговые блоки. Архитектурное планирование кристалла. Языки описания поведенческих моделей аналого-цифровых блоков: Verilog-A, Verilog-AMS, VHDL-AMS. Симуляторы SPICE, Spectre-Verilog и AMS Designer фирмы Cadence. Основы языка Verilog-A.	-	-	2	18	20
4	Разработка моделей вычислительных устройств на языке Verilog-A	Моделирование аналого-цифровых систем с использованием языка Verilog-A. Основные конструкции языка Verilog-A. Описание комбинационных и последовательностных блоков. Модели цифровых устройств на языке Verilog-A.	-	-	2	18	20
5	Программные средства для разработки встраиваемых микропроцессорных систем на ПЛИС СнК	Характеристики микропроцессорных ядер PicoBlaze, MicroBlazer, PowerPC фирмы Xilinx для реализации в базисе ПЛИС. Архитектура, функциональные модули и набор инструкций микропроцессорного ядра Nios II фирмы Altera для реализации в базисе ПЛИС. Компоненты SOPC Builder. Интерфейс Avalon.	-	-	-	20	20
6	Модельно-ориентированное проектирование устройств ЦОС на ПЛИС в среде Xilinx System Generator.	Разработка имитационных моделей цифровых устройств обработки сигналов на верхнем уровне с использованием стандартных блоков DSP System Toolbox системы Matlab/Simulink. Разработка имитационных моделей цифровых устройств обработки сигналов на верхнем уровне с использованием блоков среды Xilinx System Generator и системы Matlab/Simulink.	-	-	-	20	20
7	Программные инструменты для Verilog-синтеза в базис ПЛИС/БИС	Использование программных инструментов с открытым кодом Yosys/ABC (синтез и оптимизация), Graywolf (глобальное размещение), Qrouter (детальная трассировка), Magic (топологический редактор) для автоматического синтеза топологии	-	-	-	20	20
8	Промышленные форматы САПР заказных БИС	Ознакомление с промышленными форматами САПР БИС — LEF (Library Exchange Format, формат библиотечного обмена), DEF (Design Exchange Format, формат конструктивного обмена), GDSII (для топологии)	-	-	-	20	20
9	Использование системы визуально-имитационного моделирования MATLAB/SIMULINK для проектирования цифровых фильтров в САПР ISE DESIGN SUITE с применением среды System Generator	Система Xilinx System Generator САПР ISE Design Suite. Создание проектов КИХ-фильтров.	-	2	-	20	22
10	Использование системы визуально-имитационного моделирования MATLAB/SIMULINK для проектирования цифровых фильтров в САПР QUARTUS II с применением среды Altera DSP Builder	Система визуально-имитационного моделирования Matlab/Simulink с использованием Altera DSP Builder. Создание проектов КИХ-фильтров	-	2	-	19	21
Всего			4	4	8	191	207
Контроль							9
Итого							216

5.2 Перечень лабораторных работ

1. Изучение среды системной интеграции Qsys САПР ПЛИС СнК Quartus II Altera
2. Реализация интерфейсов в Qsys САПР ПЛИС СнК Quartus II Altera
3. Разработка маршрута проектирования БИС СнК с использованием программных средств с открытым доступом.
4. Изучение программного инструмента Yosys для Verilog-синтеза в базис заказных БИС
5. Проектирование аналогового блока СнК, на примере блока АЦП. Создание модели АЦП на языке Verilog-A
6. Проектирование цифрового блока СнК, на примере SPI-интерфейса. Создание модели ядра на языке Verilog-A
7. Разработка цифрового блока БИС СнК 8-разрядного микропроцессорного ядра на языке Verilog-A
8. Разработка модели микропроцессора MIPS на языке VHDL и его реализация в базисе ПЛИС
9. Получение практических навыков работы с отладочной платой DE2 фирмы Altera со встроенным софт-процессором Nios II на ПЛИС типа CYCLONE
10. Программирование на языке ассемблера софт-процессора Nios II
11. Пример разработки имитационной модели ЦОС-блока верхнего уровня в элементах Xilinx System Generator
12. Пример проектирования КИХ-фильтров на распределенной арифметике в базисе ПЛИС с применением генератора параметризованных ядер XLogiCORE IP и функции FIR Compiler v5.0 САПР Xilinx
13. Изучение маршрута проектирования заказных БИС Qflow
14. Синтез микропроцессорные ядра or1200 в базис заказной БИС с использованием Qflow и топологической библиотеки OSU 0.35 с топологическими проектными нормами масштабируемой КМОП-технологии 0,35 мкм
15. Настройка промышленных форматов LEF, DEF, GDSII с помощью скриптового языка tcl в маршруте заказных БИС Qflow 2.0
16. Проектирование последовательных КИХ-фильтров со структурой МАС-блоков в системе Xilinx System Generator САПР ISE Design Suite
17. Проектирование последовательных КИХ-фильтров в системе визуально-имитационного моделирования Matlab/Simulink с использованием Altera DSP Builder

6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсовой работы в 3 семестре для очной формы обучения, в 4 семестре для заочной формы обучения.

Примерная тематика курсовых работ:

