

МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

**РАБОЧАЯ ПРОГРАММА
УЧЕБНОЙ ДИСЦИПЛИНЫ**

МДК.02.02 Программирование микроконтроллеров

программы подготовки специалистов среднего звена

09.02.01 Компьютерные системы и комплексы

Форма обучения: очная

Владивосток 2026

Рабочая программа учебной дисциплины МДК.02.02 Программирование микроконтроллеров профессионального модуля ПМ.02 Проектирование управляющих программ компьютерных систем и комплексов разработана на основе Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.01 Компьютерные системы и комплексы, утверждённого приказом Минпросвещения России от 25 мая 2022 г. № 362, примерной образовательной программой.

Разработчики: Д.И. Головин, преподаватель КСД ВВГУ

Рассмотрено и одобрено на заседании цикловой методической комиссии
Протокол № 9 от «18» 05 2026 г.

Председатель ЦМК  Е.А. Стефанович
подпись

Согласована:

Начальник отдела информационных технологий, Филиал Российской телевизионной и радиовещательной сети «Приморский краевой радиотелевизионный передающий центр»


(подпись, печать) Д.М. Шумов


СОДЕРЖАНИЕ

- 1 ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ
УЧЕБНОЙ ДИСЦИПЛИНЫ
- 2 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ
- 3 УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ
- 4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ
УЧЕБНОЙ ДИСЦИПЛИНЫ
- 5 ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

1. Общая характеристика рабочей программы учебной дисциплины

1.1 Место дисциплины в структуре основной образовательной программы

Рабочая программа учебной дисциплины МДК.02.02 Программирование микрконтроллеров является частью профессионального цикла основной образовательной программы (далее ООП) в соответствии с ФГОС СПО по специальности 09.02.01 Компьютерные системы и комплексы.

1.2 Цель и планируемые результаты освоения дисциплины

По итогам освоения дисциплины, обучающиеся должны продемонстрировать результаты обучения, соотнесённые с результатами освоения ООП СПО, приведенные в таблице:

Код ОК, ПК	Код результата обучения	Наименование результата обучения
ПК.2.1- ПК.2.5 ОК1-ОК9	У1	использовать методы и приемы формализации задач
	У2	использовать методы и приемы алгоритмизации поставленных задач
	У3	использовать программные продукты для графического отображения алгоритмов
	У4	применять стандартные алгоритмы в соответствующих областях
	У5	применять выбранные языки программирования для написания программного кода
	У6	использовать выбранную среду программирования и средства системы управления базами данных
	У7	использовать возможности имеющейся технической и/или программной архитектуры
	У8	применять нормативные документы, определяющие требования к оформлению программного кода
	У9	применять инструментарий для создания и актуализации исходных текстов программ
	У10	выявлять ошибки в программном коде
	У11	применять методы и приемы отладки программного кода
	У12	интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов
	У13	применять современные компиляторы, отладчики и оптимизаторы программного кода
	У14	документировать произведенные действия, выявленные проблемы и способы их устранения
	У15	проводить оценку работоспособности программного продукта
	У16	создавать резервные копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта и данных
	У17	использовать выбранную систему контроля версий
	У18	выполнять действия, соответствующие установленному ре-

		гламенту используемой системы контроля версий;
	Y19	интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов
	Y20	применять современные компиляторы, отладчики и оптимизаторы программного кода
	Y21	документировать произведенные действия, выявленные проблемы и способы их устранения
	Y22	создавать резервные копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта и данных
	Y23	выполнять процедуры сборки программных модулей и компонент в программный продукт
	Y24	производить настройки параметров программного продукта и осуществлять запуск процедур сборки;
	Y25	писать программный код процедур интеграции программных модулей
	Y26	использовать выбранную среду программирования для разработки процедур интеграции программных модулей
	Y27	применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов
	Y28	разрабатывать и оформлять контрольные примеры для проверки работоспособности программного обеспечения
	Y29	разрабатывать процедуры генерации тестовых наборов данных с заданными характеристиками
	Y30	подготавливать наборы данных, используемых в процессе проверки работоспособности программного обеспечения
	Y31	выявлять соответствие требований заказчиков к существующим продуктам;
	Y32	соблюдать процедуру установки прикладного программного обеспечения в соответствии с требованиями организации-производителя
	Y33	идентифицировать инциденты, возникающие при установке программного обеспечения, и принимать решение по изменению процедуры установки
	31	методы и приемы формализации и алгоритмизации задач
	32	языки формализации функциональных спецификаций
	33	нотации и программные продукты для графического отображения алгоритмов
	34	алгоритмы решения типовых задач, области и способы их применения
	35	синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования

	36	методологии разработки программного обеспечения
	37	методологии и технологии проектирования и использования баз данных
	38	технологии программирования
	39	особенности выбранной среды программирования и системы управления базами данных
	310	компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними
	311	инструментарий для создания и актуализации исходных текстов программ
	312	методы повышения читаемости программного кода
	313	системы кодировки символов, форматы хранения исходных текстов программ
	314	нормативные документы, определяющие требования к оформлению программного кода
	315	методы и приемы отладки программного кода
	316	типы и форматы сообщений об ошибках, предупреждений
	317	способы использования технологических журналов, форматы и типы записей журналов
	318	современные компиляторы, отладчики и оптимизаторы программного кода
	319	сообщения о состоянии аппаратных средств
	320	методы и средства верификации работоспособности выпусков программных продуктов
	321	языки, утилиты и среды программирования, средства пакетного выполнения процедур
	322	возможности используемой системы контроля версий и вспомогательных инструментальных программных средств
	323	установленный регламент использования системы контроля версий
	324	методы и средства сборки и интеграции программных модулей и компонент
	325	интерфейсы взаимодействия с внешней средой
	326	интерфейсы взаимодействия внутренних модулей системы
	327	методы и средства сборки модулей и компонент программно-го обеспечения
	328	методы и средства разработки процедур для развертывания программного обеспечения
	329	методы и средства миграции и преобразования данных
	330	методы создания и документирования контрольных примеров и тестовых наборов данных
	331	правила, алгоритмы и технологии создания тестовых наборов данных
	332	требования к структуре и форматам хранения тестовых наборов данных

	333	основные понятия в области качества программных продуктов
	334	лицензионные требования по настройке устанавливаемого программного обеспечения
	335	типовые причины инцидентов, возникающих при установке программного обеспечения
	336	основы архитектуры, устройства и функционирования вычислительных систем
	337	принципы организации, состав и схемы работы операционных систем
	338	стандарты информационного взаимодействия систем

2 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1 Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Объем образовательной программы учебной дисциплины	485
В том числе	
- теоретическое обучение	160
- практические занятия	146
- курсовое проектирование	48
- самостоятельная работа	91
- промежуточная аттестация (экзамен) 6, 8 семестры	12

1.1. Тематический план и содержание профессионального модуля (ПМ)

Наименование разделов и тем профессионального модуля (ПМ), междисциплинарных курсов (МДК)	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная учебная работа обучающихся, курсовая работа (проект)	Объем, ак. ч / в том числе в форме практической подготовки, ак. ч
1	2	3
Раздел 2. Программирование микроконтроллеров		485
МДК. 02.02. Программирование микроконтроллеров		485
Тема 2.1. Особенности программирования микроконтроллеров STM32/ATmega328	Содержание	30
	1. Принципы построения программ для микроконтроллеров. Средства разработки и отладки.	18
	2. Правила составления алгоритмов. Типы алгоритмов. Диаграммы состояний. Конечный автомат.	
	3. Особенности синтаксиса для программ на МК	
	В том числе практических занятий	12
	Практическая работа № 1. Изучение устройства микроконтроллера ATmega328	2
	Практическая работа № 2. Установка и настройка программного обеспечения. Подключение микроконтроллера к компьютеру	2
	Практическая работа № 3. Изучение основ программирования микроконтроллера на высокоуровневом языке (C/C++)	6

	Практическая работа № 4. Изучение первой программы по управлению вводом-выводом	2
Тема 2.2. Модульное программирование микроконтроллеров STM32	Содержание	166
	1 Библиотеки для платформы ARM Cortex (STM32). Низкоуровневые библиотеки для платформы ARM Cortex (STM32)	102
	2 Средства программирования и отладки. Конфигуратор STM32CubeMX. Средства разработки System Workbench for STM32. Средства разработки STM32CubeIDE. САПР Proteus Design Suite.	
	3 Программирование систем микроконтроллера. 3.1 Программирование памяти: программирование памяти микроконтроллеров; программирование Flash памяти STM32. 3.2 Программирование подсистем ввода-вывода: программирование подсистем ввода/вывода микроконтроллера через регистры CMSIS; программирование портов ввода/вывода через функции библиотеки HAL; программирование портов ввода/вывода через LL-функции. 3.3 Программирование системы прерываний: система прерываний; организация и управление прерываниями. 3.4 Программирование таймеров-счетчиков: таймеры в режиме счетчиков, генерация циклических прерываний от счетчиков; конфигурация таймеров с помощью библиотек CMSIS, HAL и LL. 3.5 Программирование последовательных интерфейсов: работа с интерфейсом UART через регистры CMSIS, использование прерывания UART; работа с UART через библиотеку HAL, инициализация интерфейса и передача данных в блокирующем режиме; работа с UART через библиотеку HAL, прием данных в блокирующем режиме; работа с UART через библиотеку HAL с использованием прерываний. 3.6 Программирование синхронных интерфейсов: программирование последовательных портов SPI, I ² C, USART. 3.7 Программирование прямого доступа к памяти DMA: прямой доступ к памяти в STM32, контроллер DMA; работа с контроллером DMA через CMSIS регистры; программный модуль контроллера DMA. 3.8 Программирование внешней памяти: подключение SD-карты к микроконтроллеру; файловая система FAT на внешней SD-карте; настройка FatFs в STM CubeM.	

	3.9 Программирование режима потребления микроконтроллера: режимы пониженного потребления, SLEEP. 3.10 Программирование АЦП/ЦАП: установка конфигурации АЦП через регистры CMSIS; работа с АЦП через регистры CMSIS, основные режимы преобразования; работа с АЦП в различных режимах; режим сдвоенных преобразований АЦП STM32; работа АЦП в режиме оконного компаратора; работа с АЦП через функции библиотеки HAL. 3.11 Программирование интерфейса USB: USB HID (human interface device) – класс устройств для взаимодействия с человеком.	
	В том числе практических занятий	64
	Практическая работа № 5. Создание проекта программы в STM32CubeIDE	2
	Практическая работа № 6. Структура проекта в STM32CubeIDE	2
	Практическая работа № 7. Перспектива Device Configuration Tool в STM32CubeIDE	2
	Практическая работа № 8. Конфигурация рабочего окна в STM32CubeIDE	2
	Практическая работа № 9. Перспектива C/C++ в STM32CubeIDE	2
	Практическая работа № 10. Структура файла main.c в STM32CubeIDE	2
	Практическая работа № 11. Изучение функции Hal_GPIO_TogglePin	2
	Практическая работа № 12. Изучение функции HAL_GPIO_WritePin	2
	Практическая работа № 13. Обработка дребезга контактов кнопки. Изучение интерфейса связи между программными блоками	2
	Практическая работа № 14. Генерация управляющих импульсных сигналов	2
	Практическая работа № 15. Изучение классов в программах микроконтроллера. Кнопка как объект	2
	Практическая работа № 16. Программирование цифровой фильтрации сигналов микроконтроллера	2
	Практическая работа № 17. Создание библиотеки для программирования микроконтроллера	2

	Практическая работа № 18. Изучение прерывания по таймеру микроконтроллера. Параллельные процессы	2
	Практическая работа № 19. Изучение программных таймеров в микроконтроллерах	2
	Практическая работа № 20. Изучение порта UART. Библиотека Serial. Отладка программ	2
	Практическая работа № 21. Программирование аналоговых входов микроконтроллера	4
	Практическая работа № 22. Программирование EEPROM микроконтроллера. Контроль целостности данных	4
	Практическая работа № 23. Указатели C в микроконтроллерах. Преобразование разных типов данных в байты	4
	Практическая работа № 24. Повышение надежности программ для микроконтроллера. Сторожевой таймер	4
	Практическая работа № 25. Изучение процесса обмена данными между платами микроконтроллера	4
	Практическая работа № 26. Программирование обмена данными между платой микроконтроллера и компьютером через интерфейс UART	4
	Практическая работа № 27. Обмен данными между платами микроконтроллеров через интерфейс UART	4
	Практическая работа № 28. Помехоустойчивость и физическая среда стандартного интерфейса UART	4
Тема 2.3. Автоматизация процессов на основе систем с микроконтроллерами STM32	Содержание	110
	1 Направления автоматизации процессов. Структура управляющих систем. Основы создания алгоритмов и программ для взаимодействия систем на основе МК с пользователем. Особенности работы с внешними устройствами. Взаимодействие систем на основе МК по телекоммуникационным сетям.	40
	2 Операционные системы реального времени	
	В том числе практических занятий	70
	Практическая работа № 29. Программирование генератора меандра	2

	Практическая работа № 30. Программирование генератора сигнала SOS	2
	Практическая работа № 31. Программирование генератора сигнала SOS со звуковым сопровождением	2
	Практическая работа № 32. Программирование линейного обработчика светофора	2
	Практическая работа № 33. Программирование обработчика светофора со стрелкой	2
	Практическая работа № 34. Программирование обработчика светофора с пешеходными секциями в автоматном режиме	2
	Практическая работа № 35. Программирование семисегментного индикатора	2
	Практическая работа № 36. Программирование обработчика семисегментного индикатора	2
	Практическая работа № 37. Программирование обратного таймера 5 секунд	2
	Практическая работа № 38. Программирование таймера 10 секунд	2
	Практическая работа № 39. Программирование бегущего огня	2
	Практическая работа № 40. Программирование индикатора состояния	2
	Практическая работа № 41. Программирование таймера 1 минута	2
	Практическая работа № 42. Программирование светофора с таймером обратного отсчета	4
	Практическая работа № 43. Программирование тактовых кнопок	4
	Практическая работа № 44. Программирование телеграфного манипулятора	4

	Практическая работа № 45. Программирование вызывного светофора	4
	Практическая работа № 46. Программирование ямбического ключа	4
	Практическая работа № 47. Программирование счетчика внешних событий	4
	Практическая работа № 48. Программирование индикатора переключения режимов АКПП	4
	Практическая работа № 49. Программирование радиальных интерфейсов RS-232 и RS-242	4
	Практическая работа № 50. Программирование интерфейса ИРПС (цифровой токовой петли)	4
	Практическая работа № 51. Программирование последовательной передачи данных ModBus RTU	4
	Практическая работа № 52. Обмен данными между контроллером и компьютером через UART по протоколу ModBus	4
Курсовой проект (работа) <i>Выполнение курсового проекта (работы) по модулю является обязательным.</i> Тематика курсовых проектов (работ) Система контроля температуры на основе МК Система ограничения скорости автомобиля на основе МК Система трекинга автомобиля на основе МК Система учета электроэнергии на основе МК Система пожаробезопасности и обнаружения газов в помещении на основе МК Разработка программы управления на микроконтроллере для системы контроля допуска в здание Разработка программы управления на микроконтроллере для управляющей системы охлаждения ПК Разработка программы управления на микроконтроллере для калькулятора Разработка программы управления на микроконтроллере для часов Разработка программы управления на микроконтроллере для цифровой клавиатура для ПК Разработка программы управления на микроконтроллере для системы проверки кабеля типа витая пара		48

Разработка программы управления на микроконтроллере для системы вывода изображений на светодиодную матрицу Разработка программы управления на микроконтроллере для системы включения и выключения света по звуковому сигналу Разработка программы управления на микроконтроллере для системы включения и выключения света в помещении, по введенному графику. Разработка программы управления на микроконтроллере для системы поддержания равновесия в полете для квадрокоптера Разработка программы управления на микроконтроллере для системы управления коммуникациями частного домовладения Разработка программы управления на микроконтроллере для системы пульта управления Разработка программы управления на микроконтроллере для подвижного робота, с автопарковкой Разработка программы управления на микроконтроллере для системы зарядки и индикации аккумуляторных батарей Разработка программы управления на микроконтроллере для измерения скорости ветра на улице и ее индикации Разработка программы управления на микроконтроллере для цифрового амперметра Разработка программы управления на микроконтроллере для тахометра Разработка программы управления на микроконтроллере для телефонной сети из трех абонентов Разработка программы управления на микроконтроллере для автомобильной сигнализации Разработка программы управления на микроконтроллере для проигрывателя рингтонов Разработка программы управления на микроконтроллере для дистанционного инфракрасного управления Разработка программы управления на микроконтроллере для сигнализации в холодильной установке Разработка программы управления на микроконтроллере для сетевой метеостанции Разработка программы управления на микроконтроллере для создание игровой приставки «тетрис» Разработка программы управления на микроконтроллере для создания светодиодной RGB матрицы, с выводом на нее изображения Разработка программы управления на микроконтроллере для системы контроля доступа на основе RFID Разработка программы управления на микроконтроллере для системы управления роботом через Bluetooth Разработка программы управления на микроконтроллере для считывания и записи показаний датчиков для создания массива данных. Разработка программы управления на микроконтроллере для считывания команд радиопульта управления Разработка программы управления на микроконтроллере для управления миро-робота паука	
---	--

Обязательные аудиторные учебные занятия по курсовому проекту (работе)	48
Самостоятельная учебная работа обучающегося над курсовым проектом (работой): Систематическая проработка конспектов занятий, учебной и специальной технической литературы (по вопросам к параграфам, главам учебных пособий, составленным преподавателем). Подготовка к лабораторным и практическим работам с использованием методических рекомендаций преподавателя, оформление лабораторно-практических работ, отчетов и подготовка к их защите. Самостоятельное изучение правил выполнения чертежей и технологической документации по ЕСКД и ЕСТД.	91
Консультации по МДК. 02.01. Микропроцессорные системы	28
Промежуточная аттестация	12
Итого	485

3 УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

3.1 Материально-техническое обеспечение

Для реализации программы профессионального модуля должны быть предусмотрены следующие специальные помещения: Оборудование учебных кабинетов и рабочих мест кабинетов:

Оборудование лаборатории и рабочих мест **Проектирования цифровых устройств:**

количество посадочных мест – 25, стол для преподавателя 1 шт., стул для преподавателя 1 шт., шкаф 3 шт., компьютерный стол 20 шт., персональный компьютер ПК i3 2120/500Gb/4Gb 20 шт., мультимедийный комплект: проектор, интерактивная доска Elite Panaboard UBT-T880W 1 шт., колонки ОКЛИК 1 шт., доска маркерная меловая комбинированная 1 шт., информационный стенд 2 шт., сервер (процессор-i7-6700 (4 ядра, 3.4Ghz, L3 8 Mb), оперативная память-32Gb; накопитель-HDD 5 Tb) 1 шт., типовой состав для монтажа и наладки компьютерной сети: набор для обжима и тестирования кабеля UTP, кабель UTP - 305м 1шт., коннекторы 8P8C, 12 шт., коммутатор: CISCO WS-C2960-24TT-L 1 шт., блок бесперебойного питания, фильтр: APC 2200 1 шт., фильтр 6 розеток 6 шт., дидактические пособия.

