Минобрнауки России

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Оренбургский государственный университет имени В.А. Бондаренко»

Кафедра математики и цифровых технологий

РАБОЧАЯ ПРОГРАММА

ДИСЦИПЛИНЫ

«Б1.Д.В.Э.3.2 Разработка и отладка программного обеспечения для встраиваемых систем»

Уровень высшего образования

БАКАЛАВРИАТ

Направление подготовки
09.03.04 Программная инженерия
(код и наименование направления подготовки)

Разработка корпоративных систем и цифровых решений
(наименование направленности (профиля) образовательной программы)

Квалификация
Бакалавр

Форма обучения
Очная

Год набора 2026

Рабочая программа дисциплины *«Б1.Д.В.Э.3.2 Разработка и отладка программного обеспечения для встраиваемых систем»* рассмотрена и утверждена на заседании кафедры

Кафедра математики и цифровых технологий
наименование кафедры

протокол № _____ от "___" _____ 20__ г.

Заведующий кафедрой
Кафедра математики и цифровых технологий _____ А.Е. Шухман
наименование кафедры *подпись* *расшифровка подписи*

Исполнители:
Доцент _____ А.И. Сарайкин
должность *подпись* *расшифровка подписи*

_____
должность *подпись* *расшифровка подписи*

СОГЛАСОВАНО:
Председатель методической комиссии по направлению подготовки
09.03.04 Программная инженерия _____ И.П. Болодурина
код наименование *личная подпись* *расшифровка подписи*

Заведующий отделом формирования фонда и научной обработки документов
_____ С.А. Бихтимирова
личная подпись *расшифровка подписи*

Уполномоченный по качеству института
_____ С.Н. Морозова
личная подпись *расшифровка подписи*

№ регистрации _____

© Сарайкин А.И., 2026
© ОГУ, 2026

1 Цели и задачи освоения дисциплины

Цель (цели) освоения дисциплины: формирование у студентов целостного представления о жизненном цикле разработки программного обеспечения для встраиваемых систем и закрепление практических навыков создания, отладки и оптимизации низкоуровневого кода, работающего на ограниченных аппаратных ресурсах.

Задачи:

1. Изучение архитектурных особенностей целевых аппаратных платформ (RISC-V, ARM) и их влияния на разработку программного обеспечения.
2. Освоение методов разработки низкоуровневого кода и драйверов на языках C и ассемблера с использованием кросс-компиляторов и инструментальных цепочек (GCC, ARMCC).
3. Получение практических навыков настройки и использования периферийных модулей микроконтроллера (UART, SPI, I2C, GPIO, таймеры, АЦП) через прямое взаимодействие с регистрами управления.
4. Освоение принципов проектирования и реализации обработчиков прерываний для работы в режиме реального времени.
5. Изучение методов и инструментов отладки встраиваемых систем, включая внутрисхемные отладчики (JTAG), трассировку, логические анализаторы и программные отладчики (GDB).
6. Формирование навыков разработки и отладки кода в среде с операционной системой реального времени (RTOS) и в среде Bare-metal.
7. Развитие умения разрабатывать и выполнять тестовые сценарии для проверки работоспособности и надежности встраиваемого ПО.

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к дисциплинам (модулям) по выбору вариативной части блока Д «Дисциплины (модули)»

Пререквизиты дисциплины: *Б1.Д.В.8 Тестирование цифровых решений, Б1.Д.В.11 Интеграция информационных систем*

Постреквизиты дисциплины: *Отсутствуют*

3 Требования к результатам обучения по дисциплине

Процесс изучения дисциплины направлен на формирование следующих результатов обучения

Код и наименование формируемых компетенций	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций
ПК*-13 Способен разрабатывать, оптимизировать и отлаживать системное программное обеспечение	ПК*-13-В-2 Оптимизирует код под ограниченные ресурсы ПК*-13-В-3 Работает с ОС, загрузчиками и аппаратурой	**Знать:** архитектуру ядра Linux; принципы работы встраиваемых операционных систем. **Уметь:** портировать код между различными вариантами загрузчиков и ОС; выполнять разработку на стыке программного

Код и наименование формируемых компетенций	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций
		обеспечения и оборудования. **Владеть:** инструментальными цепочками кросс-компиляции; методами отладки встраиваемых систем (трассировка, логические анализаторы); навыками разработки и отладки кода в среде RTOS; навыками разработки и выполнения тестовых сценариев.