1. Разработка модели КИХ-фильтра Добеши в системе визуально-имитационного моделирования Matlab/Simulink с использованием Altera DSP Builder
2. Разработка модели последовательного КИХ-фильтра на 4 отвода в системе визуально-имитационного моделирования Matlab/Simulink с применением Altera DSP Builder с использованием линии задержки на регистрах
3. Разработка имитационной модели последовательного КИХ-фильтра на 4 отвода в системе Xilinx System Generator с применением библиотеки Reference BlockSet/DSP
4. Разработка имитационной модели последовательного КИХ-фильтра на 32 отвода в системе Xilinx System Generator САПР ISE Design Suite
5. Разработка имитационной модели параллельного КИХ-фильтра на 8 отводов в системе Xilinx System Generator САПР ISE Design Suite
6. Разработка функциональной модели систолического КИХ-фильтра в базе ПЛИС Xilinx с применением генератора параметризованных ядер XLogiCORE IP и функции FIR Compiler v6.3
7. Разработка функциональной модели и испытательного стенда на языке VHDL систолического КИХ-фильтра в базе ПЛИС с использованием системы цифрового моделирования ModelSim-Altera
8. Разработка функциональной модели систолического КИХ-фильтра в базе ПЛИС Altera с использованием САПР Quartus II
9. Разработка функциональной модели КИХ-фильтра на распределенной арифметике в базе ПЛИС с применением генератора параметризованных ядер XLogiCORE IP и функции FIR Compiler v5.0 САПР Xilinx ISE Design Suite
10. Разработка функциональной модели КИХ-фильтра на параллельной распределенной арифметике в базе ПЛИС Altera. Линия задержки на мегафункции ALTSHIFT_TAPS, дерево многоразрядных сумматоров на мегафункциях LPM_ADD_SUB, частичные произведения реализовать в LUT на языке VHDL
11. Разработка функциональной модели КИХ-фильтра на последовательной распределенной арифметике в базе ПЛИС Altera
12. Разработка модели последовательного КИХ-фильтра на 4 отвода в системе визуально-имитационного моделирования Matlab/Simulink с применением Altera DSP Builder с использованием линии задержки на двухпортовой памяти
13. Разработка модели последовательного КИХ-фильтра на 32 отвода в системе визуально-имитационного моделирования Matlab/Simulink с применением Altera DSP Builder. Коэффициенты фильтра рассчитать в среде FDA-Tool.
14. Разработка модели параллельного КИХ-фильтра на 4 отвода в системе визуально-имитационного моделирования Matlab/Simulink с применением Altera DSP Builder.
15. Разработка функциональной модели КИХ-фильтра на параллельной

распределенной арифметике в базе ПЛИС Altera с использованием мегаядра FIR Compiler САПР Quartus II

16. Разработка функциональной модели КИХ-фильтра на последовательной распределенной арифметике в базе ПЛИС Altera с использованием мегаядра FIR Compiler САПР Quartus II

Задачи, решаемые при выполнении курсовой работы:

– изучить структуры цифровых фильтров для реализации как в базе сигнальных процессоров так и ПЛИС;

– получить навыки работы с системой визуально-имитационного моделирования Matlab/Simulink с пакетами расширения Altera DSP Builder и Xilinx System Generator;

– получить навыки работы с САПР ПЛИС Altera и Xilinx для разработки функциональных моделей цифровых устройств обработки сигналов.

Курсовая работа включает в себя графическую часть и расчетно-пояснительную записку.

Учебным планом по дисциплине «Проектирование больших интегральных схем на системном уровне» не предусмотрено выполнение контрольной работы.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

7.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

7.1.1 Этап текущего контроля

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
ПК-5	знать основные узлы вычислительных устройств цифровой обработки сигналов; основы высокоуровневого языка описания аппаратных средств (VHDL);	Способность описать цифровой узел HDL-кодом	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь строить имитационные модели сложно-функциональных цифровых устройств обработ-	Демонстрация навыков расчета коэффициентов цифровых фильтров с помощью М-скриптов и построение моделей с использованием последова-	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

	ки сигналов в системе Matlab/Simulink с использованием языка М-файла, fi-объектов и графического представления цифровых автоматов (State-Flow); функциональные модели цифровых устройств обработки сигналов с применением языка VHDL и мегафункций в САПР ПЛИС Altera Quartus II и Xilinx ISE;	тельной и параллельной распределенной арифметике с учетом архитектурных особенностей ПЛИС		
	владеть навыками работы с САПР ПЛИС Altera Quartus II или Xilinx ISE (Vivado).	Демонстрация навыков реализации устройств ЦОС в САПР ПЛИС Altera Quartus II или Xilinx ISE (Vivado)	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ПК-7	знать теорию цифровых кодов; форматы представления чисел с фиксированной и плавающей запятой; методы обработки цифровых сигналов; методы расчета аналоговых и дискретных фильтров; основные структуры цифровых фильтров;	Способность использования функций Matlab для расчета частотных характеристик линейных цепей, для расчета аналоговых фильтров-прототипов	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь работать с пакетом Signal Processing среды FDA-Tool системы Matlab для расчета параметров цифровых фильтров;	Демонстрация навыков работы со спецификациями фильтров и их синтез в FDATool	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть навыками работы с системой визуально-имитационного моделирования Matlab/Simulink и приложением по извлечению кода языка VHDL из моделей устройств цифровой обработки сигналов; проектирования цифровых фильтров в базе ПЛИС.	Способность реализовать проект в САПР ПЛИС Quartus II по HDL-коду имитационной модели извлеченного с помощью HDL Coder системы Matlab/Simulink или по HDL- коду извлеченному с помощью пакета расширения Altera DSP Builder	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 3 семестре для очной формы обучения, 4 семестре для заочной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

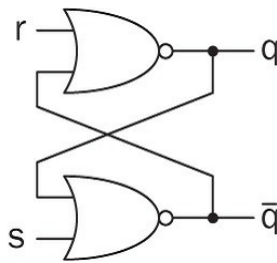
«неудовлетворительно».