ПО: 1. Windows 10 (профессиональная лицензия, ООО "Битроникс Владивосток" Контракт№ 0320100030814000018-45081 от 09.09.14 № 48609744, №62096196, № 48958910, № 45829305, бессрочно);

2. MS Office 2010 pro (лицензия № 48958910, № 47774898 , бессрочно);
3. visual c++ 2008 express edition (свободное),
4. oracle vm virtualbox (свободное),
5. cisco packet tracer (свободное),
6. micosoft SQL server 2008 (свободное),
7. k-lite codec pack (свободное),
8. visual studio 2008 (свободное),
9. Google Chrome (свободное);
10. Internet Explorer (свободное)

Лаборатория «Прикладного программирования»

1. автоматизированные рабочие места на 20 обучающихся (персональный компьютер Lenovo ThinkStation P330 Tiny (процессор I7 9700T, оперативная память 32gb, жесткий диск 1tb) монитор Lenovo ThinkCentre TIO27 2560x1440 (веб камера, микрофон, динамики встроены в монитор), мышь, клавиатура;

2. автоматизированное рабочее место преподавателя (ноутбук Lenovo IdealPad L340, процессор i7 9750H, оперативная память 8gb, жесткий диск 1tb), мышь, клавиатура;

3. специальная эргономическая мебель для работы за компьютером: компьютерный стол 20шт., компьютерное кресло 20 шт., стол преподавателя 1 шт., компьютерное кресло для преподавателя 1шт., 1 проектор CASIO XJ-F210WN 1 шт., экран 1 шт., принтер МФУ Xerox VersaLink C7020, цветной 1 шт., графический планшет WACOM Cintiq 16-10 шт.

ПО:

1. Windows 10 Pro, ИП Струлев О.Ю. Д№32008976244 от 06.04.2020, OEM

2. OfficeProfessional Plus 2019 AcademicEdition, ООО "Акцент", Договор №292 от 24.04.2020 лицензия №V6635206 от 07.05.2020, бессрочно;
3. Adobe Creative Cloud, ООО"ИНФОРМИКА", Договор №32008982727 от 16.04.2020, лицензия от 19.04.2020;
4. Corona Render for 3ds Max-Educational-1WS+NODE, ООО"ИНФОРМИКА", Договор №32008982727 от 16.04.2020, лицензия от 08.04.2020;
5. Google Chrome (свободное);
6. Internet Explorer (свободное)
7. проектор, экран/маркерная доска.

3.2 Информационное обеспечение обучения

Перечень используемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники:

1. Вязовик, Н. А. Программирование на Java : учебное пособие для СПО / Н. А. Вязовик. — Саратов : Профобразование, 2019. — 604 с. — ISBN 978-5-4488-0365-9. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROобразование : [сайт]. — URL: <https://profspo.ru/books/86206> (дата обращения: 22.12.2021).
2. Гуров, В. В. Микропроцессорные системы : учебник / В.В. Гуров. — Москва : ИНФРА-М, 2022. — 336 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-16-015323-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1843024> (дата обращения: 09.12.2021). — Режим доступа: по подписке.
3. Огнева, М. В. Программирование на языке C++: практический курс : учебное пособие для среднего профессионального образования / М. В. Огнева, Е. В. Кудрина. — Москва : Издательство Юрайт, 2021. — 335 с. — (Профессиональное образование). — ISBN 978-5-534-05780-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/473118>.
4. Соколова, В. В. Разработка мобильных приложений : учебное пособие для среднего профессионального образования / В. В. Соколова. — Москва : Издательство Юрайт, 2019. — 175 с. — (Профессиональное образование). — ISBN 978-5-534-10680-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/431172> (дата обращения: 22.12.2021).
5. Смирнов, Ю. А. Технические средства автоматизации и управления : учебное пособие для спо / Ю. А. Смирнов. — Санкт-Петербург : Лань, 2021. — 456 с. — ISBN 9785-8114-6712-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/151692> — Режим доступа: для авториз. пользователей.
6. Белугина, С. В. Разработка программных модулей программного обеспечения для компьютерных систем. Прикладное программирование : учебное пособие для спо / С. В. Белугина. — 3-е изд., стер. — Санкт-Петербург : Лань, 2022. — 312 с. — ISBN 978-58114-9817-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL:

<https://e.lanbook.com/book/200390> (дата обращения: 18.07.2022). — Режим доступа: для авториз. пользователей.

7. Зубкова, Т. М. Технология разработки программного обеспечения : учебное пособие для спо / Т. М. Зубкова. — 2-е изд., стер. — Санкт-Петербург : Лань, 2022. — 252 с. — ISBN 978-5-8114-9556-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/200462>. — Режим доступа: для авториз. пользователей.

Дополнительные источники:

1. Исаченко, О. В. Программное обеспечение компьютерных сетей [Электронный ресурс]: учебное пособие / О.В. Исаченко. — 2-е изд., испр. и доп. — М.: ИНФРА-М, 2021. — 158 с. - Режим доступа: <https://znanium.com/catalog/product/1189344> 2. Комиссаров, Ю. А. Общая электротехника и электроника : учебник / Ю.А. Комиссаров, Г.И. Бабокин, П.Д. Саркисова ; под ред. П.Д. Саркисова. — 2-е изд., испр. и доп. — Москва : ИНФРА-М, 2022. — 479 с. — (Высшее образование: Бакалавриат). — DOI 10.12737/13474. - ISBN 978-5-16-010416-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1853549> (дата обращения: 09.12.2021). — Режим доступа: по подписке.

3. Кузин, А. В. Компьютерные сети [Электронный ресурс]: учебное пособие / А. В. Кузин, Д. А. Кузин. — 4-е изд., перераб. и доп. — М.: ФОРУМ: ИНФРА-М, 2020. — 190 с. — Режим доступа: <https://znanium.com/catalog/product/1088380>

4. Максимов, Н. В. Компьютерные сети [Электронный ресурс]: учебное пособие / Н. В. Максимов, И. И. Попов. — 6-е изд., перераб. и доп. — М.: ФОРУМ: ИНФРА-М, 2021. — 464 с. — Режим доступа: <https://znanium.com/catalog/product/1189333>

5. Шаньгин, В. Ф. Информационная безопасность компьютерных систем и сетей [Электронный ресурс]: учебное пособие / В.Ф. Шаньгин. — М.: ФОРУМ: ИНФРА-М, 2021. — 416 с. — Режим доступа: <https://znanium.com/catalog/product/1189327>

4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результаты обучения	Критерии оценки	Методы оценки
У1 - использовать методы и приемы формализации задач	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У2 - использовать методы и приемы алгоритмизации поставленных задач	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ

У3 - использовать программные продукты для графического отображения алгоритмов	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У4 - применять стандартные алгоритмы в соответствующих областях	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У5 - применять выбранные языки программирования для написания программного кода	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У6 - использовать выбранную среду программирования и средства системы управления базами данных	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У7 - использовать возможности имеющейся технической и/или программной архитектуры	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У8 - применять нормативные документы, определяющие требования к оформлению программного кода	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У9 - применять инструментальный для создания и актуализации исходных текстов программ	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У10 - выявлять ошибки в программном коде	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки	оценка выполнения и защиты практических работ

	ки к выполнению заданий по практическим работам	
У11 - применять методы и приемы отладки программного кода	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У12 - интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У13 - применять современные компиляторы, отладчики и оптимизаторы программного кода	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У14 - документировать произведенные действия, выявленные проблемы и способы их устранения	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У15 - проводить оценку работоспособности программного продукта	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У16 - создавать резервные копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта и данных	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У17 - использовать выбранную систему контроля версий	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У18 - выполнять действия, соответствующие установ-	- правильность выполнения практических работ;	оценка выполнения и защиты практических работ

ленному регламенту используемой системы контроля версий	- правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	
У19 - интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У20 - применять современные компиляторы, отладчики и оптимизаторы программного кода	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У21 - документировать произведенные действия, выявленные проблемы и способы их устранения	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У22 - создавать резервные копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта и данных	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У23 - выполнять процедуры сборки программных модулей и компонент в программный продукт	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У24 - производить настройки параметров программного продукта и осуществлять запуск процедур сборки	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У25 - писать программный код процедур интеграции программных модулей	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ

У26 - использовать выбранную среду программирования для разработки процедур интеграции программных модулей	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У27 - применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У28 - разрабатывать и оформлять контрольные примеры для проверки работоспособности программного обеспечения	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У29 - разрабатывать процедуры генерации тестовых наборов данных с заданными характеристиками	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У30 - подготавливать наборы данных, используемых в процессе проверки работоспособности программного обеспечения	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У31 - выявлять соответствие требований заказчиков к существующим продуктам	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У32 - соблюдать процедуру установки прикладного программного обеспечения в соответствии с требованиями организации-производителя	- правильность выполнения практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	оценка выполнения и защиты практических работ
У33 - идентифицировать ин-	- правильность выполнения	оценка выполнения и защиты

циденты, возникающие при установке программного обеспечения, и принимать решение по изменению процедуры установки	практических работ; - правильность, полнота самостоятельной подготовки к выполнению заданий по практическим работам	ты практических работ
31 - методы и приемы формализации и алгоритмизации задач	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
32 - языки формализации функциональных спецификаций	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
33 - нотации и программные продукты для графического отображения алгоритмов	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
34 - алгоритмы решения типовых задач, области и способы их применения	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
35 - синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
36 - методологии разработ-	- четкость изложения мате-	- устный опрос;

ки программного обеспечения	риала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- тестовые задания
37 - методологии и технологии проектирования и использования баз данных	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
38 - технологии программирования	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
39 - особенности выбранной среды программирования и системы управления базами данных	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
310 - компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
311 - инструментарий для создания и актуализации исходных текстов программ	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и заче-	- устный опрос; - тестовые задания

	ту	
312 - методы повышения читаемости программного кода	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
313 - системы кодировки символов, форматы хранения исходных текстов программ	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
314 - нормативные документы, определяющие требования к оформлению программного кода	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
315 - методы и приемы отладки программного кода	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
316 - типы и форматы сообщений об ошибках, предупреждений	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
317 - способы использования технологических журналов, форматы и типы записей журналов	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота	- устный опрос; - тестовые задания

	самостоятельной подготовки к устному опросу и зачету	
318 - современные компиляторы, отладчики и оптимизаторы программного кода	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
319 - сообщения о состоянии аппаратных средств	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
320 - методы и средства верификации работоспособности выпусков программных продуктов	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
321 - языки, утилиты и среды программирования, средства пакетного выполнения процедур	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
322 - возможности используемой системы контроля версий и вспомогательных инструментальных программных средств	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
323 - установленный регламент использования системы контроля версий	- четкость изложения материала при устном опросе; - правильность выполнения	- устный опрос; - тестовые задания

	теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	
324 - методы и средства сборки и интеграции программных модулей и компонент	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
325 - интерфейсы взаимодействия с внешней средой	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
326 - интерфейсы взаимодействия внутренних модулей системы	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
327 - методы и средства сборки модулей и компонент программного обеспечения	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
328 - методы и средства разработки процедур для развертывания программного обеспечения	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
329 - методы и средства ми-	- четкость изложения мате-	- устный опрос;

грации и преобразования данных	риала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- тестовые задания
330 - методы создания и документирования контрольных примеров и тестовых наборов данных	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
331 - правила, алгоритмы и технологии создания тестовых наборов данных	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
332 - требования к структуре и форматам хранения тестовых наборов данных	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
333 - основные понятия в области качества программных продуктов	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
334 - лицензионные требования по настройке устанавливаемого программного обеспечения	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и заче-	- устный опрос; - тестовые задания

	ту	
335 - типовые причины инцидентов, возникающих при установке программного обеспечения	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
336 - основы архитектуры, устройства и функционирования вычислительных систем	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
337 - принципы организации, состав и схемы работы операционных систем	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания
338 - стандарты информационного взаимодействия систем	- четкость изложения материала при устном опросе; - правильность выполнения теста - правильность, полнота самостоятельной подготовки к устному опросу и зачету	- устный опрос; - тестовые задания

МИНОБРАЗОВАНИЯ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
для проведения текущего контроля и промежуточной аттестации
по учебной дисциплине

МДК.02.02 Программирование микроконтроллеров

программы подготовки специалистов среднего звена

09.02.01 Компьютерные системы и комплексы

Форма обучения: очная

Владивосток 2026

Фонд оценочных средств для проведения текущего контроля и промежуточной аттестации по МДК.02.02 Программирование микроконтроллеров разработан в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.01 Компьютерные системы и комплексы, утверждённого приказом Минпросвещения России от 25 мая 2022 г. № 362, примерной образовательной программой.

Разработчик: Д.И. Головин, преподаватель КСД ВВГУ

Рассмотрено и одобрено на заседании цикловой методической комиссии

Протокол № 9 от « 18 » 05 2026 г.

Председатель ЦМК  Е.А. Стефанович

1 Общие положения

Фонд оценочных средств (далее – ФОС) предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины 09.02.01 Разработка прикладных приложений.

ФОС включают в себя контрольные материалы для проведения текущего контроля успеваемости в виде устного собеседования или теста и промежуточной аттестации по дисциплине, которая проводится в форме дифференцированного зачета (с использованием оценочного средства - опрос в форме ответов на тест).

2 Планируемые результаты обучения по дисциплине, обеспечивающие результаты освоения образовательной программы

Код ОК, ПК	Код результата обучения	Наименование результата обучения
ПК 2.1, ОК 1	У1	использовать методы и приемы формализации задач
ПК 2.1, ОК 1	У2	использовать методы и приемы алгоритмизации поставленных задач
ПК 2.1, ОК 1	У3	использовать программные продукты для графического отображения алгоритмов
ПК 2.1, ОК 1, 2	У4	применять стандартные алгоритмы в соответствующих областях
ПК 2.1, ОК 1, 2	У5	применять выбранные языки программирования для написания программного кода
ПК 2.1, ОК 1, 2	У6	использовать выбранную среду программирования и средства системы управления базами данных
ПК 2.1, 2.2, ОК 1, 2	У7	использовать возможности имеющейся технической и/или программной архитектуры
ПК 2.1, 2.2, ОК 1, 9	У8	применять нормативные документы, определяющие требования к оформлению программного кода
ПК 2.1, 2.2, ОК 1, 9	У9	применять инструментарий для создания и актуализации исходных текстов программ
ПК 2.1, 2.4, ОК 1, 9	У10	выявлять ошибки в программном коде
ПК 2.1, 2.4, ОК 1, 9	У11	применять методы и приемы отладки программного кода
ПК 2.1, 2.4, ОК 1, 9	У12	интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов
ПК 2.1, 2.4, ОК 1, 9	У13	применять современные компиляторы, отладчики и оптимизаторы программного кода
ПК 2.1, 2.4, ОК 1, 9	У14	документировать произведенные действия, выявленные проблемы и способы их устранения
ПК 2.1, 2.4, ОК 1, 9	У15	проводить оценку работоспособности программного продукта
ПК 2.3, 2.5, ОК 1, 9	У16	создавать резервные копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта и данных
ПК 2.3, 2.4, ОК 1, 2	У17	использовать выбранную систему контроля версий

ПК 2.4, 2.5, ОК 1, 2	У18	выполнять действия, соответствующие установленному регламенту используемой системы контроля версий;
ПК 2.1, 2.4, ОК 1, 9	У19	интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов
ПК 2.2, 2.4, ОК 2	У20	применять современные компиляторы, отладчики и оптимизаторы программного кода
ПК 1.2, 1.4, ОК 2	У21	документировать произведенные действия, выявленные проблемы и способы их устранения
ПК 1.2, 1.4, ОК 2	У22	создавать резервные копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта и данных
ПК 2.3, 2.5, ОК 1, 2	У23	выполнять процедуры сборки программных модулей и компонент в программный продукт
ПК 2.3, 2.5, ОК 1, 2	У24	производить настройки параметров программного продукта и осуществлять запуск процедур сборки;
ПК 2.3, 2.5, ОК 1, 2	У25	писать программный код процедур интеграции программных модулей
ПК 2.1, 2.5, ОК 1, 2	У26	использовать выбранную среду программирования для разработки процедур интеграции программных модулей
ПК 2.1 - 2.5, ОК 1, 2	У27	применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов
ПК.2.1 - 2.5, ОК 1,2	У28	разрабатывать и оформлять контрольные примеры для проверки работоспособности программного обеспечения
ПК.2.4 - 2.5, ОК 1, 2	У29	разрабатывать процедуры генерации тестовых наборов данных с заданными характеристиками
ПК.2.4 - 2.5, ОК 1, 2	У30	подготавливать наборы данных, используемых в процессе проверки работоспособности программного обеспечения
ПК.2.4 - 2.5, ОК 1, 2	У31	выявлять соответствие требований заказчиков к существующим продуктам;
ПК.2.4 - 2.5, ОК 1, 2	У32	соблюдать процедуру установки прикладного программного обеспечения в соответствии с требованиями организации-производителя
ПК.2.4 - 2.5, ОК 1, 2	У33	идентифицировать инциденты, возникающие при установке программного обеспечения, и принимать решение по изменению процедуры установки
ПК 2.1, 2.2, ОК 1, 9	31	методы и приемы формализации и алгоритмизации задач
ПК 2.1, 2.2, ОК 1, 9	32	языки формализации функциональных спецификаций
ПК 2.1, 2.2, ОК 1, 9	33	нотации и программные продукты для графического отображения алгоритмов
ПК 2.1, 2.2, ОК 1, 9	34	алгоритмы решения типовых задач, области и способы их применения
ПК 2.1, 2.2, ОК 1, 9	35	синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные

		библиотеки языка программирования
ПК 2.1, 2.2, ОК 1, 9	36	методологии разработки программного обеспечения
ПК 2.1, 2.2, ОК 1, 9	37	методологии и технологии проектирования и использования баз данных
ПК 2.1, 2.2, ОК 1, 9	38	технологии программирования
ПК 2.1, 2.2, ОК 1, 9	39	особенности выбранной среды программирования и системы управления базами данных
ПК 2.1- 2.3, ОК 1, 9	310	компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними
ПК 2.1- 2.3, ОК 1, 9	311	инструментарий для создания и актуализации исходных текстов программ
ПК 2.1- 2.3, ОК 1, 9	312	методы повышения читаемости программного кода
ПК 2.1- 2.3, ОК 1, 9	313	системы кодировки символов, форматы хранения исходных текстов программ
ПК 2.1- 2.3, ОК 1, 9	314	нормативные документы, определяющие требования к оформлению программного кода
ПК 2.4, ОК 1, 2, 9	315	методы и приемы отладки программного кода
ПК 2.4, ОК 1, 2, 9	316	типы и форматы сообщений об ошибках, предупреждений
ПК 2.4, ОК 1, 2, 9	317	способы использования технологических журналов, форматы и типы записей журналов
ПК 2.1 – 2.5, ОК 1, 2, 9	318	современные компиляторы, отладчики и оптимизаторы программного кода
ПК 2.1 – 2.5, ОК 1, 2, 9	319	сообщения о состоянии аппаратных средств
ПК 2.4, ОК 1, 2, 9	320	методы и средства верификации работоспособности выпусков программных продуктов
ПК 2.1 – 2.5, ОК 1, 2, 9	321	языки, утилиты и среды программирования, средства пакетного выполнения процедур
ПК 2.4, ОК 1, 2, 9	322	возможности используемой системы контроля версий и вспомогательных инструментальных программных средств
ПК 2.4, ОК 1, 2, 9	323	установленный регламент использования системы контроля версий
ПК 2.1 – 2.5, ОК 1, 2, 9	324	методы и средства сборки и интеграции программных модулей и компонент
ПК.2.3 - 2.5, ОК 1, 2, 9	325	интерфейсы взаимодействия с внешней средой
ПК.2.3 - 2.5, ОК 1, 2, 9	326	интерфейсы взаимодействия внутренних модулей системы
ПК.2.3 - 2.5, ОК 1, 2, 9	327	методы и средства сборки модулей и компонент программного обеспечения
ПК 2.1 – 2.5, ОК 1, 2, 9	328	методы и средства разработки процедур для развертывания программного обеспечения
ПК.2.5, ОК 1, 2, 9	329	методы и средства миграции и преобразования данных
ПК.2.4 ОК 1, 2, 9	330	методы создания и документирования контрольных примеров и тестовых наборов данных