4 Структура и содержание дисциплины

4.1 Структура дисциплины

Общая трудоемкость дисциплины составляет 6 зачетных единиц (216 академических часов).

Вид работы	Трудоемкость, академических часов	
	6 семестр	всего
Общая трудоёмкость	**216**	**216**
Контактная работа:	**105**	**105**
Лекции (Л)	34	34
Лабораторные работы (ЛР)	68	68
Консультации	1	1
Индивидуальная работа и инновационные формы учебных занятий	1,5	1,5
Промежуточная аттестация (зачет, экзамен)	0,5	0,5
Самостоятельная работа: *- выполнение курсового проекта (КП);* *- самоподготовка (проработка и повторение лекционного материала и материала учебников и учебных пособий;* *- изучение разделов курса в системе электронного обучения;* *- подготовка к лабораторным занятиям;* *- подготовка к рубежному контролю).*	**111** +	**111**
Вид итогового контроля (зачет, экзамен, дифференцированный зачет)	**экзамен**	

Разделы дисциплины, изучаемые в 6 семестре

№ раздела	Наименование разделов	Количество часов				
		всего	аудиторная работа			внеауд. работа
			Л	ПЗ	ЛР	
1	Введение в разработку встраиваемых систем. Архитектуры ARM и RISC-V	24	6		6	12

№ раздела	Наименование разделов	Количество часов				
		всего	аудиторная работа			внеауд. работа
			Л	ПЗ	ЛР	
2	Инструментальные цепочки и кросс-компиляция. Системы сборки (CMake, Make)	24	4		8	12
3	Программирование периферийных модулей: GPIO, UART, SPI, I2C	36	8		12	16
4	Прерывания и таймеры. Работа в режиме реального времени	30	6		10	14
5	Аналого-цифровые преобразователи (АЦП). Обработка аналоговых сигналов	24	4		8	12
6	Отладка встраиваемых систем: JTAG, GDB, трассировка, логические анализаторы	30	4		12	14
7	Операционные системы реального времени (RTOS) и Bare-metal разработка	48	2		12	34
	Итого:	216	34		68	114
	Всего:	216	34		68	114

4.2 Содержание разделов дисциплины

№ 1. Введение в разработку встраиваемых систем. Архитектуры ARM и RISC-V.

Понятие встраиваемой системы. Классификация встраиваемых систем, области применения. Особенности разработки ПО для встраиваемых систем: ограниченные ресурсы, работа с аппаратурой, реальное время. Архитектура ARM: история развития, семейства процессоров, основные характеристики. Архитектура RISC-V: особенности, преимущества, перспективы применения. Гарвардская и фон-неймановская архитектуры, сравнительный анализ. Система команд, регистры процессора, режимы адресации. Понятие системы на кристалле (SoC).

№ 2. Инструментальные цепочки и кросс-компиляция. Системы сборки (CMake, Make).

Кросс-компиляция: назначение, принципы, особенности. Инструментальная цепочка GCC для ARM и RISC-V: состав, назначение компонентов (препроцессор, компилятор, ассемблер, линкер). Системы сборки Make и CMake: синтаксис, примеры, сравнительный анализ. Линкер-скрипты: назначение, структура, настройка размещения кода и данных. Флаги оптимизации GCC и их влияние на размер и скорость исполняемого кода. Настройка среды разработки (VS Code, Eclipse) для работы с кросс-компилятором.

№ 3. Программирование периферийных модулей: GPIO, UART, SPI, I2C.