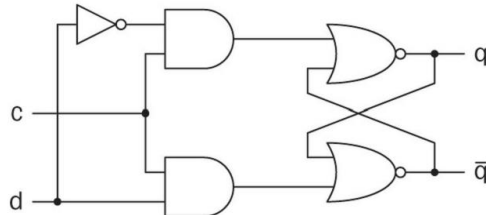
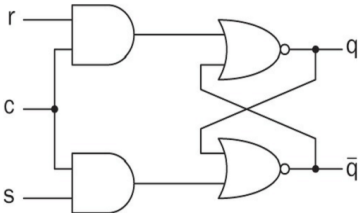
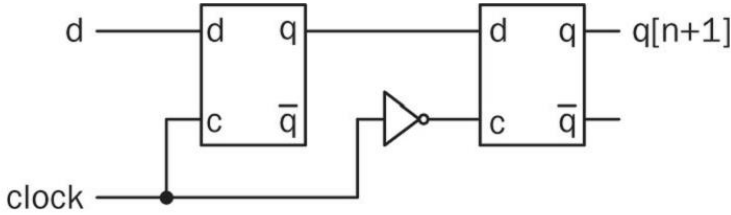
Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл	Неудовл
ПК-5	знать основные узлы вычислительных устройств цифровой обработки сигналов; основы высокоуровневого языка описания аппаратных средств (VHDL);	Тест	Выполнение теста на 90 – 100 %	Выполнение теста на 80 – 90 %	Выполнение теста на 70 – 80 %	В тесте менее 70 % правильных ответов
	уметь строить имитационные модели сложных функциональных цифровых устройств обработки сигналов в системе Matlab/Simulink с использованием языка М-файла, fi-объектов и графического представления цифровых автоматов (StateFlow); функциональные модели цифровых устройств обработки сигналов с применением языка VHDL и мегафункций в САПР ПЛИС Altera Quartus II и Xilinx ISE;	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть навыками работы с САПР ПЛИС Altera Quartus II или Xilinx ISE (Vivado).	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ПК-7	знать теорию цифровых кодов; форматы представления чисел с фиксированной и плава-	Тест	Выполнение теста на 90 – 100 %	Выполнение теста на 80 – 90 %	Выполнение теста на 70 – 80 %	В тесте менее 70 % правильных ответов

	ющей запятой; методы обработки цифровых сигналов; методы расчета аналоговых и дискретных фильтров; основные структуры цифровых фильтров;					
	уметь работать с пакетом Signal Processing среды Fdatool системы Matlab для расчета параметров цифровых фильтров;	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть навыками работы с системой визуальной имитационного моделирования Matlab/Simulink и приложением по извлечению кода языка VHDL из моделей устройств цифровой обработки сигналов; проектирования цифровых фильтров в базе ПЛИС.	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

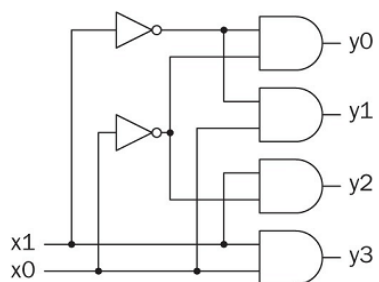
7.2 Примерный перечень оценочных средств (типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

7.2.1 Примерный перечень заданий для подготовки к тестированию

1	Руководствуясь электрической схемой, таблицей истинности, определите ее функциональное назначение.  <table><tr><th colspan="2">Inputs</th><th>Output</th></tr><tr><th>s</th><th>r</th><th>q</th></tr><tr><td>0</td><td>0</td><td>q_{prev}</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>U</td></tr></table> <table><tr><th>1</th><th>2</th><th>3</th><th>4</th></tr><tr><td>RS-защелка</td><td>RSC-защелка</td><td>JK-триггер</td><td>D-триггер</td></tr></table>	Inputs		Output	s	r	q	0	0	q_{prev}	0	1	0	1	0	1	1	1	U	1	2	3	4	RS-защелка	RSC-защелка	JK-триггер	D-триггер
Inputs		Output																									
s	r	q																									
0	0	q_{prev}																									
0	1	0																									
1	0	1																									
1	1	U																									
1	2	3	4																								
RS-защелка	RSC-защелка	JK-триггер	D-триггер																								
2	Руководствуясь электрической схемой, определите ее функциональное назначение.																										

	 <table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>RS-защелка</td><td>RS-защелка со входом управления</td><td>D-триггер, тактируемый уровнем синхросигнала</td><td>D-триггер, тактируемый фронтом синхросигнала</td></tr></table>	1	2	3	4	RS-защелка	RS-защелка со входом управления	D-триггер, тактируемый уровнем синхросигнала	D-триггер, тактируемый фронтом синхросигнала																												
1	2	3	4																																		
RS-защелка	RS-защелка со входом управления	D-триггер, тактируемый уровнем синхросигнала	D-триггер, тактируемый фронтом синхросигнала																																		
3	Руководствуясь электрической схемой, таблицей истинности, определите ее функциональное назначение.  <table><tr><th colspan="3">Inputs</th><th>Output</th></tr><tr><th>s</th><th>r</th><th>c</th><th>q</th></tr><tr><td>-</td><td>-</td><td>0</td><td>q_{prev}</td></tr><tr><td>0</td><td>0</td><td>1</td><td>q_{prev}</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>U</td></tr></table><table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>RS-защелка</td><td>RS-защелка со входом управления</td><td>JK-триггер</td><td>D-триггер</td></tr></table>	Inputs			Output	s	r	c	q	-	-	0	q_{prev}	0	0	1	q_{prev}	0	1	1	0	1	0	1	1	1	1	1	U	1	2	3	4	RS-защелка	RS-защелка со входом управления	JK-триггер	D-триггер
Inputs			Output																																		
s	r	c	q																																		
-	-	0	q_{prev}																																		
0	0	1	q_{prev}																																		
0	1	1	0																																		
1	0	1	1																																		
1	1	1	U																																		
1	2	3	4																																		
RS-защелка	RS-защелка со входом управления	JK-триггер	D-триггер																																		
4	Руководствуясь электрической схемой, определите ее функциональное назначение.  <table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>RS-защелка</td><td>RS-защелка со входом управления</td><td>D-триггер, тактируемый уровнем синхросигнала</td><td>D-триггер, тактируемый фронтом синхросигнала</td></tr></table>	1	2	3	4	RS-защелка	RS-защелка со входом управления	D-триггер, тактируемый уровнем синхросигнала	D-триггер, тактируемый фронтом синхросигнала																												
1	2	3	4																																		
RS-защелка	RS-защелка со входом управления	D-триггер, тактируемый уровнем синхросигнала	D-триггер, тактируемый фронтом синхросигнала																																		
5	Руководствуясь электрической схемой, определите ее функциональное назначение.																																				