ПК.2.4 ОК 1, 2, 9	331	правила, алгоритмы и технологии создания тестовых наборов данных
ПК.2.4 ОК 1, 2, 9	332	требования к структуре и форматам хранения тестовых наборов данных
ПК.2.4 ОК 1, 2, 9	333	основные понятия в области качества программных продуктов
ПК.2.4 ОК 1, 2, 9	334	лицензионные требования по настройке устанавливаемого программного обеспечения
ПК.2.4 ОК 1, 2, 9	335	типовые причины инцидентов, возникающих при установке программного обеспечения
ПК 2.1 – 2.5, ОК 1, 2, 9	336	основы архитектуры, устройства и функционирования вычислительных систем
ПК 2.1 – 2.5, ОК 1, 2, 9	337	принципы организации, состав и схемы работы операционных систем
ПК 2.1 – 2.5, ОК 1, 2, 9	338	стандарты информационного взаимодействия систем

3 Соответствие оценочных средств контролируемым результатам обучения

3.1 Средства, применяемые для оценки уровня теоретической подготовки

Краткое наименование раздела (МДК), темы, параграфа дисциплины	Код результата обучения	Показатель овладения результатами обучения	Наименование оценочного средства и представление его в КОС	
			Текущий контроль	Промежуточная аттестация
Тема 1 Особенности программирования микроконтроллеров STM32/ATmega328: - Принципы построения программ для микроконтроллеров. Средства разработки и отладки; - Правила составления алгоритмов. Типы алгоритмов. Диаграммы состояний. Конечный автомат; - Особенности синтаксиса для программ на МК. Тема 2 Модульное программирование микроконтроллеров STM32: 2.1 Библиотеки для платформы ARM Cortex (STM32). Низкоуровневые библиотеки для платформы ARM Cortex (STM32). 2.2 Средства программирования и отладки. Конфигуратор STM32CubeMX. Средства разработки System Workbench for STM32. Средства разработки STM32CubeIDE.	31	Способность использовать знания методов и приемов формализации и алгоритмизации задач при программировании микроконтроллеров.	<i>Устные ответы на вопросы: 1 - 162. (п. 5.1)</i> <i>Выполнение реферата (п.5.2)</i>	<i>Ответы на вопросы тестов 1- 4 (п. 5.3).</i> <i>Ответы на экзаменационные билеты (п.6.1)</i>
	32	Способность использовать знания языков формализации функциональных спецификаций при программировании микроконтроллеров.		
	33	Способность использовать знания нотаций и программных продуктов для графического отображения алгоритмов при программировании микроконтроллеров.		
	34	Способность использовать знания алгоритмов решения типовых задач, областей и способов их применения при программировании микроконтроллеров.		
	35	Способность использовать знания синтаксиса выбранного языка программирования, особенностей программирования на этом языке, стандартных библиотек языка програм-		

САПР Proteus Design Suite. 2.3 Программирование систем микроконтроллера: - Программирование памяти: программирование памяти микроконтроллеров; программирование Flash памяти STM32; - Программирование подсистем ввода-вывода: программирование подсистем ввода/вывода микроконтроллера через регистры CMSIS; программирование портов ввода/вывода через функции библиотеки HAL; программирование портов ввода/вывода через LL-функции; - Программирование системы прерываний: система прерываний; организация и управление прерываниями; - Программирование таймеров-счетчиков: таймеры в режиме счетчиков, генерация циклических прерываний от счетчиков; конфигурация таймеров с помощью библиотек CMSIS, HAL и LL; - Программирование последовательных интерфейсов: работа с интерфейсом UART через регистры CMSIS, использование прерывания UART; работа с UART через библиотеку HAL, инициализация интерфейса и передача данных в блокирующем режиме; работа с UART через		мирования при программировании микроконтроллеров.		
	36	Способность использовать знания методологий разработки программного обеспечения при программировании микроконтроллеров.		
	37	Способность использовать знания методологий и технологий проектирования и использования баз данных при программировании микроконтроллеров.		
	38	Способность использовать знания технологий программирования при программировании микроконтроллеров.		
	39	Способность использовать знания особенностей выбранной среды программирования и системы управления базами данных при программировании микроконтроллеров.		
	310	Способность использовать знания компонентов программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними при программировании микроконтроллеров.		
	311	Способность использовать знания инструментария для создания и актуализации исходных текстов программ при программировании микроконтроллеров.		
	312	Способность использовать знания ме-		

библиотеку HAL, прием данных в блокирующем режиме; работа с UART через библиотеку HAL с использованием прерываний - Программирование синхронных интерфейсов: программирование последовательных портов SPI, I2C, USART; - Программирование прямого доступа к памяти DMA: прямой доступ к памяти в STM32, контроллер DMA; работа с контроллером DMA через CMSIS регистры; программный модуль контроллера DMA; - Программирование внешней памяти: подключение SD-карты к микроконтроллеру; файловая система FAT внешней SD-карты; настройка FatFs в STM CubeM; - Программирование режима потребления микроконтроллера: режимы пониженного потребления, SLEEP; - Программирование АЦП/ЦАП: установка конфигурации АЦП через регистры CMSIS; работа с АЦП через регистры CMSIS, основные режимы преобразования; работа с АЦП в различных режимах; режим сдвоенных преобразований АЦП STM32; работа АЦП в режиме оконного компаратора; работа с АЦП через функции библиотеки HAL;		тодов повышения читаемости программного кода при программировании микроконтроллеров.		
	313	Способность использовать знания систем кодировки символов, форматы хранения исходных текстов программ при программировании микроконтроллеров.		
	314	Способность использовать знания нормативных документов, определяющих требования к оформлению программного кода при программировании микроконтроллеров.		
	315	Способность использовать знания методов и приемов отладки программного кода при программировании микроконтроллеров.		
	316	Способность использовать знания типов и форматов сообщений об ошибках, предупреждений при программировании микроконтроллеров.		
	317	Способность использовать способов использования технологических журналов, форматов и типов записей журналов при программировании микроконтроллеров.		
	318	Способность использовать знания современных компиляторов, отладчиков и оптимизаторов программного кода при программировании микроконтроллеров.		
	319	Способность использовать знания со-		

- Программирование интерфейса USB: USB HID (human interface device) – класс устройств взаимодействия с человеком. Тема 3 Автоматизация процессов на основе систем с микроконтроллерами STM32: - Направления автоматизации процессов. Структура управляющих систем. Основы создания алгоритмов и программ для взаимодействия систем на основе МК с пользователем. Особенности работы с внешними устройствами. Взаимодействие систем на основе МК по телекоммуникационным сетям; - Операционные системы реального времени.		общений о состоянии аппаратных средств при программировании микроконтроллеров.		
	320	Способность использовать знания методов и средств верификации работоспособности выпусков программных продуктов при программировании микроконтроллеров.		
	321	Способность использовать знания языков, утилит и сред программирования, средств пакетного выполнения процедур при программировании микроконтроллеров.		
	322	Способность использовать знания возможностей используемой системы контроля версий и вспомогательных инструментальных программных средств при программировании микроконтроллеров.		
	323	Способность использовать знания установленного регламента использования системы контроля версий при программировании микроконтроллеров.		
	324	Способность использовать знания методов и средств сборки и интеграции программных модулей и компонент при программировании микроконтроллеров.		
	325	Способность использовать знания интерфейсов взаимодействия с внешней средой при программировании микроконтроллеров.		

		роконтроллеров.		
	326	Способность использовать знания интерфейсов взаимодействия внутренних модулей системы при программировании микроконтроллеров.		
	327	Способность использовать знания методов и средств сборки модулей и компонент программного обеспечения при программировании микроконтроллеров.		
	328	Способность использовать знания методов и средств разработки процедур для развертывания программного обеспечения при программировании микроконтроллеров.		
	329	Способность использовать знания методов и средств миграции и преобразования данных при программировании микроконтроллеров.		
	330	Способность использовать знания методов создания и документирования контрольных примеров и тестовых наборов данных при программировании микроконтроллеров.		
	331	Способность использовать знания правил, алгоритмов и технологий создания тестовых наборов данных при программировании микроконтроллеров.		
	332	Способность использовать знания требований к структуре и форматам хранения тестовых наборов данных		

		при программировании микроконтроллеров.		
	333	Способность использовать знания основных понятий в области качества программных продуктов при программировании микроконтроллеров.		
	334	Способность использовать знания лицензионных требований по настройке устанавливаемого программного обеспечения при программировании микроконтроллеров.		
	335	Способность использовать знания типовых причин инцидентов, возникающих при установке программного обеспечения при программировании микроконтроллеров.		
	336	Способность использовать знания основ архитектуры, устройства и функционирования вычислительных систем при установке программного обеспечения при программировании микроконтроллеров.		
	337	Способность использовать знания принципов организации, состава и схем работы операционных систем при установке программного обеспечения при программировании микроконтроллеров.		
	338	Способность использовать знания стандартов информационного взаимодействия систем при программировании микроконтроллеров.		

	У1	Способность использовать методы и приемы формализации задач при программировании микроконтроллеров.		
	У2	Способность использовать методы и приемы алгоритмизации поставленных задач при программировании микроконтроллеров.		
	У3	Способность использовать программные продукты для графического отображения алгоритмов при программировании микроконтроллеров.		
	У4	Способность применять стандартные алгоритмы в соответствующих областях при программировании микроконтроллеров.		
	У5	Способность применять выбранные языки программирования для написания программного кода при программировании микроконтроллеров.		
	У6	Способность использовать выбранную среду программирования и средства системы управления базами данных при программировании микроконтроллеров.		
	У7	Способность использовать возможности имеющейся технической и/или программной архитектуры при программировании микроконтроллеров.		
	У8	Способность применять нормативные документы, определяющие требования к оформлению программно-		

		го кода при программировании микроконтроллеров.		
	у9	Способность применять инструментарий для создания и актуализации исходных текстов программ при программировании микроконтроллеров.		
	У10	Способность выявлять ошибки в программном коде при программировании микроконтроллеров.		
	У11	Способность применять методы и приемы отладки программного кода при программировании микроконтроллеров.		
	У12	Способность интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов при программировании микроконтроллеров.		
	У13	Способность применять современные компиляторы, отладчики и оптимизаторы программного кода при программировании микроконтроллеров.		
	У14	Способность документировать произведенные действия, выявленные проблемы и способы их устранения при программировании микроконтроллеров.		
	У15	Способность проводить оценку работоспособности программного продукта при программировании микроконтроллеров.		
	У16	Способность создавать резервные		

		копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта и данных при программировании микроконтроллеров.		
	У17	Способность использовать выбранную систему контроля версий при программировании микроконтроллеров.		
	У18	Способность выполнять действия, соответствующие установленному регламенту используемой системы контроля версий при программировании микроконтроллеров.		
	У19	Способность интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов при программировании микроконтроллеров.		
	У20	Способность применять современные компиляторы, отладчики и оптимизаторы программного кода при программировании микроконтроллеров.		
	У21	Способность документировать произведенные действия, выявленные проблемы и способы их устранения при программировании микроконтроллеров.		
	У22	Способность создавать резервные копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта		

		и данных при программировании микроконтроллеров.		
	У23	Способность выполнять процедуры сборки программных модулей и компонент в программный продукт при программировании микроконтроллеров.		
	У24	Способность производить настройки параметров программного продукта и осуществлять запуск процедур сборки при программировании микроконтроллеров.		
	У25	Способность писать программный код процедур интеграции программных модулей при программировании микроконтроллеров.		
	У26	Способность использовать выбранную среду программирования для разработки процедур интеграции программных модулей при программировании микроконтроллеров.		
	У27	Способность применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов при программировании микроконтроллеров.		
	У28	Способность разрабатывать и оформлять контрольные примеры		

		для проверки работоспособности программного обеспечения при программировании микроконтроллеров.		
	У29	Способность разрабатывать процедуры генерации тестовых наборов данных с заданными характеристиками при программировании микроконтроллеров.		
	У30	Способность подготавливать наборы данных, используемых в процессе проверки работоспособности программного обеспечения при программировании микроконтроллеров.		
	У31	Способность выявлять соответствие требований заказчиков к существующим продуктам при программировании микроконтроллеров.		
	У32	Способность соблюдать процедуру установки прикладного программного обеспечения в соответствии с требованиями организации - производителя при программировании микроконтроллеров.		
	У33	Способность идентифицировать инциденты, возникающие при установке программного обеспечения, и принимать решение по изменению процедуры установки при программировании микроконтроллеров.		

3.2 Средства, применяемые для оценки уровня практической подготовки

Краткое наименование раздела (МДК), темы, параграфа дисциплины	Код результата обучения	Показатель овладения результатами обучения	Наименование оценочного средства и представление его в КОС	
			Текущий контроль	Промежуточный контроль
Тема 1 Особенности программирования микроконтроллеров STM32/ATmega328: - Принципы построения программ для микроконтроллеров. Средства разработки и отладки; - Правила составления алгоритмов. Типы алгоритмов. Диаграммы состояний. Конечный автомат; - Особенности синтаксиса для программ на МК.	31	Показатели овладения знаниями приведены в пункте 3.2	<i>Защита практических работ 1 - 4</i>	<i>Ответы на вопросы теста 1 (п. 5.3).</i> <i>Экзаменационные билеты (п. 5.4)</i>
	32			
	33			
	34			
	35			
	36			
	37			
	38			
	39			
	310			
	311			
	312			
	313			
	314			
	315			
	316		<i>Защита практических работ 5 – 28</i>	<i>Ответы на вопросы теста 2 (п. 5.3).</i> <i>Экзаменационные билеты (п. 5.4)</i>
	318			
	319			
	320			
	321			
	322			
	323			
	324			
	325			
	326			
	327			
	328			

2.3 Программирование систем микроконтроллера: - Программирование памяти: программирование памяти микроконтроллеров; программирование Flash памяти STM32; - Программирование подсистем ввода-вывода: программирование подсистем ввода/вывода микроконтроллера через регистры CMSIS; программирование портов ввода/вывода через функции библиотеки HAL; программирование портов ввода/вывода через LL-функции; - Программирование системы прерываний: система прерываний; организация и управление прерываниями; - Программирование таймеров-счетчиков: таймеры в режиме счетчиков, генерация циклических прерываний от счетчиков; конфигурация таймеров с помощью библиотек CMSIS, HAL и LL; - Программирование последовательных интерфейсов: работа с интерфейсом UART через регистры CMSIS, использование прерывания UART; работа с UART через библиотеку HAL, инициализация интерфейса и передача данных в блокирующем режиме; работа с UART через библиотеку HAL, прием	329		Экзаменационные билеты (п. 5.4)
	330		
	331		
	332		
	333		
	334		
	335		
	336		
	337		
	338		
	Y1	Показатели овладения умениями приведены в пункте 3.2	
	Y2		
	Y3		
	Y4		
	Y5		
	Y6		
	Y7		
	Y8		
	Y9		
	Y10		
	Y11		
	Y12		
	Y13		
	Y14		
	Y15		
	Y16		
	Y17		
	Y18		
	Y19		
	Y20		
	Y21		
	Y22		
	Y23		
	Y24		

данных в блокирующем режиме; работа с UART через библиотеку HAL с использованием прерываний - Программирование синхронных интерфейсов: программирование последовательных портов SPI, I2C, USART; - Программирование прямого доступа к памяти DMA: прямой доступ к памяти в STM32, контроллер DMA; работа с контроллером DMA через CMSIS регистры; программный модуль контроллера DMA; - Программирование внешней памяти: подключение SD-карты к микроконтроллеру; файловая система FAT внешней SD-карты; настройка FatFs в STM CubeM; - Программирование режима потребления микроконтроллера: режимы пониженного потребления, SLEEP; - Программирование АЦП/ЦАП: установка конфигурации АЦП через регистры CMSIS; работа с АЦП через регистры CMSIS, основные режимы преобразования; работа с АЦП в различных режимах; режим сдвоенных преобразований АЦП STM32; работа АЦП в режиме оконного компаратора; работа с АЦП через функции библиотеки HAL; - Программирование интерфейса USB: USB HID (human interface device) – класс устройств взаимодействия с че-	У25			
	У26			
	У27			
	У28			
	У29			
	У30			
	У31			
	У32			
	У33			

ловеком. Тема 3 Автоматизация процессов на основе систем с микроконтроллерами STM32: - Направления автоматизации процессов. Структура управляющих систем. Основы создания алгоритмов и программ для взаимодействия систем на основе МК с пользователем. Особенности работы с внешними устройствами. Взаимодействие систем на основе МК по телекоммуникационным сетям; - Операционные системы реального времени.			<i>Защита практических работ 29 – 52</i>	<i>Ответы на вопросы теста 4 (п. 5.3).</i> <i>Экзаменационные билеты (п. 5.4)</i>
---	--	--	---	---

4 Описание процедуры оценивания

Результаты обучения по учебной дисциплине, уровень сформированности компетенций оцениваются по четырём бальной шкале оценками: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Текущая аттестация по профессиональному модулю проводится с целью систематической проверки достижений обучающихся. Объектами оценивания являются: степень усвоения теоретических знаний, уровень овладения практическими умениями и навыками при проведении практических работ, качество выполнения самостоятельной работы, учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой профессиональному модулю).

При проведении промежуточной аттестации оценивается достижение студентом запланированных по учебной дисциплине результатов обучения, обеспечивающих результаты освоения образовательной программы в целом. Оценка на зачете выставляется с учетом оценок, полученных при прохождении текущей аттестации.

Критерии оценивания устного ответа

Оценочные средства: собеседование, устное сообщение, семинар.

5 баллов - ответ показывает прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа; умение приводить примеры современных проблем изучаемой области.

4 балла - ответ, обнаруживающий прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа. Однако допускается одна - две неточности в ответе.

3 балла – ответ, свидетельствующий в основном о знании процессов изучаемой предметной области, отличающийся недостаточной глубиной и полнотой раскрытия темы; знанием основных вопросов теории; слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры; недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа; неумение привести пример развития ситуации, провести связь с другими аспектами изучаемой области.

2 балла – ответ, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы; незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов; неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Допускаются серьезные

ошибки в содержании ответа; незнание современной проблематики изучаемой области.

Критерии оценивания письменной работы

Оценочные средства: курсовая работа, конспект, контрольная работа, письменный отчет по лабораторной работе, доклад (сообщение), в том числе выполненный в форме презентации.

5 баллов - студент выразил своё мнение по сформулированной проблеме, аргументировал его, точно определив ее содержание и составляющие. Проблема раскрыта полностью, выводы обоснованы. Приведены данные отечественной и зарубежной литературы, статистические сведения, информация нормативно-правового характера. Студент владеет навыком самостоятельной работы по заданной теме; методами и приемами анализа теоретических и/или практических аспектов изучаемой области. Фактических ошибок, связанных с пониманием проблемы, нет; графически работа оформлена правильно.

4 балла - работа характеризуется смысловой цельностью, связностью и последовательностью изложения; допущено не более 1 ошибки при объяснении смысла или содержания проблемы. Проблема раскрыта. Не все выводы сделаны и/или обоснованы. Для аргументации приводятся данные отечественных и зарубежных авторов. Продемонстрированы исследовательские умения и навыки. Фактических ошибок, связанных с пониманием проблемы, нет. Допущены одна-две ошибки в оформлении работы.