GPIO: режимы работы (вход, выход, альтернативная функция, аналоговый режим), настройка через регистры управления, подтягивающие резисторы. UART: принцип работы, параметры (скорость, чётность, стоп-биты), режимы передачи и приёма, настройка через регистры, работа с прерываниями. SPI: принцип работы, режимы (CPOL, CPHA), ведущий и ведомый, применение для связи с внешними устройствами. I2C: принцип работы, адресация, арбитраж, применение для связи с несколькими устройствами на одной шине. Прямая работа с регистрами управления периферией. Разработка драйверов периферийных модулей на языках C и ассемблера.

№ 4. Прерывания и таймеры. Работа в режиме реального времени.

Прерывания: определение, типы (внешние, внутренние, программные), приоритеты, вектор прерываний. Проектирование и реализация обработчиков прерываний. Проблемы при работе с прерываниями: гонки, вложенность, критические секции. Таймеры: режимы работы, генерация временных интервалов, широтно-импульсная модуляция (ШИМ), захват и сравнение. Системы реального времени: требования, детерминированность, временные ограничения. Программные таймеры на основе системного тика.

№ 5. Аналого-цифровые преобразователи (АЦП). Обработка аналоговых сигналов.

АЦП: основные характеристики (разрядность, скорость, точность, диапазон), методы преобразования (последовательного приближения, сигма-дельта и др.). Настройка АЦП в микроконтроллерах, работа с регистрами. Прямое управление АЦП, работа по прерыванию, использование DMA для разгрузки процессора. Методы фильтрации аналоговых сигналов:

усреднение, медианная фильтрация, фильтр низких частот. Калибровка АЦП. Обработка аналоговых сигналов в реальном времени.

№ 6. Отладка встраиваемых систем: JTAG, GDB, трассировка, логические анализаторы.

Методы и инструменты отладки встраиваемых систем. Внутрисхемные отладчики JTAG и SWD: назначение, принцип работы, применение. Программные отладчики: GDB, OpenOCD, настройка и использование. Трассировка (ETM/ETB), вывод через SWO. Логические анализаторы и анализаторы последовательных протоколов: применение для отладки SPI, I2C, UART. Точки останова и точки наблюдения. Отладка в условиях ограниченного доступа к аппаратуре. Методы поиска и исправления сложных ошибок: зависания, сбои прерываний, проблемы синхронизации.

№ 7. Операционные системы реального времени (RTOS) и Bare-metal разработка.

RTOS: назначение, основные понятия, области применения. Задачи и планировщик в RTOS. Приоритеты и алгоритмы планирования (вытесняющее, кооперативное). Механизмы синхронизации: семафоры, мьютексы, очереди сообщений, события. Примеры популярных RTOS. Bare-metal разработка: особенности, преимущества и недостатки. Сравнительный анализ RTOS и Bare-metal. Оптимизация кода под ограниченные ресурсы: размер кода, потребление памяти, энергопотребление. Тестирование встраиваемого ПО: виды тестов, методы верификации, тестовые сценарии. Стандарты кодирования для встраиваемых систем (MISRA C, CERT C). Безопасность встраиваемых систем.