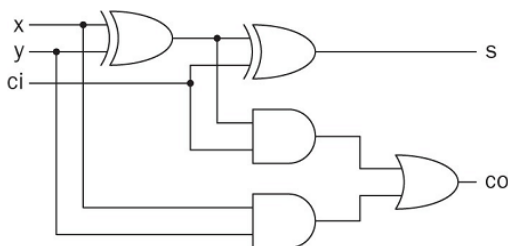
	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>RS-защелка</td><td>RS-защелка со входом управления</td><td>JK-триггер</td><td>D-триггер</td></tr></table>	1	2	3	4	RS-защелка	RS-защелка со входом управления	JK-триггер	D-триггер																																																	
1	2	3	4																																																							
RS-защелка	RS-защелка со входом управления	JK-триггер	D-триггер																																																							
6	Руководствуясь электрической схемой, таблицей истинности, определите ее функциональное назначение. <table><tr><th colspan="2">Select pins</th><th>Output</th></tr><tr><th>s1</th><th>s0</th><th>y</th></tr><tr><td>0</td><td>0</td><td>x0</td></tr><tr><td>0</td><td>1</td><td>x1</td></tr><tr><td>1</td><td>0</td><td>x2</td></tr><tr><td>1</td><td>1</td><td>x3</td></tr></table> <table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>Мультиплексор 4 в 1</td><td>Демультимплексор 1 в 4</td><td>Дешифратор 1 в 4</td><td>Дешифратор 4 в 3</td></tr></table>	Select pins		Output	s1	s0	y	0	0	x0	0	1	x1	1	0	x2	1	1	x3	1	2	3	4	Мультиплексор 4 в 1	Демультимплексор 1 в 4	Дешифратор 1 в 4	Дешифратор 4 в 3																															
Select pins		Output																																																								
s1	s0	y																																																								
0	0	x0																																																								
0	1	x1																																																								
1	0	x2																																																								
1	1	x3																																																								
1	2	3	4																																																							
Мультиплексор 4 в 1	Демультимплексор 1 в 4	Дешифратор 1 в 4	Дешифратор 4 в 3																																																							
7	Руководствуясь электрической схемой, таблицей истинности, определите ее функциональное назначение. <table><tr><th colspan="4">Inputs</th><th colspan="3">Outputs</th></tr><tr><th>x0</th><th>x1</th><th>x2</th><th>x3</th><th>y0</th><th>y1</th><th>v</th></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>-</td><td>-</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>-</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>-</td><td>-</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>-</td><td>-</td><td>-</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table> <table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>Приоритетный шифратор</td><td>Шифратор</td><td>Дешифратор</td><td>Демультимплексор</td></tr></table>	Inputs				Outputs			x0	x1	x2	x3	y0	y1	v	0	0	0	0	-	-	0	1	0	0	0	0	0	1	-	1	0	0	0	1	1	-	-	1	0	1	0	1	-	-	-	1	1	1	1	1	2	3	4	Приоритетный шифратор	Шифратор	Дешифратор	Демультимплексор
Inputs				Outputs																																																						
x0	x1	x2	x3	y0	y1	v																																																				
0	0	0	0	-	-	0																																																				
1	0	0	0	0	0	1																																																				
-	1	0	0	0	1	1																																																				
-	-	1	0	1	0	1																																																				
-	-	-	1	1	1	1																																																				
1	2	3	4																																																							
Приоритетный шифратор	Шифратор	Дешифратор	Демультимплексор																																																							
8	Руководствуясь электрической схемой, таблицей истинности, определите ее функциональное назначение.																																																									



Inputs		Outputs			
x0	x1	y0	y1	y2	y3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

1	2	3	4
Полусумматор	Сумматор	Интегратор	Дешифратор 2 в 4

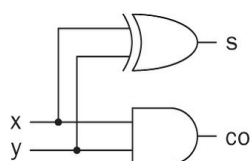
9 Руководствуясь электрической схемой, таблицей истинности, определите ее функциональное назначение.



Inputs			Outputs	
x	y	ci	s	co
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

1	2	3	4
Полусумматор	Сумматор	Интегратор	Демультимплексор 2 в 4

10 Руководствуясь электрической схемой, таблицей истинности, определите ее функциональное назначение.

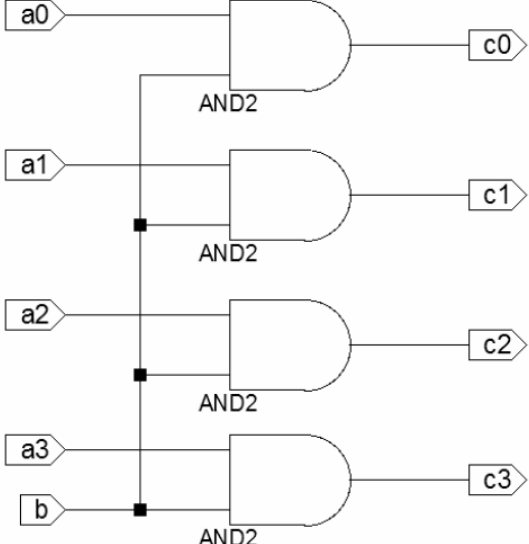


Inputs		Outputs	
x	y	s	co
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

1	2	3	4
Полусумматор	Сумматор	Интегратор	Демультимплексор 2 в 4

--	--

7.2.2 Примерный перечень заданий для решения стандартных задач

1	С помощью оператора цикла for синтезируйте схему <pre> module for_ex(input [3:0] a, input b, output reg [3:0] c); integer i; always @ (a, b) begin for (i = 0; i < 4; i = i + 1) c[i] = b & a[i]; end endmodule </pre> 
2	Разработайте проект мультиплексора 2 в 1 на языке Verilog с использованием процедурного блока always: <pre> module MUX2 (input A, B, SEL, output reg OUT1); always @ (A, B, SEL) if (SEL) OUT1 = A; else OUT1 = B; endmodule </pre>
3	Разработайте проект D-триггера на языке Verilog: <pre> module dff(input clk, input d, output reg q); always @ (posedge clk) q <= d; endmodule </pre>