3 балла – студент проводит достаточно самостоятельный анализ основных этапов и смысловых составляющих проблемы; понимает базовые основы и теоретическое обоснование выбранной темы. Проблема раскрыта не полностью. Выводы не сделаны и/или выводы не обоснованы. Проведен анализ проблемы без привлечения дополнительной литературы. Допущено не более 2 ошибок в смысле или содержании проблемы, оформлении работы.

2 балла - работа представляет собой пересказанный или полностью переписанный исходный текст без каких бы то ни было комментариев, анализа. Не раскрыта структура и теоретическая составляющая темы. Проблема не раскрыта. Выводы отсутствуют. Допущено три или более трех ошибок в смысловом содержании раскрываемой проблемы, в оформлении работы.

Критерии оценивания тестового задания

Оценка	<i>Отлично</i>	<i>Хорошо</i>	<i>Удовлетворительно</i>	<i>Неудовлетворительно</i>
Количество правильных ответов	91 % и $\geq$	от 81% до 90,9 %	не менее 60%	менее 60%

Критерии выставления оценки студенту на зачете

Оценочные средства: устный опрос в форме ответов на контрольные вопросы, устный опрос в форме собеседования, выполнение реферата.

Оценка по промежуточной аттестации	Характеристика качества сформированности компетенций
------------------------------------	--

«зачтено» / «отлично»	Студент демонстрирует сформированность дисциплинарных компетенций на продвинутом уровне: обнаруживает всестороннее, систематическое и глубокое знание учебного материала, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
«зачтено» / «хорошо»	Студент демонстрирует сформированность дисциплинарных компетенций на базовом уровне: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
«зачтено» / «удовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на пороговом уровне: имеет знания только основного материала, но не усвоил его деталей, в ходе контрольных мероприятий допускаются значительные ошибки, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ, при оперировании знаниями и умениями при их переносе на новые ситуации.
«не зачтено» / «неудовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на уровне ниже порогового: выявляется полное или практически полное отсутствие знаний значительной части программного материала, студент допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы, умения и навыки не сформированы.

5. Примеры оценочных средств для проведения текущей аттестации

5.1 Вопросы для собеседования (устного опроса):

1 Особенности программирования микроконтроллеров STM32/ATmega328

Тема 1.1 Принципы построения программ для микроконтроллеров. Средства разработки и отладки.

1. Назовите принципы построения программ для микроконтроллеров.
2. Перечислите среды разработки и отладки ARM STM32.
3. Охарактеризуйте первую абстракцию CMSIS.
4. Охарактеризуйте вторую абстракцию SPL.
5. Охарактеризуйте третью абстракцию HAL.

Тема 1.2 Правила составления алгоритмов. Типы алгоритмов. Диаграммы состояний. Конечный автомат

6. Опишите свойства алгоритма.
7. Опишите формы представления алгоритма.
8. Какие структуры представления алгоритма Вы можете назвать?
9. Опишите алгоритмическую конструкцию «цикл с предусловием».

10. Опишите алгоритмическую конструкцию «цикл с постусловием».
11. Опишите алгоритмическую конструкцию «цикл с параметром».
12. Опишите алгоритмическую конструкцию «цикл, вложенный в неполную развилку».
13. Опишите алгоритмическую конструкцию «цикл в цикле».
14. Опишите алгоритмическую конструкцию «развилка в развилке».
15. Опишите алгоритмическую конструкцию «трехкратное вложение структур друг в друга».
16. Что такое диаграмма конечных состояний?
17. Что такое конечный автомат?
18. Опишите свойства переходов конечного автомата.
19. Опишите события триггера конечного автомата.
20. Что такое хронологические состояния конечного автомата?
21. Опишите абстрактный конечный автомат.
22. Что такое цепочки состояний конечного автомата?

Тема 1.3 Особенности синтаксиса для программ на МК

23. Перечислите особенности типов данных языка Си для STM32.

2 Модульное программирование микроконтроллеров STM32

Тема 2.1 Низкоуровневые библиотеки для платформы ARM Cortex (STM32)

24. Перечислите библиотеки для платформы ARM Cortex (STM32).
25. Опишите слои абстракций программирования ARM Cortex (STM32).
26. Охарактеризуйте работу с регистрами STM32 через библиотеку CMSIS.
27. Охарактеризуйте работу с библиотекой HAL при программировании STM32.
28. Сравните CMSIS и HAL при программировании STM32.

Тема 2.2 Средства программирования и отладки

2.2.1 Конфигуратор STM32CubeMX

29. Что такое STM32CubeMX?
30. Опишите процесс генерации кода в STM32CubeMX.
31. Охарактеризуйте интерфейс код-генератора STM32CubeMX.
32. Расскажите про конфигурацию системы тактирования в STM32CubeMX.

2.2.2 Средства разработки System Workbench for STM32

33. Что такое System Workbench for STM32?
34. Опишите архитектуру System Workbench for STM32.
35. Опишите процесс работы в Архитектура System Workbench for STM32.

2.2.3 Proteus Design Suite

35. Что такое Proteus Design Suite?

Тема 2.3 Программирование систем микроконтроллера

2.3.1 Программирование памяти микроконтроллера

36. Опишите фиксированное адресное пространство памяти Cortex-M.

- 37. Опишите карту памяти области кода на микроконтроллерах STM32.
- 38. Как выбирается COM-порт в Arduino IDE?
- 39. Что такое технология битовых лент (bit-banding)?
- 40. Опишите карту памяти и области доступа к битам (битовых лент)
- 41. Как выполняется защита флеш-памяти от копирования в STM32?

2.3.2 Программирование Flash памяти STM32

- 42. Обоснуйте необходимость Flash памяти в STM32.
- 43. Перечислите достоинства и недостатки применения Flash памяти STM32.
- 44. Опишите структуру Flash памяти STM32.
- 45. Перечислите и охарактеризуйте операции с Flash памяти STM32 (инициализация, чтение и запись данных).
- 46. Опишите использование структур данных в Flash памяти STM32.
- 47. Как выполняется блокировка чтения/записи в Flash памяти STM32?
- 48. Опишите использование утилиты STM32 ST-LINK Utility.

2.3.3 Программирование подсистем ввода/вывода микроконтроллеров через регистры CMSIS

- 49. Объясните преимущества и недостатки программирования подсистем ввода/вывода микроконтроллеров через регистры CMSIS.

2.3.4 Программирование портов ввода-вывода через функции библиотеки HAL

- 50. Опишите процесс конфигурации портов с помощью библиотеки HAL.

2.3.5 Программирование портов ввода-вывода через LL-функции

- 51. Сравните драйверы HAL и LL.
- 52. Опишите совместное использование библиотек HAL и LL.
- 53. Опишите процесс управления портами ввода-вывода с помощью HAL-функций.

2.3.6 Система прерываний. Организация и управление прерываниями

- 54. Чем аппаратные прерывания отличаются от событий?
- 55. Опишите контроллер прерываний STM32.
- 56. Опишите процесс программного управления прерываниями STM32.
- 57. Перечислите функции обработки прерываний STM32.

2.3.7 Таймеры в режиме счетчиков. Генерация циклических прерываний от таймеров

- 58. Назовите таймеры STM32.
- 59. Объясните фразу «Таймеры STM32 - многофункциональные устройства».
- 60. Объясните принцип работы функциональной схемы таймера STM32.
- 61. Опишите режимы работы таймера STM32.
- 62. Опишите процесс установки конфигурации таймера с помощью STM32CubeMX.

- 63. Опишите работу Prescaler (PSC) таймера STM32.
- 64. Опишите режим счетчика таймера STM32.
- 65. Опишите регистр счетчика повторов таймера STM32.

2.3.8 Конфигурация таймеров с помощью библиотек CMSIS, HAL и LL

- 66. Объясните логику работы прерываний таймеров STM32.
- 67. Перечислите и охарактеризуйте регистры STM32.
- 68. Расскажите про управление таймером с помощью библиотеки HAL.
- 69. Расскажите про настройку таймера и прерываний от него через функции библиотеки LL.

2.3.9 Работа с интерфейсом UART через регистры CMSIS. Использование прерывания UART

- 70. Опишите принцип работы протокола интерфейса UART.
- 71. Расскажите про UART в микроконтроллерах STM32.
- 72. Перечислите и охарактеризуйте регистры UART STM32.
- 73. Охарактеризуйте использование прерываний UART STM32.

2.3.10 Работа с UART через библиотеку HAL. Инициализация интерфейса и передача данных в блокирующем режиме. Отладка программ с помощью UART. Функция printf

- 74. Опишите процесс инициализации, установки конфигурации UART в STM32.
- 75. Объясните работу с UART через HAL-библиотеку в блокирующем режиме.
- 76. Как выполняется отладка программ в STM32 с помощью последовательного порта?
- 77. Что такое функция printf STM32?
- 78. Как выводятся числа с плавающей запятой с помощью printf в средах программирования микроконтроллеров?

2.3.11 Программирование последовательных портов SPI, I²C, USART

- 79. Опишите программную модель SPI в STM32.
- 80. Опишите программную модель I²C в STM32.
- 81. Опишите программную модель USART в STM32.

2.3.12 Прямой доступ к памяти в STM32. Контроллер DMA

- 82. Объясните принцип работы прямого доступа к памяти (DMA).
- 83. Опишите общие принципы работы контроллера прямого доступа к памяти в STM32.
- 84. Перечислите и охарактеризуйте каналы DMA в STM32.
- 85. Перечислите и охарактеризуйте регистры каналов DMA в STM32.
- 86. Опишите процесс конфигурации канала DMA в STM32.
- 87. Опишите специфику доступа к периферийным устройствам через шины AHB/APB в STM32.
- 88. Какие прерывания DMA в STM32 Вы можете назвать?

2.3.13 Работа с контроллером DMA через CMSIS регистры

- 89. Объясните DMA-передачу данных из памяти в порт ввода-вывода STM32.
- 90. Опишите функции (программные) управления генератором импульсов в STM32.
- 91. Как реализуется управление генератором от компьютера в STM32?
- 92. Что такое программа верхнего уровня для управления генератором импульсов в STM32?

2.3.14 Программный модуль контроллера ПДП (DMA1)

- 93. Опишите регистры программного модуля контроллера ПДП (DMA1) STM32.

2.3.15 Подключение SD-карты к микроконтроллеру

- 94. Опишите схему подключения SD-карты к микроконтроллеру.

2.3.16 Файловая система FAT на внешней SD-карте

- 95. Опишите файловую систему FAT на внешней SD-карте в STM32.

2.3.17 Настройка FatFs в STM32CubeM

- 96. Опишите схему подключения карты памяти к STM32.

2.3.18 Режимы пониженного энергопотребления. SLEEP

- 97. Какие режимы пониженного энергопотребления в STM32 Вы знаете?
- 98. Опишите режим SLEEP в STM32.

2.3.19 АЦП STM32. Общие сведения, режимы. Установка конфигурации через регистры CMSIS

- 99. Объясните принцип работы АЦП.
- 100. Охарактеризуйте АЦП в STM32.
- 101. Объясните работу регулярных и инжектированных каналов АЦП в STM32.
- 102. Как выполнить конфигурацию АЦП через регистры CMSIS?

2.3.20 Работа с АЦП через регистры CMSIS. Основные режимы преобразования

- 103. Какие мероприятия нужно выполнить, чтобы совершить преобразование аналогового сигнала в STM32?
- 104. Назовите режимы преобразования аналогового сигнала в STM32.
- 105. Объясните режим один канал, однократное преобразование аналогового сигнала в STM32.
- 106. Объясните режим один канал, непрерывное преобразование аналогового сигнала в STM32.
- 107. Объясните режим несколько каналов, однократное преобразование аналогового сигнала в STM32.
- 108. Объясните режим несколько каналов, непрерывное преобразование аналогового сигнала в STM32.

2.3.21 Работа с АЦП в различных режимах. Запуск от таймера, чтение результата с использованием прерываний.

- 109. Опишите внешний запуск опроса инжектированной группы в STM32.
- 110. Опишите автоматический запуск опроса инжектированных каналов в STM32.

2.3.22 Режим сдвоенных преобразований АЦП STM32

- 111. Что такое режим сдвоенных преобразований в АЦП STM32?
- 112. Что такое режим одновременных преобразований инжектированных каналов в АЦП STM32?
- 113. Что такое режим одновременных преобразований регулярных каналов в АЦП STM32?
- 114. Что такое режим быстрых преобразований со смещением во времени в АЦП STM32?
- 115. Что такое режим медленных преобразований со смещением во времени в АЦП STM32?
- 116. Что такое режим поочередного запуска АЦП?
- 117. Что такое комбинированный режим одновременного преобразования инжектированных и регулярных каналов в АЦП STM32?
- 118. Что такое комбинированный режим преобразования одновременного преобразования регулярных каналов и поочередного запуска в АЦП STM32?
- 119. Что такое комбинированный режим одновременного преобразования инжектированных каналов и преобразований со смещением во времени в АЦП STM32?

2.3.23 Работа АЦП в режиме оконного компаратора

- 120. Что такое режим оконного компаратора в АЦП STM32?
- 121. Как проверить режим оконного компаратора в АЦП STM32?
- 122. Что такое внутренний источник опорного напряжения в АЦП STM32?
- 123. Перечислите метрологические характеристики АЦП STM32.

2.3.24 Работа с АЦП через функции библиотеки HAL

- 124. Опишите работу АЦП STM32 через функции библиотеки HAL.

2.3.25 USB HID (human interface device) – класс устройств для взаимодействия с человеком

- 125. Как Вы можете объяснить выражение «USB HID (human interface device) – класс устройств для взаимодействия с человеком»? Приведите пример.
- 126. Что такое ChoiceGroup?
- 127. Что такое CustomItem?

Тема 3 Автоматизация процессов на основе систем с МК

3.1 Структура управляющих систем. Понятие обратной связи и принципы управления. Принципы автоматического управления

- 128. Объясните две основные схемы управляющих систем.
- 129. Опишите свойства обратной связи управляющих систем.
- 130. Объясните принципы автоматического управления.

3.2 Основы создания алгоритмов и программ для взаимодействия систем на основе МК с пользователем

131. Опишите этап проектирования интерфейса (UI/UX) взаимодействия пользователя с системой.
132. Опишите основы разработки алгоритмов.
133. Как выполняется программная реализация алгоритмов в микроконтроллерах?
134. Что такое пользовательский интерфейс?
135. Опишите формы представления пользовательского интерфейса.
136. Перечислите и охарактеризуйте виды пользовательских интерфейсов.
137. Назовите компоненты пользовательского интерфейса.
138. Опишите этапы разработки пользовательского интерфейса.
139. Объясните правила и принципы разработки хорошего интерфейса.

3.3 Особенности работы с внешними устройствами

140. Объясните низкоуровневое взаимодействие микроконтроллеров с внешним миром.
141. Объясните высокоуровневое взаимодействие микроконтроллеров с внешним миром.
142. Перечислите достоинства и недостатки низкоуровневого и высокоуровневого взаимодействий микроконтроллеров с внешним миром.
143. Опишите подключение контроллера HD44780 к STMF103C06.
144. Опишите подключение разъема USB2 микроконтроллеру.
145. Как обрабатываются сигналы энкодера в микроконтроллере?

3.4 Взаимодействие систем на основе МК по телекоммуникационным сетям с другими вычислительными системами

146. Опишите способы взаимодействия МК с другими вычислительными системами.
147. Перечислите основные компоненты МК с другими вычислительными системами.
148. Как организуется подключение микроконтроллера к локальной сети? Приведите пример.
149. Назовите и охарактеризуйте уровни модели OSI.
150. Опишите структуру кадра Ethernet.
151. Что такое протокол ARP и как он работает.
152. Объясните понятия ARP-запрос и ARP-ответ.
153. Опишите работу с протоколом ICMP.
154. Опишите работу с протоколом IP.
155. Опишите формат заголовка IP.
156. Опишите формат заголовка ICMP.

3.5 Операционные системы реального времени

157. Какие три типа операционных систем с точки зрения критерия использования существуют?
158. Назовите примеры операционных систем реального времени.
159. Опишите граф состояния задач в ОС реального времени.

5.2 Примерная тематика курсовых работ

- 1 Система контроля температуры на основе МК
- 2 Система ограничения скорости автомобиля на основе МК
- 3 Система трекинга автомобиля на основе МК
- 4 Система учета электроэнергии на основе МК
- 5 Система пожаробезопасности и обнаружения газов в помещении на основе МК
- 6 Разработка программы управления на микроконтроллере для системы контроля допуска в здание
- 7 Разработка программы управления на микроконтроллере для управляющей системы охлаждения ПК
- 8 Разработка программы управления на микроконтроллере для калькулятора
- 9 Разработка программы управления на микроконтроллере для часов
- 10 Разработка программы управления на микроконтроллере для цифровой клавиатуры для ПК
- 11 Разработка программы управления на микроконтроллере для системы проверки кабеля типа витая пара
- 12 Разработка программы управления на микроконтроллере для системы вывода изображений на светодиодную матрицу
- 13 Разработка программы управления на микроконтроллере для системы включения и выключения света по звуковому сигналу
- 14 Разработка программы управления на микроконтроллере для системы включения и выключения света в помещении, по введенному графику.
- 14 Разработка программы управления на микроконтроллере для системы поддержания равновесия в полете для квадрокоптера
- 15 Разработка программы управления на микроконтроллере для системы управления коммуникациями частного домовладения
- 16 Разработка программы управления на микроконтроллере для системы пульта управления
- 17 Разработка программы управления на микроконтроллере для подвижного робота, с автопарковкой
- 18 Разработка программы управления на микроконтроллере для системы зарядки и индикации аккумуляторных батарей
- 19 Разработка программы управления на микроконтроллере для измерения скорости ветра на улице и ее индикации
- 20 Разработка программы управления на микроконтроллере для цифрового амперметра
- 21 Разработка программы управления на микроконтроллере для тахометра
- 22 Разработка программы управления на микроконтроллере для телефонной сети из трех абонентов
- 23 Разработка программы управления на микроконтроллере для автомобильной сигнализации
- 24 Разработка программы управления на микроконтроллере для проигрывателя рингтонов
- 25 Разработка программы управления на микроконтроллере для дистанционного инфракрасного управления

- 26 Разработка программы управления на микроконтроллере для сигнализации в холодильной установке
- 27 Разработка программы управления на микроконтроллере для сетевой метеостанции
- 28 Разработка программы управления на микроконтроллере для создание игровой приставки «тетрис»
- 29 Разработка программы управления на микроконтроллере для создания светодиодной RGB матрицы, с выводом на нее изображения
- 30 Разработка программы управления на микроконтроллере для системы контроля доступа на основе RFID
- 31 Разработка программы управления на микроконтроллере для системы управления роботом через Bluetooth
- 32 Разработка программы управления на микроконтроллере для считывания и записи показаний датчиков для создания массива данных.
- 33 Разработка программы управления на микроконтроллере для считывания команд радиопульта управления
- 34 Разработка программы управления на микроконтроллере для управления мироробота паука

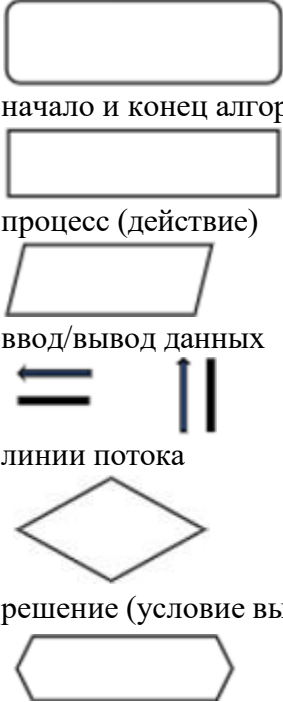
5.3 Примеры тестовых заданий

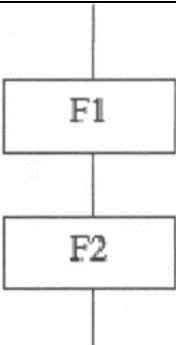
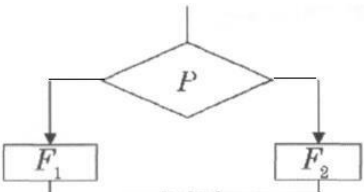
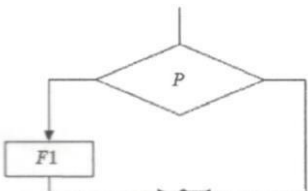
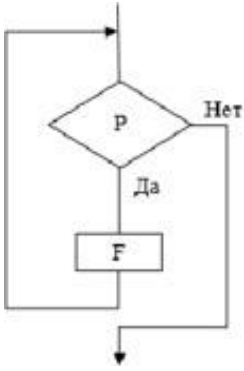
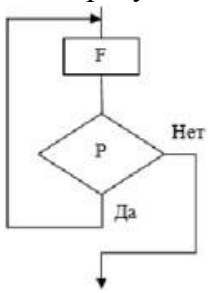
Тест 1 Особенности программирования микроконтроллеров STM32 или аналогов

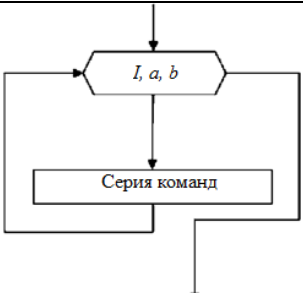
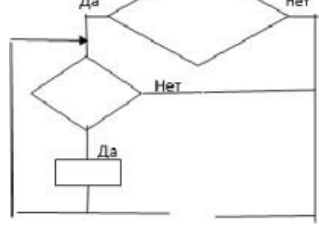
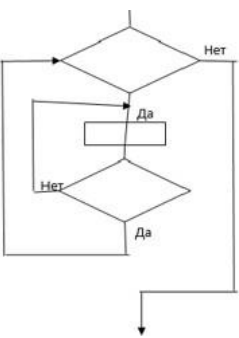
Количество вопросов – 50. Преподаватель формирует нужное количество вариантов теста с любым количеством вопросов.