4.3 Лабораторные работы

№ ЛР	№ раздела	Наименование лабораторных работ	Кол-во часов
1	1	Изучение архитектуры микроконтроллера и среды разработки. Настройка инструментальной цепочки кросс-компиляции	4
2	2	Сборка проекта с использованием Make и CMake. Анализ влияния флагов оптимизации на размер и скорость кода	4
3	3	Программирование GPIO	4
4	3	Реализация обмена данными по UART: передача и приём, работа с прерываниями	4
5	3	Работа с SPI: подключение и опрос внешнего устройства	4
6	3	Работа с I2C: обмен данными с несколькими устройствами на одной шине	4
7	4	Программирование таймеров: генерация временных интервалов и ШИМ	4
8	4	Реализация обработчиков прерываний: внешние прерывания и приоритеты	4
9	4	Программные таймеры на основе системного тика. Работа в режиме реального времени	4
10	5	Настройка АЦП и считывание аналоговых сигналов. Работа по прерыванию	4
11	5	Использование DMA для сбора данных с АЦП. Программная фильтрация сигналов	4
12	6	Отладка с использованием GDB и OpenOCD. Точки останова и наблюдения	4
13	6	Отладка протоколов SPI/I2C/UART с помощью логического анализатора	4
14	6	Трассировка и вывод отладочной информации через SWO. Поиск сложных ошибок	4
15	7	Разработка приложения с использованием FreeRTOS: задачи, очереди, семафоры	4
16	7	Синхронизация задач в RTOS. Сравнение RTOS и Bare-metal подходов	4
17	7	Оптимизация кода под ограниченные ресурсы. Разработка тестовых сценариев	4
		Итого:	68

4.4 Курсовой проект (6 семестр)

Примерные темы курсового проекта:

1. Система мониторинга температуры с выводом данных на дисплей и передачей по UART.
2. Система управления шаговым двигателем с использованием таймера и ШИМ.
3. Драйвер внешнего АЦП (например, MCP3208) с использованием интерфейса SPI.
4. Система сбора данных с нескольких датчиков с логированием на SD-карту.
5. Реализация протокола обмена данными между двумя микроконтроллерами по UART с проверкой целостности (CRC).
6. Приложение на FreeRTOS для управления несколькими задачами (сбор данных, обработка, вывод).
7. Система с низким энергопотреблением с использованием режимов сна микроконтроллера.
8. Эмулятор периферийного устройства (например, EEPROM) на микроконтроллере.
9. Устройство управления освещением с плавной регулировкой яркости (ШИМ) и интерфейсом пользователя.
10. Система беспроводной передачи данных с датчиков на базе ESP32.

5 Учебно-методическое обеспечение дисциплины

5.1 Основная литература

1. Таненбаум Э. Современные операционные системы = Modern Operating Systems [Текст] / Э. Таненбаум; [пер. А. Леонтьев]. - 2-е изд. - СПб. : Питер, 2007. - 1038 с. : ил. - (Классика computer science). - Парал. тит. л. англ. - Библиогр.: с. 989-1020. - Алф. указ.: с. 1021-1037. - ISBN 978-5-318-00299-1. - ISBN 5-318-00299-4.

2. Бройдо, В.Л. Архитектура ЭВМ и систем [Текст] : учеб. для вузов / В. Л. Бройдо, О. П. Ильина. - СПб. : Питер, 2006. - 718 с. : ил. - (Учебник для вузов). - Библиогр.: с. 709-712. - Алф. указ.: с. 713-717. - ISBN 5-469-00742-1. Столяров А. В. Программирование: введение в профессию. – М.: МАКС Пресс, 2020. – 512 с.

3. Конова, Е. А. Алгоритмы и программы. Язык С++ [Текст] : учебное пособие для студентов, обучающихся по направлению "Прикладная информатика" / Е. А. Конова, Г. А. Поллак.- 3-е изд., стер. - Санкт-Петербург : Лань, 2018. - 384 с. : ил. - (Учебники для вузов. Специальная литература). - Библиогр.: с. 382. - ISBN 978-5-8114-2020-9.

5.2 Дополнительная литература

1. Алгоритмы: построение и анализ = Introduction to Algorithms [Текст] / Т. Кормен [и др.]; [пер. с англ. И. В. Красикова, Н. А. Ореховой, В. Н. Романова; под ред. И. В. Красикова].- 2-е изд. - Москва ; Санкт-Петербург ; Киев : Вильямс, 2013. - 1296 с. : ил. - Парал. тит. л. англ. - Прил.: с. 1189-1256. - Библиогр.: с. 1257-1276. - Предм. указ.: с. 1277-1290. - ISBN 978-5-8459-0857-5. - ISBN 0-07-013151-1.