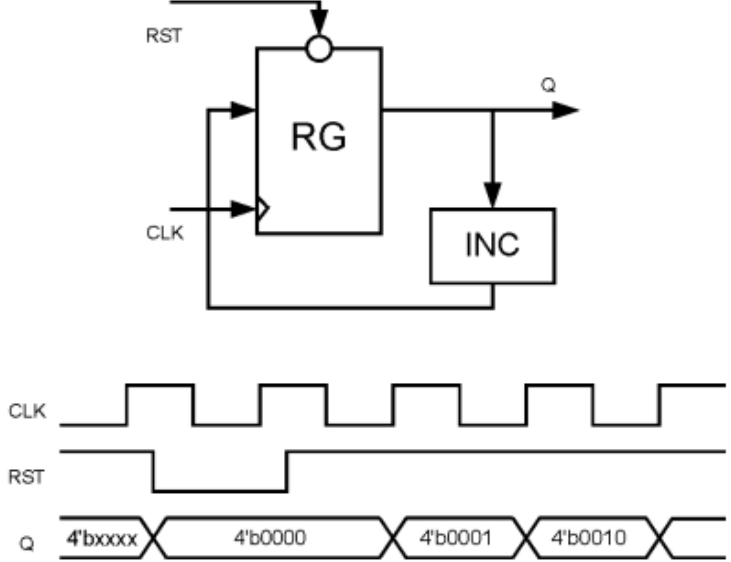
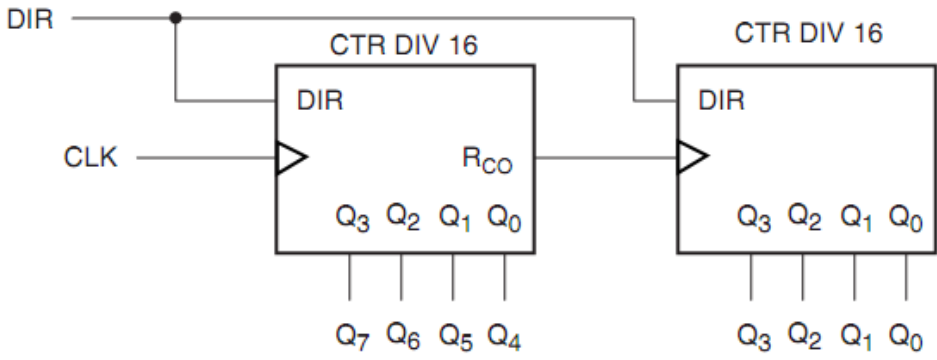
4	Используя Verilog-код, разработайте проекты 8- и 16-разрядных регистров. <pre> module reg8(input clk, input [7:0] d, output reg [7:0] q); always @ (posedge clk) q <= d; endmodule 16-разрядный регистр: module reg16(input clk, input [15:0] d, output reg [15:0] q); always @ (posedge clk) q <= d; endmodule </pre>
5	По фрагменту Verilog-кода разработайте проект D-триггера с синхронным сбросом. <pre> always @(posedge clk) if (reset) begin q <= 0; end else begin q <= d; end end </pre>
6	По фрагменту Verilog-кода разработайте проект D-триггера с асинхронным сбросом и разрешением сброса ce. <pre> always @(posedge clk or posedge reset) if (reset) begin q <= 0; end else if (ce) begin q <= d; end end </pre>
7	По фрагменту Verilog-кода с использованием оператора case разработайте проект мультиплексора 4 в 1. <pre> always @ * begin case (sel) 2'b00 : q = a; 2'b01 : q = b; 2'b10 : q = c; 2'b11 : q = d; default : q = 1'bx; endcase end </pre>
8	Используя Verilog-код, разработайте проект сдвигового регистра.

	<pre> module shift_reg(input clk, input d_in, input ce, output [15:0] q); reg [15:0] data; always @(posedge clk) begin if (ce) data <= {data[15:1], d_in}; end assign q = data; endmodule </pre>
9	Используя фрагмент Verilog-кода, разработайте проект реверсивного счетчика. <pre> reg [7:0] cnt; always @(posedge <clock>) if (up_down) cnt <= cnt + 1; else cnt <= cnt - 1; </pre>
10	Используя фрагмент Verilog-кода, разработайте проект счетчика с синхронным сбросом reset и загрузкой load . <pre> reg [7:0] cnt; always @(posedge clk) if (reset) cnt <= 0; else if (ce) if (load) cnt <= d_in; else cnt <= cnt + 1; </pre>

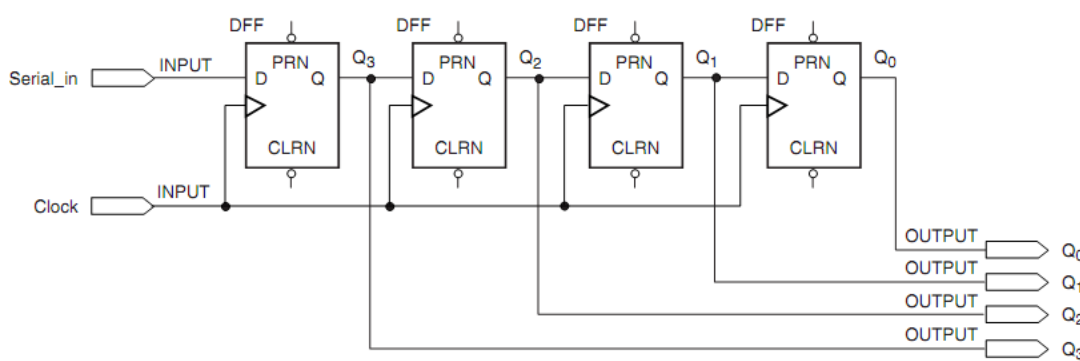
7.2.3 Примерный перечень заданий для решения прикладных задач

1	С использованием языка VHDL разработайте проект 4-разрядного счетчика с синхронной загрузкой LOAD, асинхронным сбросом RESET и сигналом разрешения счета Count Enable в САПР ПЛИС Altera Quartus II. Правильность разработки подтвердите функциональным моделированием.
---	--