№ п/п	Вопрос с вариантами ответов	Правильный ответ
1.	_____ — это документ, который детально описывает требования к проекту, товару, работе или услуге, включая цели, задачи, методы выполнения, характеристики и сроки (два слова в именительном падеже).	Техническое задание
2.	_____ — это программа, которая переводит исходный код, написанный на языке программирования, в машинный код, понятный компьютеру (одно слово в именительном падеже).	компилятор, транслятор
3.	_____ — это не само программирование, а процесс преобразования кода, написанного программистом, в машинный код, который может исполнить компьютер (одно слово в именительном падеже)	компиляция, трансляция
4.	_____ — это общий термин для двоичных (binary) файлов, содержащих данные в машинно-читаемом формате, в отличие от текстовых файлов. (название файла в именительном падеже).	BIN-файл, binary-файл, двоичный файл
5.	_____ — это текстовый файл специального формата, который используется для представления двоичной информации в виде читаемых символов шестнадцатеричной системы счисления. (название файла в именительном падеже).	HEX-файл, шестнадцатеричный файл
6.	_____ — это текстовый файл с инструкциями для утилиты make, которая автоматизирует процесс сборки программного обеспечения или других задач в проекте (название файла в именительном падеже).	make файл make-файл Makefile
7.	Как называется комплекс программного обеспечения, который объединяет инструменты для написания, тестирования и отладки кода в одном интерфейсе (несколько вариантов ответов). 1. интегрированная среда разработки 2. язык программирования высокого уровня 3. транслятор 4. IDE 5. отладчик	1, 4
8.	Какие IDE, бесплатно распространяемые для микроконтроллеров ARM STM32, в настоящее время не рекомендованы для разработки новых продуктов, а могут использоваться только уже для реализуемых проектов (несколько вариантов ответов). 1. System Workbench for STM32 2. STM32CubeIDE 3. winIDEA Open 4. Coocox CoIDE	4, 5

	5. Atollic TrueSTUDIO	
9.	Сколько слоев абстрагирования интерфейсов программного обеспечения для микроконтроллеров с ядром Cortex-M существует? 1 2 3 4 5	3
10.	Набор стандартов и интерфейсов программного обеспечения для микроконтроллеров на базе процессоров ARM Cortex-M, предоставляющий разработчикам унифицированный и независимый от производителя уровень абстракции для аппаратных компонентов, таких как процессор, периферия и инструменты отладки. - CMSIS - SPL - HAL - LL	1
11.	Низкоуровневая библиотека драйверов, которая предоставляет прямой доступ к регистрам периферии микроконтроллера, что позволяет разработчикам выполнять операции с периферийными устройствами более быстро и с меньшим потреблением памяти по сравнению с высокоуровневыми драйверами. - CMSIS - SPL - HAL - LL	4
12.	Уровень аппаратной абстракции, представляющий собой программную библиотеку, которая служит интерфейсом между вашим кодом и аппаратным обеспечением микроконтроллера, скрывает низкоуровневые детали конкретных аппаратных компонентов, таких как GPIO или таймеры, и предоставляет набор стандартизированных функций (API) для их использования. - CMSIS - SPL - HAL - LL	3
13.	В какой слой входит кроссплатформенная утилита с графическим интерфейсом STM32CubeMX? - CMSIS - SPL - HAL - LL	3
14.	_____ — свойство алгоритма, заключающееся в том, что за конечное число шагов должен быть получен результат, либо доказано его отсутствие (одно слово в именительном падеже)	определенность
15.	_____ — свойство алгоритма, заключающееся в обязательном получении некоторого результата или сиг-	результативность

	нала о том, что данный алгоритм неприменим для решения поставленной задачи (одно слово в именительном падеже)	
16.	_____ — свойство алгоритма, заключающееся в возможности получения результата при различных исходных данных для некоторого класса сходных задач (одно слово в именительном падеже)	массовость
17.	_____ - свойство алгоритма, заключающееся в отвлечении от содержания поставленной задачи и строгом выполнении некоторого правила, инструкции (одно слово в именительном падеже)	формальность
18	_____ — свойство алгоритма, заключающееся в возможности разбиения алгоритма на отдельные элементарные действия (одно слово в именительном падеже)	дискретность
19.	В какой слой входит кроссплатформенная утилита с графическим интерфейсом STM32CubeMX? 1. CMSIS 2. SPL 3. HAL 4. LL	3
20.	Алгоритм, записанный на языке программирования это _____ (одно слово в именительном падеже)	программа
21.	 начало и конец алгоритма процесс (действие) ввод/вывод данных линии потока решение (условие выбора) заголовок цикла ошибка алгоритма	
22.	Установить соответствие между структурой алгоритма и ее графическим изображением	

	 линейная структура  ветвление ЕСЛИ – ТО – ИНАЧЕ  ветвление ЕСЛИ – ТО повторить цикл	
23.	Установить соответствие между структурой алгоритма и ее графическим изображением  цикл с предусловием  цикл с постусловием	

	 цикл с параметром  цикл, вложенный в неполную развилку  развилка в развилке трехкратное вложение структур друг в друга	
24	_____ - это графическая нотация (система условных знаков, символов и правил) для изображения спецификаций конечных автоматов, применяемых для моделирования динамического поведения элементов модели (три слова)	диаграмма конечных состояний
25	_____ - средство моделирования динамического поведения элемента модели, набор состояний, связанный переходами (два слова)	конечный автомат
26	_____ - это отношение между двумя состояниями, указывающее, что объект в первом состоянии выполнит определенные действия и перейдет в другое состояние, когда произойдет какое-либо событие, и будут выполнены указанные условия (одно слово в именительном падеже)	переход
27	В нотации конечного автомата показать соответствие его состояния номеру на рисунке	

	Состояние машины 1 начальное состояние 2 конечное состояние 3 состояние 4 переход 5 действие 6 событие 7 внутренний переход внешний переход	
28	Если рассматривать редактор, как конечный автомат, какие события являются переходами (несколько вариантов ответов)? 1. пусто 2. ожидание команды 3. загрузить файл 4. вставить текст 5. ожидание текста 6. Вставить символ 7. сохранить и завершить работу	3, 4, 6, 7
29	Установить соответствие между названием элемента конечного автомата и его свойством исходное состояние – состояние, из которого выполняется переход. событие триггера – событие, инициирующее осуществление перехода, если объект находится в исходном состоянии и выполнено сторожевое условие сторожевое условие – булевское выражение, которое вычисляется в момент срабатывания триггера для перехода действие – атомарная вычислительная процедура, которая может быть прямо выполнена с объектом целевое состояние – состояние, которое будет активным после перехода конечное состояние	
30	Имеет ли внутреннюю структуру простое состояние? 1. да 2. нет	2
31	Верно ли, что вложенный конечный автомат может иметь не более одного начального состояния? 1. Да 2. нет	1
32	Верно ли, что вложенный конечный автомат может иметь	1

	только одно конечное состояние? 1. да 2. нет	
33	Верно ли, что хронологические состояния позволяют повторно войти в подсостояние, которое было активным при выходе из составного состояния 1. да 2. нет	1
34	Моделирование существования объекта подразумевает следующие аспекты (несколько вариантов ответа) 1. события, на которые может отвечать объект 2. ответ на события 3. влияние прошлого на текущее поведение 4. промежуточные события 5. конечные события	1, 2, 3
35	Проект для программирования на языке СИ состоит из файлов с расширениями (несколько вариантов ответа) 1. *.c 2. *.k 3. *.m 4. *.h 5. *.e	1 4
36	35. Основной файл, в котором находится код программы на языке СИ, называется_____ (слово на английском языке, точка, расширение)	main.c
37	Необходимо ли добавить в проект библиотеку с функциями, написанными сторонними разработчиками, при необходимости их использования в проекте? 1. да 2. нет	1
38	Переменная этого типа может быть только целым числом от -2147483648 до 2147483647 1. int 2. float 3. double 4. ulong 5. long 6. char 7. bool	1
39	Переменная этого типа число с точностью до 7 разрядов от $\pm 1,5 \cdot 10^{45}$ до $\pm 3,4 \cdot 10^{33}$ 1. int 2. float 3. double 4. ulong 5. long 6. char 7. bool	2
40	Один символ 1. int 2. float	6

	3. double 4. ulong 5. long 6. char 7. bool	
41	Логическая переменная, которая может принимать одно из двух возможных значений: истина (1) или ложь (0) 1. int 2. float 3. double 4. ulong 5. long 6. char 7. bool	7
42	Какая цифра должна стоять в квадратных скобках в записи <code>char str.Message[] = "Слово"</code>? 1. 1 2. 2 3. 3 4. 4 5. 5 6. 6	5
43	Если $a = 5$ и $b = 3$, чему будет равен результат вычисления выражения $a \% b$? 1. 1 2. 2 3. 3 4. 10 5. 8	2
44	Верно ли, что выражение $a < b$ становится ложным если значение переменной «a» меньше значения переменной «b» и истинным, если значения равны или «a» больше «b»? 1. да 2. нет	2
45	Как называется выражение? <pre>while(условие) { <i>тело цикла</i> }</pre> 1. цикл с предпроверкой условий 2. цикл со счетчиком 3. условный переход 4. безусловный переход	1
46	Как называется выражение? <pre>for (начальное_значение; цикл_выполняется_до; шаг) { <i>тело цикла</i> , }</pre> 1. цикл с предпроверкой условий	2

	2. цикл со счетчиком 3. условный переход 4. безусловный переход	
47	Как называется выражение? <pre> if (условие) { тело 1 } else { тело 2 } </pre> 1. цикл с предпроверкой условий 2. цикл со счетчиком 3. условный переход 4. безусловный переход	3
48	Как называется выражение? <pre> if (условие 1) { тело 1 } elseif (условие 2) { тело 2 } </pre> 1. цикл с предпроверкой условий 2. цикл со счетчиком 3. условный переход 4. безусловный переход	3
49	Верно ли, что комментарии влияют на размер программы? 1. да 2. нет	2
50	Разрядность переменной типа int для микроконтроллера Ардуино 1. 4 2. 8 3. 16 4. 32	3
51	Разрядность переменной типа int для микроконтроллера STM32 1. 4 2. 8 3. 16 4. 32	4

Тест 2 Модульное программирование микроконтроллеров STM32

Количество вопросов – 76. Преподаватель формирует нужное количество вариантов теста с любым количеством вопросов.

№ п/п	Вопрос с вариантами ответов	Правильный ответ
1.	Показать соответствие названия библиотеки для платформы ARM Cortex (STM32) ее содержанию содержит низкоуровневые, не зависящие от производителя методы для доступа к аппаратуре микроконтроллеров ARM Cortex – CMSIS библиотека поверх CMSIS или в дополнение к CMSIS – STM32CUBE высокоуровневый, независимый от производителя интерфейс программирования (C++ API) для аппаратуры микроконтроллеров различных производителей – MBED выполнение конкретных задач для 8-разрядных микроконтроллеров	
2.	Расставить уровни в программе на основе библиотеки MBED, спускаясь от высокого к самому низкому 1- слой библиотек CMSIS, нацеленный на определенную модель микроконтроллера 2- слой HAL библиотек, предоставленный производителем (такой как STM32CubeMX) - не обязательный слой 3 – микроконтроллер 4 - ваша программа на C++ 5 - слой HAL библиотеки MBED, нацеленный на определенную модель микроконтроллера 6 - библиотека MBED C++ 1. 4, 6, 5, 2, 1, 3 2. 6, 5, 2, 1, 4, 3 3. 1, 2, 5, 4, 3, 6, 4. 5, 2, 1, 3, 4, 6, 5. 2, 1, 4, 3, 6, 5	1
3.	Показать соответствие слоев программного обеспечения микроконтроллера их содержанию слой CMSIS - легкая абстракция, находящаяся между остальным кодом и железом микроконтроллера, гарантирует, что доступ к регистрам CPU, прерываниям и т. п. будет (почти) одинаковым для всех поддерживаемых моделей микроконтроллеров слой HAL – предоставляется производителем, может использоваться для упрощения доступа к остальной периферийной части микроконтроллера библиотека MBED C++ - содержит обычные объекты C++ для различных периферийных устройств (например, DigitalOut, SPI, Serial) программа - находится на самой вершине этого стека, используя	

	ет объекты C++ для управления микроконтроллером на верхнем уровне, используя все эти промежуточные слои абстракции слой SPL	
4.	Верно ли что программирование любого микроконтроллера – это работа с регистрами? 1. да 2. нет	1
5.	Какой бит регистра управления тактированием (RCC_CR) надо установить в единицу, чтобы включить режим тактирования от кварцевого резонатора? 1. бит 16 HSEON 2. бит 18 HSEBYP 3. оба бита	1
6.	Какой бит регистра управления тактированием (RCC_CR) надо установить в единицу для использования в качестве источника тактирования сигнала внешнего генератора? 1. бит 16 HSEON 2. бит 18 HSEBYP 3. оба бита	3
7.	Для включения кварцевого резонатора надо выполнить команду 1. RCC->CR = 0x00010000 2. RCC->CR = 0x00110000 3. RCC->CR = 0x00011000 4. RCC->CR = 0x00001000 5. RCC->CR = 0x00010100	1
8.	Какие качества присущи библиотеке CMSIS? (несколько вариантов ответов) 1. компактность программы 2. высокое быстродействие 3. простота программирования 4. сложность программирования 5. громоздкость программы 6. низкое быстродействие	1, 2, 4
9.	Верно ли, что библиотека HAL является промежуточным уровнем между приложением и уровнем CMSIS регистров? 1. да 2. нет	1
10.	Драйверы HAL являются 1. функционально-ориентированными 2. ориентированными на конкретные периферийные узлы	1
11.	Какие качества присущи библиотеке HAL? (несколько вариантов ответов) 1. компактность программы 2. высокое быстродействие 3. простота программирования 4. сложность программирования 5. громоздкость программы 6. низкое быстродействие	3, 5, 6
12.	Показать соответствие шагов конфигурации микроконтроллера	

	с использованием STM32CubeMX шаг 1 – выбирается микроконтроллер или микропроцессор, который соответствует требуемому набору периферийных устройств шаг 2 – конфигурируются GPIO и настраиваются таймеры системы шаг 3 – запускается генерация кода, соответствующая выбранному варианту конфигурации обеспечивается разрешение конфликтов и формируется код инициализации МК и периферии на языке СИ.	
13.	Показать соответствие цветовой подсветки CubeMX при выявлении и устранении конфликтов, присутствующих в проекте зеленый цвет – отсутствие проблем желтый цвет – ограниченная функциональность красный цвет – недоступность настроек синий цвет	
14.	Частота конфигурация тактирования в STM32CubeMX по умолчанию _____ МГц (вставить число).	8
15.	В отладочном модуле установлен кварцевый резонатор на _____ МГц (вставить число).	12
16.	Для достижения максимальной тактовой частоты в 72 МГц для ядра МК и периферийной шины APB2 множитель PLL должен быть равным _____ (вставить число)	6
17.	Максимальная частота периферийной шины APB1 равна _____ МГц (вставить число)	36
18.	Поддерживает ли System Workbench for STM32 на основе среды разработки программного обеспечения IDE Eclipse все семейство микроконтроллеров STM32 и связанных с ними плат? 1. да 2. нет	1
19.	Являются ли STM32CubeMX и System Workbench for STM32 отдельными инструментами, что накладывает определенные неудобства при изменении или частичной модернизации проекта? 1. да 2. нет	1
20.	Является ли STM32CubeIDE бесплатно распространяемой средой разработки, загрузить которую можно с сайта производителя? 1. да 2. нет	1
21.	В STM32CubeIDE основной исполняемый код вносится в файл _____ (вставить название файла)	main.c
22.	В STM32CubeIDE по умолчанию при компиляции создается _____ (вставить название файла)	elf-файл
23.	В STM32CubeIDE после выполнения указанных настроек при компиляции дополнительно к выбранным будет формироваться _____ (вставить название файла)	hex-файл
24.	Доступна ли версия STM32CubeIDE для установки в операционной системе Linux?	1

	1. да 2. нет	
25	Стандартизированное адресное пространство памяти, общее для всех ядер Cortex-M, имеет размер ____ГБ (вставить число)	4
26	Для области кода адресного пространства памяти Cortex-M выделены первые _____МБ (вставить число)	512
27	Все процессоры Cortex-M отображают область кода, начиная с адреса 1. 0x0000 00000 2. 0x0000 00001 3. 0x0000 00002 4. 0x0000 00003 5. 0x0000 00004 6. 0x0000 00005	6
28	Для всех устройств STM32 область, связанная с внутренней Flash-памятью микроконтроллера, в которой находится программный код, начинается с адреса 1. 0x0500 0000 2. 0x0600 0000 3. 0x0700 0000 4. 0x0800 0000 5. 0x0900 0000	4
29	Область, ограниченная внутренним статическим ОЗУ (SRAM) микроконтроллера, начинается с адреса 1. 0x1000 0000 2. 0x2000 0000 3. 0x3000 0000 4. 0x4000 0000 5. 0x5000 0000	2
30	Разрядность команд процессоров ARM 1. 8 2. 16 3. 24 4. 32 5. 64	4
31	Разрядность системы команд Thumb 1. 8 2. 16 3. 24 4. 32 5. 64	2
32	Разрядность системы команд Thumb-2 (несколько вариантов ответа) 1. 8 2. 16 3. 24 4. 32 5. 64	2, 4
33	Имеет ли серия микроконтроллеров STM32f1xx встроенную EEPROM? 1. да	2