2. Гордеев, А. В. Операционные системы [Текст] : учебник / А. В. Гордеев.- 2-е изд. - Санкт Петербург : Питер, 2007. - 416 с. - (Учебник для вузов). - Библиогр.: с. 406-408. - Алф. указ.: с. 409-415. - ISBN 978-5-94723-632-3.

3. Финогенов, К. Г. Использование языка Ассемблера [Текст] : учеб. пособие / К. Г. Финогенов. - М. : Горячая линия-Телеком, 2004. - 438 с. : ил. - (Учебное пособие для вузов) - ISBN 5-93517-118-X.

5.3 Периодические издания

1. Программирование (ИПМ им. М.В. Келдыша РАН): журнал. - Москва: издательство «Наука».
2. Вестник компьютерных и информационных технологий: журнал. - Москва: ИД «Спектр».
3. Информационные технологии: журнал. - Москва: Агентство «Роспечать».

5.4 Интернет-ресурсы

− www.elibrary.ru - научная электронная библиотека. Журналы, книги, патенты;

− SCOPUS [Электронный ресурс]: реферативная база данных / компания Elsevier. – Режим доступа: https://www.scopus.com/, в локальной сети ОГУ.

5.5 Программное обеспечение, профессиональные базы данных и информационные справочные системы

1. Операционная система РЕД ОС.
2. Пакет офисных приложений «МойОфис Образование».
3. Университетская платформа электронного обучения «Электронные курсы ОГУ в системе обучения Moodle» (http://moodle.osu.ru).
4. Для работы с ресурсами Интернет - веб-браузер Яндекс https://yandex.ru/.
5. ГАРАНТ Платформа F1 [Электронный ресурс]: справочно-правовая система. / Разработчик ООО НПП «ГАРАНТ-Сервис», 119992, Москва, Воробьевы горы, МГУ, [1990–2026]. – Режим доступа в сети ОГУ http://garant.net.osu.ru
6. Автоматизированная интерактивная система сетевого тестирования - АИССТ (зарегистрирована в РОСПАТЕНТ, Свидетельство о государственной регистрации программы для ЭВМ №2011610456, правообладатель – Оренбургский государственный университет), режим доступа - http://aist.osu.ru.
7. Набор свободных компиляторов для различных языков программирования GCC (GNU Compiler Collection).
8. CMake – свободная кроссплатформенная система автоматизации сборки программного кода.
9. QEMU – свободная программа с открытым исходным кодом для эмуляции аппаратного обеспечения и виртуализации.
10. OpenOCD – программа с открытым исходным кодом для внутрисхемной отладки, программирования и тестирования микроконтроллеров, процессоров и FPGA.
11. Бесплатные инструменты для написания и редактирования компьютерного кода – VS Code, Eclipse.
12. ESP-IDF – программный каркас для создания прошивок на языках C и C++ для микроконтроллеров семейства ESP32 (свободно распространяемое).
13. STM32Cube – экосистема программного обеспечения для разработки проектов на 32-битных микроконтроллерах семейства STM32 (свободно распространяемое).

6 Материально-техническое обеспечение дисциплины

Учебные аудитории для проведения занятий лекционного типа, семинарского типа, курсового проектирования, для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации.

Аудитории оснащены комплектами ученической мебели, техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения лабораторных занятий используется компьютерный класс оснащенный: отладочными платами STM32, ESP32, отечественными аппаратными платформами; внутрисхемными отладчиками; логическими анализаторами; анализаторами последовательных протоколов; персональными компьютерами с установленным программным обеспечением (GCC для ARM/RISC-V, CMake, OpenOCD, VS Code, Eclipse, STM32Cube, ESP-IDF); комплектами ученической мебели; техническими средствами обучения для представления учебной информации.

Помещение для самостоятельной работы обучающихся оснащено компьютерной техникой, подключенной к сети «Интернет», и обеспечением доступа в электронную информационно-образовательную среду ОГУ.