2	Используя шаблон, разработайте проект последовательного устройства на языке Verilog. <pre> always @(posedge CLK or negedge RST) if (RST == 1'b0) //Если сигнал сброса нулевой (активный) D <= 1'b0; else //Если сигнал сброса не нулевой, событие – фронт CLK D <= A & B; </pre>
3	Разработайте проект мультиплексора, используется оператор выбора case <pre> case (sel[1:0]) 2'b00: A = 1'b0; 2'b01: A = B; 2'b10: A = ~B; default: A = 1'b1; // все остальные случаи endcase </pre>
4	По RTL-коду разработайте проект 4-разрядного сумматора. <pre> module ADD4(S, A, B); output [3:0] S; input [3:0] A, B; wire [3:0] P; assign P[3:0] = A[3:0] & B[3:0] A[3:0] & {P[2:0],1'b0} {P[2:0],1'b0} & B[3:0]; assign S[3:0] = A[3:0] ^ B[3:0] ^ {P[2:0],1'b0}; endmodule </pre>
5	По структурной схеме 4-разрядного счетчика и временной диаграмме его работы разработайте проект.

	
6	Разработайте проект каскадирования 4-разрядных синхронных счетчиков с помощью сигнала сквозного переноса RCO в САПР ПЛИС Quartus II. 
7	Разработайте проект 8-разрядного реверсивного счетчика со схемой окончания счета (сигнал max_min) с использованием высокоуровневого языка VHDL в САПР ПЛИС Quartus II. Счетчик дополнительно имеет синхронный сигнал разрешения счета count_ena, асинхронную загрузку load и сброс clear. <pre> ENTITY pre_ct8a IS PORT (clk, count_ena : IN BIT; clear, load, direction : IN BIT; p : IN INTEGER RANGE 0 TO 255; max_min : OUT BIT; qd : OUT INTEGER RANGE 0 TO 255); END pre_ct8a; ARCHITECTURE a OF pre_ct8a IS BEGIN PROCESS (clk, clear, load) VARIABLE cnt : INTEGER RANGE 0 TO 255; BEGIN IF (clear = '0') THEN — Asynchronous clear cnt := 0; ELSIF (load = '1' and clear = '1') THEN — Asynchronous load cnt := p; ELSE </pre>

	<pre> IF (clk'EVENT AND clk = '1') THEN IF (count_ena = '1' and direction = '0') THEN cnt := cnt - 1; ELSIF (count_ena = '1' and direction = '1') THEN cnt := cnt + 1; END IF; END IF; END IF; qd <= cnt; — — Terminal count decoder IF (cnt = 0 and direction = '0') THEN max_min <= '1'; ELSIF (cnt = 255 and direction = '1') THEN max_min <= '1'; ELSE max_min <= '0'; END IF; END PROCESS; END a; </pre>
8	Разработайте проект 8-разрядного реверсивного счетчика со схемой окончания счета (сигнал max_min) с использованием высокоуровневого языка VHDL в САПР ПЛИС Quartus II. Счетчик дополнительно имеет синхронный сигнал разрешения счета count_ena, синхронную загрузку load и синхронный сброс clear. <pre> ENTITY pre_ct8s IS PORT (clk, count_ena : IN BIT; clear, load, direction : IN BIT; p : IN INTEGER RANGE 0 TO 255; max_min : OUT BIT; qd : OUT INTEGER RANGE 0 TO 255); END pre_ct8s; ARCHITECTURE a OF pre_ct8s IS BEGIN PROCESS (clk) VARIABLE cnt : INTEGER RANGE 0 TO 255; BEGIN IF (clk'EVENT AND clk = '1') THEN IF (clear = '0') THEN — — Synchronous clear cnt := 0; ELSIF (load = '1') THEN — — Synchronous load cnt := p; ELSIF (count_ena = '1' and direction = '0') THEN cnt := cnt - 1; ELSIF (count_ena = '1' and direction = '1') THEN cnt := cnt + 1; END IF; END IF; END IF; qd <= cnt; — — Terminal count decoder IF (cnt = 0 and direction = '0') THEN max_min <= '1'; ELSIF (cnt = 255 and direction = '1') THEN </pre>

	<pre> max_min <= '1'; ELSE max_min <= '0'; END IF; END PROCESS; END a; </pre>
9	Разработайте проект цифрового устройства с использованием высокоуровневого языка Verilog в САПР ПЛИС Quartus II: 4-разрядный реверсивный счетчик, с асинхронным входом установки (Set), синхронным сбросом (Reset), сигналом разрешения загрузки Eна. Для отладки проекта используйте учебно-лабораторный стенд LESO2.1.
10	С использованием языка VHDL разработайте проект 4-разрядного последовательного регистра со сдвигом вправо в САПР ПЛИС Altera Quartus II. Правильность разработки подтвердите функциональным моделированием. 

7.2.4 Примерный перечень вопросов для подготовки к зачету

Не предусмотрено учебным планом

7.2.5 Примерный перечень вопросов для подготовки к экзамену

1. Маршрут проектирования БИС СнК и технологические платформы
2. Основные этапы проектирования БИС и систем на кристалле.
3. Системное проектирование и верификация.
4. Архитектура программного обеспечения.
5. Системная интеграция.
6. Функциональное проектирование и верификация.
7. Физическое проектирование и верификация топологии.
8. Особенности СнК-проектирования.
9. Язык SystemC в проектировании на системном уровне.
10. Точность моделирования с использованием SystemC.
11. Построение модели функционального виртуального прототипа (FVP) на основе ARM.
12. Создание встроенных программ.
13. Разработка кода SW-приложения.
14. Функционально-архитектурное совместное проектирование (HW-SW).
15. Средства алгоритмического проектирования, анализа и реализации