	2. нет	
34	Имеет ли серия микроконтроллеров LP (Low Power) встроенную EEPROM? 1. Да 2. Нет	1
35	Инструкции программного кода микроконтроллера хранятся в 1. HDD 2. ПЗУ 3. Flash 4. SRAM	3
36	Может ли использоваться часть Flash памяти для хранения настроек микроконтроллера? 1. да 2. нет	1
37	Чтобы сохранить ресурс Flash памяти, рекомендуется (несколько вариантов ответа): 1. не писать постоянно во Flash 2. производить операции очистки / записи лишь для сохранения данных в "энергонезависимой" памяти 3. в основном работать с оперативной памятью	1, 2, 3
38	Верно ли, что минимум можно использовать 1 страницу Flash (даже для одного байта)? 1. да 2. нет	1
39	Flash память состоит из следующих блоков (несколько вариантов ответа): 1. основной блок памяти 2. резервный блок памяти 3. информационный блок 4. операционный блок 5. блок регистров управления	1, 3, 5
40	Операции чтения flash-памяти и доступ к данным выполняются через шину 1. АHB 2. ICode 3. DCode	1
41	Интерфейс flash-памяти (FLITF (Flash Memory Interface)) позволяет выполнять следующие операции (несколько вариантов ответа): 1. Чтение из памяти с возможностью буферизации 2. Операции очистки памяти 3. Операции записи и очистки памяти 4. Организация защиты flash от чтения и/или записи	1, 3, 4
42	Буферизация чтения flash-памяти, для выполнения команд, выполняется через шину 1. АHB 2. ICode 3. DCode	2
43	Приоритет доступу к данным flash-памяти отдается шине 1. АHB 2. ICode	3

	3. DCode	
44	Требуется ли инициализация МК, чтобы сконфигурировать задержку (Latency) flash-памяти? 1. да 2. нет	2
45	Какой режим чтения данных из flash-памяти конфигурируется во время инициализации МК? 1. Задержка (Latency) 2. Буферизация (2 слова по 64 бита) (Prefetch buffer) 3. Половинный цикл (Half cycle)	2
46	Показать соответствие уровня периода ожидания flash-памяти к его значению Уровень «0» $0 < \text{SYSCLK} \leq 24 \text{ MHz}$ Уровень «1» $24 \text{ MHz} < \text{SYSCLK} \leq 48 \text{ MHz}$ Уровень «2» $48 \text{ MHz} < \text{SYSCLK} \leq 72 \text{ MHz}$ Уровень «3»	
47	Показать соответствие битов регистра FLASH_ACR функции управления инициализацией Flash 0÷2 управление задержкой 3 доступ к полному циклу Flash 4 включение буферизации 5 состояние буферизации 6	
48	Можно ли записать данные в память Flash без предварительной очистки ячейки? 1. да 2. нет	2
49	Какое значение битов памяти устанавливается при стирании Flash? 1. 1 2. 0	1
50	Какие страницы памяти МК обычно используют программисты под свои нужды? 1. первые 2. средние 3. последние 4. любые	3
51	Какой способ программирования управления портами ввода-вывода является самым распространенным и удобным? 1. с помощью STM32CubeMX и библиотеки HAL 2. через регистры CMSIS 3. комбинированный	2
52	Какой файл в IDE содержит имена регистров микроконтроллера? 1. stm32f103xb.h 2. stm32f103xx.h 3. system_32f1xb.h	1
53	32-битный конфигурационный регистр для высокопорядковых выводов порта ввода-вывода (GPIO), обычно для выводов с 8-го	1

	по 15-й, который используется для установки их режима работы (вход, выход, альтернативная функция) и скорости, определяя, как эти выводы будут взаимодействовать с внешним миром 1. CRH 2. IDR 3. ODR 4. BSRR 5. BRR 6. LCKR	
54	Специальный регистр памяти, предназначенный для чтения текущего физического состояния (уровня напряжения 1 или 0) выводов порта ввода/вывода (GPIO), к которым подключены внешние сигналы 1. CRH 2. IDR 3. ODR 4. BSRR 5. BRR 6. LCKR	2
55	Специальный регистр портов ввода-вывода (GPIO), который используется для записи данных в выводы порта, устанавливая их в состояние 1 (высокий уровень) или 0 (низкий уровень). 1. CRH 2. IDR 3. ODR 4. BSRR 5. BRR 6. LCKR	3
56	32-разрядный регистр, который позволяет одновременно устанавливать (выставлять в '1') и сбрасывать (выставлять в '0') отдельные выводы порта GPIO, не затрагивая другие выводы 1. CRH 2. IDR 3. ODR 4. BSRR 5. BRR 6. LCKR	4
57	Специальный регистр, который хранит значение делителя частоты, необходимого для установки точной скорости передачи данных (baud rate) по последовательному порту 1. CRH 2. IDR 3. ODR 4. BSRR 5. BRR 6. LCKR	5
58	Регистр, который используется для блокировки конфигурационных битов порта после записи корректной последовательности в 16 бит регистра 1. CRH 2. IDR 3. ODR	6

	4. BSRR 5. BRR 6. LCKR	
59	Показать соответствие между программным продуктом и его назначением и уровнем абстракции STM32CubeMX – графический редактор конфигурации микроконтроллера, генератор начального кода Библиотека HAL – драйверы периферии высокого уровня абстракции Библиотека LL – низкоуровневые драйверы Библиотека SPL	
60	Показать соответствие между программным продуктом и его назначением HAL-драйверы - ориентированы не на периферийные устройства микроконтроллера, а на функциональные возможности этих устройств Драйверы LL - обеспечивают элементарные операции с регистрами периферии микроконтроллера SPL - драйверы	
61	Показать соответствие между программным продуктом и его назначением HAL - это высокоуровневые драйверы, ориентированные на функциональные возможности периферии LL – это драйверы низкого уровня, ориентированные на регистры SPL	
62	Правда ли, что с портами ввода-вывода проще работать через LL-функции, чем с помощью HAL-драйверов? 1. да 2. нет	1
63	Отметить правильные утверждения (несколько вариантов ответа) 1. без события прерывание невозможно 2. без события прерывание возможно 3. события без прерываний могут происходить 4. события без прерываний не могут происходить	1, 3
64	Прием байта портом UART вызывает событие, которое в свою очередь запускает контроллер прямого доступа к памяти (DMA) для пересылки этого байта в память. Вызовет ли это событие прерывание программы? 1. да 2. нет	2
65	При возникновении разрешенного события _____ прерывает выполнение программы, сохраняет в стеке необходимые данные и передает управление по адресу функции обработки прерывания (два слова на русском языке)	контроллер прерываний
66	_____ - ситуация, когда запрещены все преры-	глобальный за-

	вания (два слова на русском языке)	прет
67	_____ - регистр, который может запретить или разрешить отдельные прерывания (два слова на русском языке)	регистр разрешения, регистр маски, регистр масок
68	Каждому из прерываний соответствует свой фиксированный _____, по которому передается управление для обработки прерывания (два слова на русском языке)	адрес-вектор
69	_____ - регистр, который может вызвать прерывание программно (три слова на русском языке)	регистр программных прерываний
70	При использовании таймера в качестве счетчика импульсов можно выбрать один из режимов (несколько вариантов ответа): 1. прямой счет 2. обратный счет 3. двунаправленный счет 4. ненаправленный счет	1, 2, 3
71	. Поставить в соответствие режим счетчика алгоритму его работы прямой счет - содержимое счетчика с каждым импульсом тактирования увеличивается на 1 обратный счет - с каждым входным импульсом содержимое счетчика уменьшается на 1 двунаправленный счет - счетчик считает в прямом направлении от 0 до значения перезагрузки, а затем переходит в реверсивный режим и счет ведется до 0 ненаправленный счет	
72	Поставить в соответствие режим счетчика моменту генерации прерывания прямой счет - в момент перезагрузки формируется прерывание обратный счет - в момент перезагрузки формируется прерывание двунаправленный счет - при изменении направления счета и сбросе генерируется прерывание ненаправленный счет	
73	_____ делит частоту тактирования таймера, поступающую на основной счетчик (одно слово)	Предделитель, Prescaler, PSC
74	Поставить в соответствие название счета режиму счетчика Up – прямой счет. Down – реверсивный счет. Center Aligned mode 1 – двунаправленный счет, прерывание генерируется в момент, когда счетчик считает в обратную сторону и доходит до 0. Center Aligned mode 2 – двунаправленный счет, прерывание генерируется, когда счетчик считает в прямом направлении и достигает значения перезагрузки. Center Aligned mode	
75	Поставить в соответствие название регистра таймера его назначению	

	регистр предделителя - делит частоту тактирования таймера, поступающую на основной счетчик режим счетчика - определяет в какую сторону считать регистр перезагрузки - задает период работы таймера регистр счетчика повторов	
76	UART – это 1. асинхронный интерфейс 2. синхронный интерфейс 3. синхронно-асинхронный интерфейс	1

Тест 3 3 Автоматизация процессов на основе систем с микроконтроллерами STM32

Количество вопросов – 70. Преподаватель формирует нужное количество вариантов теста с любым количеством вопросов.

№ п/п	Вопрос с вариантами ответов	Правильный ответ
1.	Показать соответствие между названием обратной связи и ее содержанием отрицательная обратная связь - обеспечивает подачу на управляемый объект со стороны управляющего устройства команд, направленных на ликвидацию рассогласования действий системы с заданной программой положительная обратная связь - ведет не к устранению, а к усилению рассогласования двунаправленная обратная связь	
2.	Основоположник кибернетики 1. Джон фон Нейман 2. Норберт Винер 3. Клод Шеннон 4. Джон Эккерт 5. Алан Тьюринг 6. Гордон Мур	2
3.	Какие принципы автоматического управления существуют (несколько вариантов ответа)? 1. управление по внешнему воздействию 2. принцип управления по заданию 3. принцип управления по возмущению 4. управление по внутреннему воздействию 5. управление по отклонению 6. управление по ошибке	2, 3, 5. 6
4.	Показать соответствие принципов автоматического управления их содержанию и реализации принцип управления по заданию - реализован в разомкнутых системах автоматического управления или регулирования, ко-	

	которые не требуют контроля поведения выходной величины принцип управления по возмущению - имеют контур управления от объекта управления до сравнивающего устройства, который не выполняет функций обратной связи, т.к. не контролирует выходную величину управление по отклонению или ошибке - контролируется сама выходная величина, вступает в работу контур обратной связи, и ошибка управления выделяется в сравнивающем устройстве управление по внешнему воздействию управление по внутреннему воздействию	
5.	Перечислите этапы алгоритмов и программ для взаимодействия систем на основе микроконтроллеров с пользователем (несколько вариантов ответа) 1. проектирование пользовательского интерфейса 2. разработку логики обработки пользовательских действий 3. реализацию обратной связи 4. описание управляющих воздействий 5. описание выходных сигналов и сигналов обратной связи	1, 2, 3
6.	Что входит в этап проектирования интерфейса (несколько вариантов ответа)? 1. определение пользовательских сценариев 2. описание входных сигналов 3. описание выходных сигналов 4. выбор аппаратных средств интерфейса 5. проектирование логики состояний	1, 4, 5
7.	Показать соответствие этапов обработки ввода информации их назначению обработка ввода - периодическая проверка состояния кнопок, переключателей или сенсорных панелей прерывания - реакция на изменение состояния ввода только тогда, когда оно происходит события	
8.	Для обеспечения низкоуровневого доступа к аппаратному обеспечению и оптимизации производительности программирование микроконтроллеров осуществляется: 1. в машинных кодах 2. на языке ассемблера 3. на языках C и C++	3
9.	Типичная программа для МК имеет основной цикл _____, в котором постоянно проверяются условия и выполняются соответствующие функции (название программного кода на английском языке)	while(1)
10.	Расставьте по порядку этапы разработки пользовательского интерфейса исследование 1 прототипирование 2	

	дизайн 3 тестирование 4 реализация проекта 5 анализ обратных связей и итераций 6	
11.	Показать соответствие принципов работы пользовательского интерфейса их содержанию визуальное представление - User Interface взаимодействует с интерфейсом через устройства ввода, такие как мышь, клавиатура или сенсорный экран обработка действий пользователя - система интерпретирует и выполняет соответствующие команды пользователя: отправку данных, переключение экранов или выполнение заказов обратная связь - помогает пользователю понять, что действия восприняты системой и выполнены навигация - с помощью нее пользователь перемещается между разделами и экранами адаптивность	
12.	Показать соответствие принципов работы пользовательского интерфейса их содержанию адаптивность - UI меняется в зависимости от размера экрана: он разный на компьютерах, планшетах и мобильных устройствах учет пользовательского контента - система может запоминать предпочтения пользователя, предлагать часто используемые функции или адаптировать интерфейс в зависимости от выполняемых задач интерактивность - обеспечивает плавное и быстрое реагирование на действия пользователя визуальное представление	
13.	Показать соответствие между видом пользовательского интерфейса, его содержанием и областью применения графический пользовательский интерфейс (GUI) - элементы отображаются на экране и взаимодействуют с пользователем через мышь, клавиатуру или сенсорный экран мобильные интерфейсы - адаптированы под небольшие экраны и сенсорное управление и обеспечивают удобное взаимодействие с приложениями на мобильных устройствах веб-интерфейсы - включают два основных компонента: элементы навигации и интерактивные компоненты голосовые интерфейсы - обеспечивают естественное и удобное взаимодействие, особенно в ситуациях, когда использование традиционных методов ввода неудобно или невозможно текстовые интерфейсы	
14.	Показать соответствие между видом пользовательского интерфейса и его назначением текстовые интерфейсы - были популярны в ранние годы вычислительной техники и до сих пор используются в специализированных областях, таких как системное администрирование и разработка программного обеспечения жестовые интерфейсы - применяются в устройствах с сенсор-	

	ным вводом и технологией распознавания движений интерфейсы виртуальной и дополненной реальности - используют гарнитуры и контроллеры, позволяя пользователю взаимодействовать с полностью виртуальными объектами консольные интерфейсы - включают элементы навигации, меню и настройки, а также поддерживают различные устройства ввода, такие как игровые контроллеры и сенсорные панели мобильные интерфейсы	
15.	Показать соответствие между компонентами пользовательского интерфейса и их назначением кнопки - переводят по ссылкам и дают пользователю возможность выполнять такие действия, как отправка формы, запуск программы или подтверждение выбора меню - помогают пользователю перемещаться, быстро находить нужные функции и разделы, минимизируя время на поиск информации текстовые поля - используются для ввода и редактирования текста иконки - помогают пользователю всегда находить нужные опции окна и панели	
16.	Показать соответствие между компонентами пользовательского интерфейса и их назначением окна и панели - используются для организации и отображения контента, предоставляют доступ к дополнительным функциям и настройкам формы - используются для сбора данных от пользователя навигационные элементы - помогают пользователям ориентироваться в системе модальные окна - используются для отображения информации или запроса действия от пользователя, блокируют доступ к основному контенту до тех пор, пока пользователь не выполнит действие, не подтвердит или не отменит запрос списки и таблицы	
17.	Показать соответствие между компонентами пользовательского интерфейса и их назначением списки и таблицы - помогает пользователю легко находить модальные интерфейсные функции графики и диаграммы - визуально представляют данные и информацию, помогают пользователям быстро понять и проанализировать данные анимации и переходы - улучшают пользовательский опыт, делая взаимодействие с системой более интуитивным и приятным сообщения и уведомления - информируют пользователя о событиях, таких как ошибки, успехи или важные обновления модальные окна	
18	Показать соответствие методов исследования их содержанию анализ целевой аудитории - определение демографических ха-	

	ра характеристик, поведения и предпочтений пользователей конкурентный анализ - изучение конкурентов для выявления сильных и слабых сторон интервью и опросы - проводится для получения обратной связи и выявления потребностей изучение рынка сбыта	
19.	Показать соответствие методов прототипирования их содержанию создание скетчей - наброски элементов интерфейса и их расположения создание вайрфреймов - детализированные макеты, показывающие структуру и компоновку интерфейса создание интерактивных прототипов - используются инструменты для создания кликабельных прототипов, на которых тестируют навигацию и взаимодействие создание эскизов	
20.	Показать соответствие методов дизайна их содержанию выбор цветовой схемы - определяются главные и дополнительные цвета разработка типографики - выбор шрифтов и стилей для элементов текста создание иконок и графики - разработка всех графических элементов создание макетов - финальный вариант внешнего вида экранов и элементов интерфейса создание опытных образцов	
21.	Показать соответствие методов тестирования их содержанию юзабилити-тестирование - проводятся тесты с участием пользователей для оценки удобства и эффективности интерфейса А/В-тестирование - сравниваются несколько вариантов интерфейса для определения работоспособного решения тестирование на различных устройствах - интерфейсы проверяются на адаптивность и корректную работу итоговое тестирование	
22.	Показать соответствие методов реализации их содержанию создание стилей и компонентов - разработка CSS-стилей, HTML-кода и JavaScript-компонентов для реализации интерфейса интеграция с системой - внедрение интерфейса в продукт и интеграция с бэкенд-системами оптимизация производительности - обеспечение загрузки и работы интерфейса на устройствах выбор производственной базы	
23.	Показать соответствие методов реализации анализа обратной связи и итерации их содержанию сбор обратной связи - использование опросов, интервью и анализа поведения пользователей анализ данных - изучение метрик и аналитики для оценки эф-	

	эффективности интерфейса итерации и улучшения - внесение изменений, на основе полученных данных и обратной связи создание дополнительных обратных связей	
24	Показать соответствие правил и принципов разработки хорошего интерфейса их содержанию простота - упрощение интерфейса до основных элементов поможет пользователю легко и быстро освоить продукт интуитивность - интерфейс должен быть понятным, чтобы пользователи сразу понимали, как с ним взаимодействовать без обучения последовательность - элементы интерфейса должны быть единообразными: иметь одинаковый внешний вид и функционировать одинаково на всех экранах и во всех частях приложения обратная связь - пользователь понимает, что система реагирует на его действия и что происходит в данный момент доступность	
25	Показать соответствие правил и принципов разработки хорошего интерфейса их содержанию эстетика - хороший визуальный дизайн повышает доверие пользователей к продукту и стимулирует их возвращаться к нему снова доступность - учитываются потребности групп, используются контрастные цвета, крупный шрифт, альтернативный текст для изображений и поддержки экранных читателей с ограниченными возможностями скорость - минимизация задержки при загрузке страниц и выполнении операций, чтобы обеспечить комфортное использование адаптивность	
26	Через какие сенсоры осуществляется взаимодействие микроконтроллеров с внешним миром (несколько вариантов ответа)? 1. низкоуровневые 2. высокоуровневые 3. среднеуровневые 4. комбинированные	1, 2
27	Верно ли, что микроконтроллер читает только аналоговые сигналы через свои порты ввода-вывода? 1. да 2. нет	2
28	Верно ли, что микроконтроллер читает только цифровые сигналы через свои порты ввода-вывода? 1. да 2. нет	2
29	Верно ли, что микроконтроллер читает как аналоговые, так и цифровые сигналы через свои порты ввода-вывода? 1. да 2. нет	1