- (SPW).
16. Средства совместного проектирования (CoDesign).
 17. Аналоговые и цифровые сложно-функциональные блоки БИС СнК.
 18. Архитектурное планирование кристалла.
 19. Языки описания поведенческих моделей аналого-цифровых блоков:
Verilog-A, Verilog-AMS, VHDL-AMS.
 20. Симуляторы SPICE, Spectre-Verilog и AMS Designer фирмы Cadence.
 21. Основы языка Verilog-A. Разработка моделей вычислительных устройств на языке Verilog-A.
 22. Основные конструкции языка Verilog-A.
 23. Описание комбинационных и последовательностных блоков.
 24. Модели цифровых устройств на языке Verilog-A.
 25. Программные средства для разработки встраиваемых микропроцессорных систем на ПЛИС СнК
 26. Характеристики микропроцессорных ядер PicoBlaze, MicroBlazer, PowerPC фирмы Xilinx для реализации в базисе ПЛИС.
 27. Архитектура, функциональные модули и набор инструкций микропроцессорного ядра Nios II фирмы Altera для реализации в базисе ПЛИС.
 28. Компоненты SOPC Builder. Интерфейс Avalon.
 29. Модельно-ориентированное проектирование устройств ЦОС на ПЛИС в среде Xilinx System Generator
 30. Разработка имитационных моделей цифровых устройств обработки сигналов на верхнем уровне с использованием стандартных блоков DSP System Toolbox системы Matlab/Simulink.
 31. Разработка имитационных моделей цифровых устройств обработки сигналов на верхнем уровне с использованием блоков среды Xilinx System Generator и системы Matlab/Simulink.
 32. Программные инструменты для Verilog-синтеза в базис ПЛИС/БИС
 33. Использование программных инструментов с открытым кодом Yosys/ABC (синтез и оптимизация), Graywolf (глобальное размещение), Qrouter (детальная трассировка), Magic (топологический редактор) для автоматического синтеза топологии
 34. Промышленные форматы САПР заказных БИС LEF (Library Exchange Format, формат библиотечного обмена)
 35. Промышленные форматы САПР заказных БИС DEF (Design Exchange Format, формат конструктивного обмена)
 36. Промышленные форматы САПР заказных БИС GDSII (для топологии).
 37. Использование системы визуально-имитационного моделирования MATLAB/SIMULINK для проектирования цифровых фильтров в САПР ISE DESIGN SUITE с применением среды System Generator
 38. Система Xilinx System Generator САПР ISE Design Suite. Создание проектов КИХ-фильтров.
 39. Использование системы визуально-имитационного моделирования MATLAB/SIMULINK для проектирования цифровых фильтров в САПР QUARTUS II с применением среды Altera DSP Builder

40. Система визуально-имитационного моделирования Matlab/Simulink с использованием Altera DSP Builder. Создание проектов КИХ-фильтров

7.2.6. Методика выставления оценки при проведении промежуточной аттестации

Промежуточная аттестация проводится по билетам, каждый из которых содержит 3 вопроса и задачу. Каждый правильный ответ на вопрос оценивается 2 баллами, задача оценивается в 4 балла. Максимальное количество набранных баллов – 10.

1. Оценка «Отлично» ставится, если студент набрал от 9 до 10 баллов.
2. Оценка «Хорошо» ставится в случае, если студент набрал от 6 до 8 баллов.
3. Оценка «Удовлетворительно» ставится в случае, если студент набрал от 3 до 5 баллов.
4. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 3 баллов.

При получении оценок «отлично», «хорошо» и «удовлетворительно» требуемые в рабочей программе знания, умения, владения по соответствующим компетенциям на промежуточном этапе считаются достигнутыми.

7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1	Маршрут проектирования БИС СнК и технологические платформы	ПК-5, ПК-7	Тест, требования к курсовой работе
2	Особенности СнК-проектирования	ПК-5, ПК-7	Тест, требования к курсовой работе
3	Аналоговые и цифровые сложно-функциональные блоки БИС СнК	ПК-5, ПК-7	Тест, требования к курсовой работе
4	Разработка моделей вычислительных устройств на языке Verilog-A	ПК-5, ПК-7	Тест, требования к курсовой работе
5	Программные средства для разработки встраиваемых микропроцессорных систем на ПЛИС СнК	ПК-5, ПК-7	Тест, требования к курсовой работе
6	Модельно-ориентированное проектирование устройств ЦОС на ПЛИС в среде Xilinx System Generator	ПК-5, ПК-7	Тест, требования к курсовой работе
7	Программные инструменты для Verilog-синтеза в базис ПЛИС/БИС	ПК-5, ПК-7	Тест, требования к курсовой работе
8	Промышленные форматы САПР заказных БИС	ПК-5, ПК-7	Тест, требования к курсовой работе
9	Использование системы визуально-имитационного моделирования MATLAB/SIMULINK для проектирования цифровых фильтров в САПР ISE DESIGN SUITE с приме-	ПК-5, ПК-7	Тест, требования к курсовой работе

	нением среды System Generator		
10	Использование системы визуально-имитационного моделирования MATLAB/SIMULINK для проектирования цифровых фильтров в САПР QUARTUS II с применением среды Altera DSP Builder	ПК-5, ПК-7	Тест, требования к курсовой работе

7.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Тестирование осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных тест-заданий на бумажном носителе. Время тестирования 30 мин. Затем осуществляется проверка теста экзаменатором и выставляется оценка согласно методики выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Защита курсовой работы осуществляется согласно требованиям, предъявляемым к работе, описанным в методических материалах. Примерное время защиты на одного студента составляет 20 мин.

8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

8.1 Перечень учебной литературы, необходимой для освоения дисциплины

1. Мартюшев Ю.Ю. Практика функционального цифрового моделирования в радиотехнике: учеб. пособие / Ю.Ю. Мартюшев. – М.: Горячая линия-Телеком, 2014. –186 с.

2. Солонина А.И. Основы цифровой обработки сигналов : учеб. пособие / А.И. Солонина, Д.А. Улахович, С.М. Арбузов и др. – СПб. : БХВ-Петербург 2005. –753 с.