30	К какому уровню взаимодействия относится прямое считывание сигналов с физических датчиков? 1. низкоуровневое 2. высокоуровневое 3. среднеуровневое	1
31	К какому уровню взаимодействия относится использование сенсоров, которые сами выполняют предварительную обработку данных перед передачей микроконтроллеру? 1. низкоуровневое 2. высокоуровневое 3. среднеуровневое	2
32	Отметьте преимущества и недостатки низкоуровневых сенсоров 1. дешевле, меньше, проще в использовании 2. требуют более сложного программирования для обработки всех данных 3. освобождают микроконтроллер от рутинной обработки данных, упрощают разработку 4. дороже, более сложные в настройке, могут быть менее гибкими	1, 2
33	Отметьте преимущества и недостатки высокоуровневых сенсоров 1. дешевле, меньше, проще в использовании 2. требуют более сложного программирования для обработки всех данных 3. освобождают микроконтроллер от рутинной обработки данных, упрощают разработку 4. дороже, более сложные в настройке, могут быть менее гибкими	3, 4
34	Через какие устройства в проводных телекоммуникационных сетях могут быть подключены системы на основе микроконтроллеров к другим сетям (например, к Интернету) (несколько вариантов ответа)? 1. коммутаторы 2. точки доступа 3. концентраторы (хабы) 4. маршрутизаторы 5. шлюзы	4,5
35	Через какие устройства в проводных телекоммуникационных сетях несколько систем на основе МК могут быть объединены в одну локальную сеть (LAN) для совместного использования ресурсов, таких как принтеры или другие устройства (несколько вариантов ответа)? 1. коммутаторы 2. точки доступа 3. концентраторы (хабы) 4. маршрутизаторы 5. шлюзы	1, 3
36	Через какие устройства системы на основе МК могут подключаться к сетям беспроводным способом, например, через Wi-Fi, используя радиосигналы? 1. коммутаторы 2. точки доступа	2

	3. концентраторы (хабы) 4. маршрутизаторы 5. шлюзы	
37	Какие среды передачи данных используются для подключения систем на основе МК к другим системам в телекоммуникационных сетях (несколько вариантов ответа)? 1. медный кабель 2. алюминиевый кабель 3. стальной кабель 4. волоконно-оптический кабель 5. беспроводной радиоэфир	1, 4, 5
38	Какие телекоммуникационные устройства обеспечивают физическое соединение устройств в проводных локальных сетях (несколько вариантов ответа)? 1. коммутаторы 2. точки доступа 3. концентраторы (хабы) 4. маршрутизаторы 5. шлюзы 6. модемы	1, 3
39	Какие телекоммуникационные устройства обеспечивают физическое соединение устройств в сетях Wi-Fi (несколько вариантов ответа)? 1. коммутаторы 2. точки доступа 3. концентраторы (хабы) 4. маршрутизаторы 5. шлюзы 6. модемы	1, 2, 3, 6
40	Подключить микроконтроллер к локальной сети можно с помощью модуля на микросхеме 1. ENC28J60 2. UNC28J60 3. ENC28K60 4. UNC28J60	1
41	Какие уровни модели OSI присутствуют в микросхеме ENC28J60? 1. Прикладной 2. Представления 3. Сеансовый 4. Транспортный 5. Сетевой 6. Канальный 7. Физический	6, 7
42	Обмен данными с контроллером микросхема ENC28J60 осуществляется посредством шины _____ (аббревиатура на английском языке)	SPI
43	Физический уровень микросхема ENC28J60 организован по стандарту _____ (название физической спецификации Ethernet)	10BASE-T
44	Битовая скорость спецификации 10BASE-T	2

	1 2,5 Мбит/с 2 10 Мбит/с 3 100 Мбит/с 4 1000 Мбит/с	
45	Какой адрес необходим, чтобы контроллер мог обращаться к любому компьютеру, как в локальной сети, так и в глобальной? 1 MAC-адрес 2 IP-адрес 3 оба адреса	3
46	Какой интерфейс отладочной платы следит за данными, которые будут приходить из сети в контроллер? 1 USART 2 UART 3 SWD 4 USB 5 SPI	1
47	Сколько контактов используется в интерфейсе USART? 1 2 2 3 3 4 4 5 5 6	3
48	Длина заголовка и концевика кадра Ethernet (без преамбулы) в сумме _____ байт (вставить число)	18
49	Длина MAC-адреса отправителя в кадре Ethernet _____ байт (вставить число)	6
50	Длина MAC-адреса получателя в кадре Ethernet _____ байт (вставить число)	6
51	Длина контрольной суммы в кадре Ethernet _____ байт (вставить число)	4
52	Как называется блок данных, передаваемый на канальном уровне? 1 сообщение 2 сегмент 3 пакет 4 кадр 5 бит	4
53	Как называется блок данных, транспортируемый в качестве данных внутри кадра? 1 сообщение 2 сегмент 3 пакет 4 фрейм	3
54	Назначение протокола ARP 1 преобразование адреса URL в IP-адрес 2 преобразование IP-адреса в MAC-адрес 3 преобразование MAC-адреса в IP-адрес	
55	Как в моделях OSI и TCP/IP называются правила обмена данными между сетевыми узлами на одном уровне? 1 интерфейс 2 протокол	2

	3 стек 4 формат	
56	Как называется блок данных, передаваемый на сетевом уровне? 1 сообщение 2 сегмент 3 пакет 4 кадр 5 бит	3
57	Как называется блок данных, передаваемый на физическом уровне? 1 сообщение 2 сегмент 3 пакет 4 кадр 5 бит	5
58	Как называется блок данных, формируемый на транспортном уровне? 1 сообщение 2 сегмент 3 пакет 4 кадр 5 бит	2
59	Как называется режим работы операционной системы, способный обеспечить требуемое (предсказуемое) время обработки поступающих событий на конкретном оборудовании? 1 пакетный 2 разделения времени 3 реального времени 4 интерактивный	3
60	Установить соответствие между названиями стеков ЦПУ ядра Cortex-M3 и режимами работы основной стек – потоковый (Thread) режим стек процесса – режим обработчика (Handler) режим резервный стек	
61	Верно ли, что в обоих режимах Thread и Handler инструкции выполняются в непривилегированном режиме? 1 Да 2 Нет	2
62	Верно ли, что в обоих режимах Thread и Handler используется основной стек? 1. Да 2. Нет	1
63	Какая ОСРВ является одной из самых популярных, распространяемых бесплатно? 1. Salvo 2. VxWorks 3. FreeRTOS 4. RIOT 5. OS	3
64	heap_1 – предназначена только для создания задач, возможность	

	их удаления отсутствует, невозможно повторно использовать область памяти, невозможно дефрагментировать heap_3 – не выделяет в памяти «кучу», выделение памяти осуществляется разработчиком самостоятельно heap_4 – по умолчанию в STM32CubeMX установлена данная модель heap_5 – позволяет «куче» охватывать несколько несмежных областей памяти heap_2	
65	Частоту переключения задачи обеспечивает параметр 1. TICK_RATE_HZ 2. USE_TICK_HOOK 3. USE_IDLE_HOOK 4. USE_TICKLESS_IDLE	1
66	Вызов функции, которая используется для определения задачи, выполняемой в данный момент, производится активацией параметра 1. TICK_RATE_HZ 2. USE_TICK_HOOK 3. USE_IDLE_HOOK 4. USE_TICKLESS_IDLE	2
67	Вызов функции в моменты, когда процессор не задействован для выполнения какой-либо задачи, производится активацией параметра 1. TICK_RATE_HZ 2. USE_TICK_HOOK 3. USE_IDLE_HOOK 4. USE_TICKLESS_IDLE	3
68	Вызов двух функций, предусмотренных для решения задачи снижения энергопотребления, производится активацией параметра 1. TICK_RATE_HZ 2. USE_TICK_HOOK 3. USE_IDLE_HOOK 4. USE_TICKLESS_IDLE	4
69	Установить соответствие между состояниями задачи и их содержанием READY – задача запущена и готова принять на себя управление RUN – диспетчер переключил управление на задачу, процессор непосредственно работает по ее коду BLOCKED – задача ожидает событие, на которое она должна отреагировать SUSPEND – ни на какие события задача не реагирует, сама из этого состояния выйти не сможет STOP	
70	Верно ли, что диспетчер операционной системы вызывается по срабатыванию таймера, и пока он не будет вызван, ничего в системе не меняется? 1. Да 2. Нет	1

6 Примеры оценочных средств для проведения промежуточной аттестации (8 семестр)

Примеры экзаменационных билетов, 6 семестр – 30 билетов

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 1 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Перечислите и охарактеризуйте этапы построения программ для микроконтроллеров. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу мигания светодиодом Blink (с указанием номеров строк кода). 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 2 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Назовите и охарактеризуйте среды разработки и отладки программ, которые знаете, для микроконтроллеров. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу Led.ino для включения светодиода. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 3 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите абстракции стандартов на интерфейсы ПО микроконтроллеров Cortex (CMSIS, SPL, HAL). ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу blink.ino для реализации мигания светодиодом с разной частотой. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 4 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Охарактеризуйте основные алгоритмические конструкции и их представление. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу sketch-fade.ino для варьирования яркости светодиода. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 5 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните основные понятия и принципы построения диаграммы состояний и конечных автоматов (определения, состояния, переходы, подсостояния, хронологические состояния, общие методики моделирования, проектирование с абстрактными конечными автоматами, цепочки состояний). ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу led_button.ino для управления светодиодом. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 6 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните базовые понятия для программирования на языке СИ (проект, функции, константы, переменные, операторы, циклы). Приведите пример структуры программы. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу debounce.ino переключения светодиода с функцией устранениядребезга контактов кнопки. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 7 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните особенности типов данных языка Си для STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу rgb_nightlight.ino для управления трехцветным светодиодом. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 8 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Перечислите и охарактеризуйте уровни программы на основе MBED (начиная от высокого уровня до самого низкого). ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу pot.ino для считывания выходного аналогового сигнала потенциометра. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 9 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните работу с регистрами STM32 через библиотеку CMSIS. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу tempalert.ino для системы мониторинга температуры. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 10 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните работу с периферийными узлами микроконтроллеры через библиотеку HAL. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу nightlight.ino для автоматического управления яркостью светодиода. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 11 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите процесс конфигурирования микроконтроллера и генерации кода на языке С для ядра Arm Cortex с помощью STM32CubeMX. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу motor.ino для автоматического управления скоростью вращения электродвигателя. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 12 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите процесс создания нового проекта с помощью интерфейса код-генератора STM32CubeMX. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу motor_pot.ino для ручного управления скоростью вращения электродвигателя. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 13 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите процесс конфигурирования системы тактирования и часов реального времени микроконтроллера с помощью STM32CubeMX с внутренним RC-генератором и внешним кварцевым резонатором. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу вспомогательных функций для управления электродвигателем. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 14 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите калькулятор энергопотребления в STM32CubeMX. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу кода основного цикла loop(). 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 15 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите средства разработки System Workbench for STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу hbridge.ino для управления скоростью и направлением вращения электродвигателя. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 16 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите основные особенности средства разработки STM32CubeIDE. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу sag.ino для управления самоходным шасси. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 17 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите создание проекта с использованием STM32CubeIDE. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу servo.ino для управления сервоприводом с помощью потенциометра. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 18 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите пакет программ Proteus Design Suite, предназначенный для автоматизированного проектирования электронных схем. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу сканирующего ИК-дальномера sweep.ino. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 19 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Охарактеризуйте стандартизированное адресное пространство памяти, общее для всех ядер Cortex-M. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте простую программу для управления шаговым электродвигателем stepper.ino. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 20 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните технологию битовых лент (bit-banding). ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте простую программу управления хронографом. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 21 по МДК.02.02 «Программиро- вание микроконтроллеров» специальность: 09.02.01 Ком- пьютерные системы и ком- плексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните систему команд Thumb-2 и выравнивание памяти. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу задания скорости вращения и включения повышающих резисторов. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 22 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Охарактеризуйте систему защиты флеш-памяти микроконтроллера от копирования. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу одноминутного хронографа chronograph.ino 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 23 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите инициализацию флеш-памяти STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу массива для хранения значений нот и их длительности. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 24 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите процессы чтения и записи данных во флеш-памяти STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу music.ino для воспроизведения простой мелодии. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 25 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите использование структур данных во флеш-памяти STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу <code>riano.ino</code> для пятиклавишного пианино. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 26 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите процесс блокировки чтения/записи флеш-памяти STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу rot.ino для тестирования функций print. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 27 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите программирование подсистем ввода/вывода микроконтроллеров через регистры CMSIS. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу pot_tabular.ino для отображения данных в табличном формате. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 28 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите программирование портов ввода-вывода через функции библиотеки HAL. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу echo.ino для возвращения платой Arduino получаемых данных. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 29 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите библиотеку низкоуровневых драйверов LL. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу single_char_control.ino для управления светодиодом с помощью одиночных символов. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 30 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 3	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите совместное использование библиотек HAL и LL. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу list_control.ino для управления разноцветным фотодиодом. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

Примеры экзаменационных билетов, 8 семестр – 35 билетов

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 1 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите систему прерываний STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу Arduino_read_pot.ino для передачи данных на компьютер. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 2 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите организацию и управление прерываниями в STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу <code>processing_display_color.pde</code> для передачи данных на компьютер. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 3 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите таймеры и режимы счета таймера STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу processing_control_RGB.pde для управления разноцветным светодиодом. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 4 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните установку конфигурации таймера с помощью STM32CubeMX. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу csv_logger.ino регистратора данных температуры и освещенности. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 5 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните логику работы прерываний таймеров STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу lock_computer.ino для блокировки компьютера при изменении уровня освещенности. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 6 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Перечислите и опишите регистры таймера STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу alternate.ino для включения светодиодов через один. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 7 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите Управление таймером с помощью библиотеки HAL. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу bargharp.ino для управления гистограммой расстояний. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «_____» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 8 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«_____» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите работу с таймером в режиме счетчика с прерыванием через библиотеку CMSIS. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу lightrider.ino эффекта бегущего светодиода. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 9 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите настройку таймера и прерываний от него через функции библиотеки LL. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу read_temp.ino для опроса датчика температуры. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «____» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 10 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«____» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите использование прерываний UART в STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу temp_unit.ino с использованием датчика температуры, сдвигового регистра и последовательного интерфейса. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «_____» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 11 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«_____» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите протокол интерфейса UART. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу Processing display_temp.pde для вывода на экран значений температуры. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 12 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите протокол интерфейса UART в микроконтроллерах STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу orientation.pde для датчика ориентации на основе акселерометра. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 13 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Перечислите и охарактеризуйте регистры UART в микроконтроллерах STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу instrument.ino для реализации музыкального инструмента на основе акселерометра. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 14 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите инициализацию, установку конфигурации UART в микроконтроллерах STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу LCD_text.ino для вывода простого текста на ЖКД. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 15 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните работа с UART через HAL-библиотеку в блокирующем режиме в микроконтроллерах STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу LCD_progress_bar.ino для вывода на дисплей индикатора хода исполнения. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 16 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните прием данных в UART через HAL-библиотеку в блокирующем режиме в микроконтроллерах STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу кода для формирования специальных символов. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 17 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь «___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите работу с UART через библиотеку HAL с использованием прерываний. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу кода для вывода статического текста. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 18 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите программную модель SPI. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу модифицированной функции устранения дрейфа. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 19 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите программную модель I²C. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу главного цикла программы терморегулятора. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 20 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите программную модель USART. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу управления динамиком и вентилятором. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 21 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь «___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите прямой доступ к памяти в STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу LCD_thermostat.ino для управления регулятором температуры. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 21 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите принципы работы контроллера прямого доступа к памяти в STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу hw_multistatic.ino для демонстрации аппаратных прерываний. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 22 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите каналы и регистры каналов DMA в STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу timer1.ino для проверки работоспособности библиотеки TimerOne. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 23 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите программный модуль контроллера ПДП (DMA1). ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу fun_with_sound.ino для реализации музыкального инструмента. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 24 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите подключение SD-карты к микроконтроллеру. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу write_to_sd.ino для проверки системы SD-карты. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 25 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь «___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите файловую систему FAT на внешней SD-карте. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу чтения данных из файла. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 26 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите процесс настройки FatFs в STM32CubeM. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу sd_read_write.ino для чтения и записи на SD-карту. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 27 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите режимы пониженного энергопотребления в STM32CubeM. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу sd_read_write_rtc.ino для чтения и записи на SD-карту с метками RTC. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 28 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите установку конфигурации АЦП STM32 через регистры CMSIS. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу entrace_logger.ino регистра прохождений через дверь. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 29 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите основные режимы преобразования АЦП STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу rf_test.ino для выполнения простой задачи с использованием радиоканала. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «____» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 30 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«____» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите режим сдвоенных преобразований АЦП STM32. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу doorbell.ino дверного звонка с использованием простого радиоканала. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 31 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите работу АЦП STM32 в режиме оконного коммутатора. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу lamp_remoute.ino для дистанционного управления светильником через радиоканал. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 32 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите работу АЦП STM32 через функции библиотеки HAL. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу сброса модема Bluetooth. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 33 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Опишите подключение STM32 через интерфейс USB. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу считывания входящих данных. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 34 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните структуру управляющих систем и принципы автоматического управления. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу цикла анализа данных. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель Д.И. Головин		

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ КОЛЛЕДЖ СЕРВИСА И ДИЗАЙНА		
Рассмотрено ЦМК Протокол № _____ «___» _____ 2026 г. Председатель ЦМК _____ Е.А. Стефанович	Билет № 35 по МДК.02.02 «Программирование микроконтроллеров» специальность: 09.02.01 Компьютерные системы и комплексы: Группа _____ Курс 4	Утверждаю Зам. Директора по УР А.Т. Бондарь
		«___» _____ 2026 г.
Компетенции: ПК 2.1-2.5, ОК 1-9 Инструкция 1. Внимательно прочитайте задание. Текст задания: ЗАДАНИЕ № 1 - теоретическое. Объясните принципы взаимодействия микроконтроллера с внешними устройствами. ЗАДАНИЕ № 2 - практическое. Напишите и прокомментируйте программу установки уровня мощности. 2. Задание выполнить: • в виде устного ответа на теоретический вопрос; • в среде разработки Arduino IDE. Время выполнения задания - 2 академических часа Преподаватель _____ Д.И. Головин		

5.1 Ответы на вопросы для собеседования (устного опроса):

1 Особенности программирования микроконтроллеров STM32/ATmega328

Тема 1.1 Принципы построения программ для микроконтроллеров. Средства разработки и отладки.

1. Принципы построения программ для микроконтроллеров (МК) базируются на работе в реальном времени, жестких ограничениях ресурсов (память, скорость) и прямом управлении «железом». Основные подходы включают модульность, использование прерываний, конечные автоматы (FSM), основы оптимизации, циклическое выполнение (super-loop) и надежность.

2. Среды разработки (IDE) для STM32 включают как официальные бесплатные решения от STMicroelectronics (STM32CubeIDE), так и профессиональные коммерческие (Keil MDK, IAR EWARM), а также популярные связки с VS Code (PlatformIO). Большинство IDE поддерживают отладку через ST-Link.

3. Первая и самая низкая абстракция CMSIS (Cortex Microcontroller Software Interface Standard) — это CMSIS-Core, обеспечивающая прямой, стандартизированный доступ к регистрам периферии ядра Cortex-M и системным функциям. Она упрощает работу с прерываниями, NVIC, MPU и тактированием, предоставляя имена структур вместо абсолютных адресов памяти.