3. Соловьев В.В. Проектирование цифровых систем на основе ПЛИС / В.В. Соловьев. – 2-е изд., стереотип. – М.: Горячая линия – Телеком, 2007 – 636 с.

4. Грушвицкий Р.И. Проектирование систем на микросхемах с про-

граммируемой структурой: учеб. пособие / Р.И. Грушвицкий, А.Х. Мурсаев, Е.П. Угрюмов. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2006 – 736.

5. Проектирование систем цифровой и смешанной обработки сигналов / под ред.; У. Кестера; пер. с англ. под ред. А.А. Власенко. – М.: Техносфера, 2010. – 326 с.,

6. Амосов В.В. Схемотехника и средства проектирования цифровых устройств: учеб. пособие / В.В. Амосов. – СПб.: БХВ-Петербург, 2007. – 542 с.

7. Тарасов И.Е. Разработка систем цифровой обработки сигналов на базе ПЛИС / И.Е. Тарасов, И.Е. Потехин. – М.: Горячая линия – Телеком, 2007. – 248 с.

8. Тарасов И.Е. Разработка цифровых устройств на основе ПЛИС фирмы Xilinx с применением языка VHDL / И.Е. Тарасов. – М.: Горячая линия – Телеком, 2005. – 252 с.

9. Угрюмов Е.П. Цифровая схемотехника: учеб. пособие / Е.П. Угрюмов. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2004. – 528 с.

10. Строгонов А.В. Цифровая обработка сигналов в базисе программируемых логических интегральных: учеб. пособие / А.В. Строгонов. – 2-е изд., испр. и доп. – СПб.: Лань, 2015. – 310 с. [Электронный ресурс].

11. Строгонов А.В. Основы цифровой обработки сигналов: учеб. пособие / А.В. Строгонов. – Воронеж: ФГБОУ ВПО «ВГТУ», 2014. [Электронный ресурс].

12. Строгонов А.В. Проектирование устройств цифровой обработки сигналов для реализации в базисе программируемых логических интегральных схем: учеб. пособие / А.В. Строгонов. Воронеж: ФГБОУ ВПО «ВГТУ», 2013. – 323 с. [Электронный ресурс].

13. Строгонов А.В. Системное проектирование программируемых логических интегральных схем: учеб. пособие / А.В. Строгонов. – Воронеж: ФГБОУ ВПО «ВГТУ», 2012. – 322 с. [Электронный ресурс].

14. Строгонов А.В. Методические указания к выполнению лабораторных работ № 1, 2 по дисциплине «Проектирование цифровых устройств в базисе ПЛИС» / А.В. Строгонов. – Воронеж: ФГБОУ ВПО «ВГТУ», 2014 [Электронный ресурс].

15. ГОСТ 2.105-2019. ЕСКД. Общие требования к текстовым документам. – М.: ФГУП «Стандартинформ», 2019. – 35 с.

8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:

Методические указания к выполнению лабораторных работ представлены на сайте: <http://cchgeu.ru/>.

Образовательный портал ВГТУ: <https://old.education.cchgeu.ru>

САПР БИС Tanner
 САПР ПЛИС Altera Quartus II+ Altera DSP Builder
 САПР ПЛИС Xilinx ISE+ Xilinx System Generator
 Системы цифрового моделирования ModelSim-Altera
 Система визуально-имитационного моделирования Matlab/Simulink
www.labfor.ru Лаборатории электронных средств обучения, ЛЭСО ГОУ ВПО «СибГУТИ»
www.asic.ru НПК «Технологический центр»
www.tcen.ru НПК «Технологический центр»
www.e-kir.ru Электронные версии журнала «Компоненты и технология»

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

1. Специализированная лекционная аудитория, оснащенная оборудованием для лекционных демонстраций и проекционной аппаратурой.
2. Дисплейный класс, оснащенный компьютерными программами для выполнения расчетов, и рабочими местами для самостоятельной подготовки обучающихся с выходом в «Интернет».
3. Учебный лабораторный стенд на ПЛИС структуры FPGA - LESO2.

10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

По дисциплине «Проектирование больших интегральных схем на системном уровне» читаются лекции, проводятся лабораторные и практические занятия, выполняется курсовая работа.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Практические занятия проводятся путем решения стандартных и прикладных задач в аудитории.

Лабораторные работы выполняются на лабораторном оборудовании в соответствии с методиками, приведенными в указаниях к выполнению работ.

Большое значение по закреплению и совершенствованию знаний имеет самостоятельная работа студентов. Информацию о всех видах самостоятельной работы студенты получают на занятиях.

Методика выполнения курсовой работы изложена в учебно-методическом пособии. Выполнять этапы курсовой работы студенты должны своевременно и в установленные сроки.

Контроль усвоения материала дисциплины осуществляется тестированием, проверкой курсовой работы, защитой курсовой работы. Освоение дисциплины оценивается на экзамене.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно

	фиксировать основные положения, выводы, формулировки, обобщения; пометать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.
Практическое занятие	Конспектирование рекомендуемых источников. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы. Прослушивание аудио- и видеозаписей по заданной теме, выполнение расчетно-графических заданий, решение задач по алгоритму.
Лабораторная работа	Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: разобрать лекцию по соответствующей теме, ознакомиться с соответствующим разделом учебника, проработать дополнительную литературу и источники, решить задачи и выполнить другие письменные задания.
Самостоятельная работа	Самостоятельная работа студентов способствует глубокому усвоения учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: - работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций; - работа над темами для самостоятельного изучения; - участие в работе студенческих научных конференций, олимпиад; - подготовка к промежуточной аттестации.
Подготовка к промежуточной аттестации	Готовиться к промежуточной аттестации следует систематически, в течение всего семестра. Интенсивная подготовка должна начаться не позднее, чем за месяц-полтора до промежуточной аттестации. Данные перед экзаменом три дня эффективнее всего использовать для повторения и систематизации материала.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Перечень вносимых изменений	Дата вне- сения из- менений	Подпись заведую- щего кафедрой, от- ветственной за реа- лизацию ОПОП
1			
2			
3			
4			
5			