4. SPL (Standard Peripherals Library) — это библиотека промежуточного уровня (между регистрами CMSIS и HAL), предоставляющая Си-функции для управления периферией. Она обеспечивает высокую производительность и наглядность управления периферией через структуры (инициализация) и функции (управление), скрывая прямую работу с битами регистров, но оставаясь привязанной к конкретному семейству.

5. Третья абстракция в экосистеме STM32Cube — это уровень LL (Low-Layer) драйверов, разработанный для обеспечения максимальной производительности и минимального размера кода. В отличие от HAL (Hardware Abstraction Layer), LL-драйверы представляют собой тонкую обертку над регистрами, обеспечивая переносимость, но требуя от разработчика глубокого понимания работы периферии.

Тема 1.2 Правила составления алгоритмов. Типы алгоритмов. Диаграммы состояний. Конечный автомат

6. Алгоритм — это точная последовательность действий, преобразующая исходные данные в результат за конечное число шагов. Его ключевые свойства включают дискретность (шаговость), детерминированность (однозначность), результативность (конечность) и массовость (универсальность). Алгоритм должен быть понятен исполнителю и формально определен.

7. Формы представления алгоритма — это способы записи последовательности действий для решения задачи. Основные формы включают: словесную (естественный язык), графическую (блок-схемы), псевдокод (полуформальный язык) и программную (языки программирования). Они различаются степенью формализации, наглядности и применимости.

8. Структуры представления алгоритма:

- Линейная структура является основной. Она означает, что действия выполняются друг за другом.
- Разветвляющаяся структура (ветвление) — это структура, обеспечивающая альтернативный выбор в зависимости от заданного условия.
- Циклическая структура (или повторение) предусматривает повторное выполнение некоторого набора действий. Циклы позволяют записать длинные последовательности операций обработки данных с помощью небольшого числа повторяющихся команд.

9. Цикл с предусловием начинается с проверки логического выражения «Р». Если оно истинно, то выполняется «F», затем все повторяется снова, до тех пор, пока логическое выражение сохраняет значение «истина». Как только оно становится ложным, выполнение операций «F» прекращается и управление передается по программе дальше.

Так как выражение, управляющее циклом, проверяется в самом начале, то в случае, если условие сразу окажется ложным, операторы циклической части «F» могут вообще не выполняться. Операторы циклической части «F» должны изменять переменную (или переменные), влияющую на значение логического выражения, иначе программа «зациклится» — будет выполняться бесконечно. Рассмотренная циклическая конструкция называется цикл «пока», или цикл с предусловием.

10. Цикл с постусловием предусматривает проверку условия после выполнения команд, встроенных внутрь цикла.

11. В отдельную категорию можно выделить алгоритмическую конструкцию «цикл с параметром». В данной конструкции присутствует переменная, называемая «счетчиком», с заданным начальным и конечным значениями, а также величиной переменной на каждом шаге выполнения цикла. Цикл выполняется, пока счетчик не достигнет своего максимального значения.

12. Конструкция «цикл, вложенный в неполную развилку» — это структура, где цикл (повторение действий) выполняется только при истинности условия неполного ветвления (конструкция «ЕСЛИ-ТО»). Если условие ложно, цикл пропускается. Это позволяет избирательно запускать итерационные процессы, обрабатывая данные только при соблюдении специфического критерия.

13. «Цикл в цикле» (вложенные циклы) — это алгоритмическая конструкция, при которой тело внешнего цикла содержит внутренний цикл. Для каждого шага внешнего цикла внутренний цикл выполняется полностью. Это используется для

работы с многомерными данными (таблицы, матрицы) или комбинаторными задачами, где число итераций равно $N \times M$.

14. «Развилка в развилке» (или вложенное ветвление) — это алгоритмическая конструкция, где внутри одной ветви (полной или неполной) находится еще один оператор ветвления. Она позволяет обрабатывать сложные условия, когда выбор дальнейших действий зависит от нескольких последовательных проверок, где второй выбор логически следует из первого.

15. Трехкратное вложение структур (например, вложенные циклы for) — это алгоритмическая конструкция, используемая для перебора трехмерных данных или выполнения трехступенчатых зависимых действий. В ней внешний цикл выполняет итерацию i , средний — j , внутренний — k , образуя кубическую временную сложность $O(n^3)$.

16. Диаграмма конечных состояний (State Machine Diagram) — это UML-диаграмма, описывающая поведение объекта или системы через ограниченный набор состояний, переходы между ними и реакцию на внешние события. Она моделирует весь «жизненный цикл» элемента, показывая, как объект меняет состояние от начального до конечного.

17. Конечный автомат (FSM — Finite-state machine) — это математическая модель, описывающая систему, которая в каждый конкретный момент времени находится в одном из конечного числа состояний. Под воздействием входных сигналов автомат переключается между состояниями согласно заданным правилам.

18. Переходы конечного автомата (КА) — это правила, определяющие смену текущего состояния на следующее под влиянием входных сигналов. Ключевые свойства включают детерминизм (или его отсутствие), дискретность, зависимость от текущего входа, возможность самопроизвольных переходов и представление в виде графов или таблиц. Переходы определяют логику работы системы.

19. События триггера конечного автомата (FSM) — это внешние или внутренние сигналы, инициирующие переход системы из одного состояния в другое. Они выступают «причиной» изменения поведения системы, определяя, когда именно срабатывает логика перехода. События могут включать сигналы, вызовы процедур, временные интервалы или изменения состояния данных.

20. Хронологические состояния (или исторические состояния, англ. history states) в конечных автоматах — это специальные псевдосостояния, используемые в составных (вложенных) состояниях для запоминания активного подсостояния при выходе из него. Они позволяют при повторном входе в составное состояние вернуться к тому подсостоянию, которое работало последним, а не начинать работу с начального подсостояния.

21. Абстрактный конечный автомат (FSM — Finite State Machine) — это математическая модель, описывающая систему, которая находится в одном из конечного числа состояний в каждый момент времени. Он переходит из одного состояния в другое под воздействием внешних сигналов (входных символов), определяя поведение логических схем, алгоритмов или ИИ.

22. Цепочки состояний конечного автомата — это последовательность переходов из одного состояния в другое, вызванная рядом входных сигналов (событий). Она описывает логику работы системы от начального состояния до конечного (или промежуточного), определяя путь выполнения программы.

Тема 1.3 Особенности синтаксиса для программ на МК

23. При программировании STM32 (ARM Cortex-M) в Си критически важно использовать фиксированные типы данных из `<stdint.h>` (`uint32_t`, `int16_t` и т.д.) для гарантии размера переменных (8, 16, 32, 64 бита), так как стандартный `int` может варьироваться. Это обеспечивает переносимость кода и точную работу с 32-битными регистрами периферии.

2 Модульное программирование микроконтроллеров STM32

Тема 2.1 Низкоуровневые библиотеки для платформы ARM Cortex (STM32)

24. Для программирования микроконтроллеров STM32 (ARM Cortex-M) используются библиотеки от производителя (STMicroelectronics) и сторонние решения, обеспечивающие разный уровень абстракции: от прямого управления регистрами до Hardware Abstraction Layer (HAL). Основные библиотеки включают CMSIS, HAL, LL, SPL, а также альтернативы вроде `libopencm3`.

25. Программирование микроконтроллеров STM32 (ARM Cortex-M) строится на многоуровневой архитектуре абстракций, позволяющей балансировать между производительностью и простотой разработки. Основные уровни включают прямое управление регистрами (Register-level), библиотеки CMSIS, HAL (Hardware Abstraction Layer) и LL (Low-Layer), а также высокоуровневые операционные системы (RTOS) и интерпретируемые языки.

26. Работа с регистрами STM32 через CMSIS — это прямой, низкоуровневый метод программирования, предоставляющий максимальную производительность и минимальный размер кода. CMSIS (Cortex Microcontroller Software Interface Standard) стандартизирует доступ к периферии через Си-структуры, обеспечивая быстрый доступ к RCC, GPIO и таймерам, требуя детального изучения документации (Reference Manual).

27. Библиотека HAL (Hardware Abstraction Layer) для STM32 — это официальный уровень абстракции от STMicroelectronics, упрощающий разработку за счет унифицированных функций для всех семейств MCU. Она ускоряет написание кода,

обеспечивает переносимость между чипами и работает в связке с STM32CubeMX для автоматической генерации конфигурации.

28. CMSIS — это базовый, низкоуровневый стандарт (работа с регистрами напрямую), обеспечивающий высокую производительность и компактность кода. HAL (Hardware Abstraction Layer) — это высокоуровневая библиотека, скрывающая детали регистров, ускоряющая разработку и обеспечивающая переносимость между сериями STM32, но создающая "монструозный" код.

Тема 2.2 Средства программирования и отладки

2.2.1 Конфигуратор STM32CubeMX

29. STM32CubeMX — это графический инструмент настройки и генерации кода от компании STMicroelectronics, предназначенный для упрощения разработки программ для микроконтроллеров STM32. Он позволяет визуальным образом сконфигурировать выводы (GPIO), тактирование (Clock Tree) и периферию, автоматически создавая инициализационный код на C/C++ для различных IDE.

30. Генерация кода в STM32CubeMX — это графический процесс настройки периферии, тактирования и выводов микроконтроллера STM32 с автоматическим созданием Си-кода инициализации (HAL/LL). Основные этапы: выбор MCU, настройка Pinout & Configuration, настройка Clock, настройка проекта в Project Manager и нажатие «GENERATE CODE» для создания проекта под IDE.

31. Интерфейс STM32CubeMX — это графический конфигуратор, ориентированный на визуализацию настройки периферии, тактирования и выводов (pinout) микроконтроллеров STM32. Он упрощает процесс инициализации, автоматически генерируя C-код (HAL/LL), и разделен на логические вкладки: Pinout & Configuration, Clock Configuration, Project Manager.

32. Конфигурация тактирования в STM32CubeMX (вкладка «Clock Configuration») — это визуальный инструмент для настройки RCC (Reset and Clock Control), позволяющий задать частоту ядра и периферии. Он позволяет выбрать источник (HSI, HSE, PLL), настроить делители/множители и автоматически рассчитать безопасные значения частот, избегая ошибок в регистрах.

2.2.2 Средства разработки System Workbench for STM32

33. System Workbench for STM32 (SW4STM32) — это бесплатная интегрированная среда разработки (IDE) на базе Eclipse, созданная для программирования микроконтроллеров STM32. Она предоставляет полный набор инструментов (компилятор, отладчик, редактор) для создания C/C++ приложений, поддерживая все семейства STM32 и отладочные платы

34. System Workbench for STM32 (SW4STM32) — это интегрированная среда разработки (IDE) на базе Eclipse, разработанная AC6 и поддерживаемая STMicroelectronics. Она обеспечивает полный цикл разработки, включая компиляцию (GCC), отладку (OpenOCD/GDB) и генерацию кода, ориентированную на 32-битные микроконтроллеры семейства ARM Cortex-M.

35. System Workbench for STM32 (SW4STM32) — это бесплатная интегрированная среда разработки (IDE) на базе Eclipse, предназначенная для создания приложений для микроконтроллеров STM32 на базе ARM Cortex-M. Процесс включает настройку периферии в STM32CubeMX, генерацию кода, написание логики, компиляцию и отладку через ST-Link.

2.2.3 Proteus Design Suite

35. Proteus Design Suite — это комплексная система автоматизированного проектирования (САПР) электронных схем, разработанная компанией Labcenter Electronics. Она объединяет симулятор работы электронных схем (ISIS) и систему проектирования печатных плат (ARES), позволяя пройти путь от схемотехники до визуализации готовой платы.

Тема 2.3 Программирование систем микроконтроллера

2.3.1 Программирование памяти микроконтроллера

36. Архитектура ARM Cortex-M использует унифицированное 32-битное адресное пространство объемом 4 ГБ (от 0x00000000 до 0xFFFFFFFF), разделенное на четко определенные области. В отличие от архитектуры x86, оно фиксировано для всех процессоров семейства, что облегчает перенос кода. Основные зоны включают код, данные, периферию и системные регистры.

37. Карта памяти кода STM32 (на базе ARM Cortex-M) — это часть 4-гигабайтного линейного адресного пространства, где основная Flash-память программ обычно начинается с адреса 0x08000000. Она включает таблицу векторов прерываний, код программы, константы и загрузчик, обеспечивая энергонезависимое хранение команд процессора.

38. Выбор COM-порта в Arduino IDE осуществляется через меню Инструменты (Tools) > Порт (Port). Необходимо выбрать порт, рядом с которым указано имя вашей платы (например, COM3 (Arduino Uno)), при этом рекомендуется выбирать вариант с символом USB.

39. Bit-banding (битовые ленты) — это аппаратная технология в микроконтроллерах (преимущественно ARM Cortex-M3/M4), позволяющая атомарно читать или записывать отдельные биты в оперативной памяти (SRAM) или периферийных регистрах, обращаясь к ним как к целым 32-битным словам.

40. Карта памяти — это структурированная схема распределения адресного пространства микроконтроллера между RAM, Flash, периферией и регистрами. Бито-

вые ленты (Bit-banding) — технология, позволяющая обращаться к отдельным битам памяти или периферии, сопоставляя их с целыми 32-битными словами в специальной области адресов, что делает манипуляцию битами атомарной и быстрой.

41. Защита флеш-памяти STM32 от копирования осуществляется установкой битов защиты чтения (RDP — Read Protection) в Option Bytes. Основные уровни: Level 0 (нет защиты), Level 1 (чтение запрещено при отладке) и Level 2 (отладка отключена навсегда). Также применяются защита записи (WRP) и, в новых сериях, механизмы Secure Boot/Secure Firmware Update.

2.3.2 Программирование Flash памяти STM32

42. Flash-память в микроконтроллерах STM32 необходима как энергонезависимое хранилище программного кода (прошивки), обеспечивающее его сохранение при выключении питания. Она оптимизирована для быстрого чтения инструкций ядром Cortex-M, поддерживает безопасное обновление кода (bootloader) и позволяет хранить пользовательские конфигурационные данные, заменяя EEPROM.

43. Flash-память в микроконтроллерах STM32 (встроенная NOR Flash) отличается высоким быстродействием, встроенной защитой кода и возможностью работы на высоких частотах без задержек (на ряде серий). Основными плюсами являются надежность, энергонезависимость и удобство работы, однако она имеет ограниченное число циклов перезаписи и специфические особенности стирания секторами.

44. Flash-память в микроконтроллерах STM32 — это энергонезависимая память, организованная в виде трех блоков: основной памяти (Main Memory) для кода, информационного блока (System Memory/Option Bytes) и регистров управления (FPEC). Она позволяет хранить прошивку, поддерживая посекторное или страничное стирание.

45. Операции с Flash-памятью STM32 включают инициализацию (настройку латентности), чтение (прямой доступ), стирание (страницами/массовое) и запись (программирование слов/полуслов). Ключевой особенностью является необходимость разблокировки контроллера Flash (FPEC) через регистры FLASH_KEYR перед записью/стиранием и работа с разделением памяти на страницы.

46. Использование структур данных во Flash-памяти STM32 позволяет энергонезависимо хранить конфигурации, калибровки и журналы. Основным принципом заключается в записи C-структур в выделенные страницы Flash-памяти, что требует предварительного стирания сектора (erase) и побайтовой записи (program) с использованием HAL-библиотек (например, HAL_FLASH_Unlock, HAL_FLASH_Program).

47. Блокировка чтения/записи Flash-памяти в STM32 выполняется через регистры контроллера Flash (FLASH_CR/FLASH_OPTCR), обеспечивая защиту от записи

(WRP) и чтения (RDP). Запись блокируется сбросом бита LOCK в FLASH_CR, а защита от чтения/записи (уровень 1 или 2) — установкой опциональных байтов (Option Bytes).

48. STM32 ST-LINK Utility — это официальная бесплатная утилита от STMicroelectronics, предназначенная для внутрисхемного программирования, прошивки, отладки и проверки памяти микроконтроллеров STM32 и STM8 через программаторы ST-LINK/V2/V3. Основные функции включают загрузку бинарных файлов (.bin, .hex), стирание flash-памяти, сравнение данных и настройку защиты.

2.3.3 Программирование подсистем ввода/вывода микроконтроллеров через регистры CMSIS

49. Программирование подсистем ввода-вывода (GPIO) через регистры CMSIS (Cortex Microcontroller Software Interface Standard) обеспечивает максимальную производительность и минимальный размер кода за счет прямого доступа к аппаратным средствам, но требует глубокого понимания архитектуры микроконтроллера, увеличивает время разработки и снижает переносимость кода между разными семействами MCU.

2.3.4 Программирование портов ввода-вывода через функции библиотеки HAL

50. Конфигурация портов (GPIO) в STM32 с использованием библиотеки HAL включает включение тактирования порта, заполнение структуры GPIO_InitTypeDef (режим, скорость, подтяжка) и вызов функции HAL_GPIO_Init(). Это обеспечивает настройку регистра вывода, переключение состояний и обработку прерываний.

2.3.5 Программирование портов ввода-вывода через LL-функции

51. HAL (Hardware Abstraction Layer) — это высокоуровневая, переносимая между семействами библиотека, упрощающая разработку, но требовательная к памяти. LL (Low-Layer) — это низкоуровневые, легкие и быстрые драйверы, работающие почти напрямую с регистрами, обеспечивая высокую производительность и минимальный размер кода.

52. Совместное использование HAL (Hardware Abstraction Layer) и LL (Low-Layer) в STM32 позволяет сочетать высокую скорость разработки с оптимизацией производительности. HAL применяется для инициализации и сложных периферийных узлов (USB, Ethernet), а LL — для критичных по времени задач, обеспечивая быстрый доступ к регистрам и малый размер кода.

53. Управление портами ввода-вывода (GPIO) через библиотеку HAL (Hardware Abstraction Layer) от STMicroelectronics сводится к трем основным этапам: инициализация, чтение/запись и обработка прерываний.

2.3.6 Система прерываний. Организация и управление прерываниями

54. Аппаратные прерывания — это принудительные сигналы от периферии (таймер, клавиатура), заставляющие процессор немедленно приостановить текущую работу и выполнить обработчик. Событие — это факт изменения состояния (нажатие кнопки, изменение уровня напряжения), который может не требовать немедленной реакции процессора.

55. Контроллер прерываний в STM32 (NVIC — Nested Vectored Interrupt Controller) — это встроенный в ядро ARM Cortex-M аппаратный блок, обеспечивающий приоритетную обработку событий. Он поддерживает вложенные прерывания (одно может прервать другое), имеет фиксированные векторы для быстрого вызова обработчиков, позволяет настраивать уровни приоритета (0–15) и маскировать отдельные источники.

56. Программное управление прерываниями в STM32 (на базе Cortex-M) — это процесс настройки NVIC (Nested Vectored Interrupt Controller) для обработки сигналов периферии. Он включает включение линий EXTI/периферии, установку приоритетов в NVIC, написание короткого обработчика (ISR) и сброс флага прерывания в конце.

57. Функции обработки прерываний (ISR) в STM32 — это специальные подпрограммы, выполняемые асинхронно при возникновении событий (таймер, GPIO, UART). Они требуют сброса флага источника (например, `_HAL_TIM_CLEAR_IT` или `_HAL_GPIO_EXTI_CLEAR_IT`), минимального времени выполнения, использования `volatile` для переменных и вызова callback-функций (например, `HAL_GPIO_EXTI_Callback`) для обработки логики.

2.3.7 Таймеры в режиме счетчиков. Генерация циклических прерываний от таймеров

58. Таймеры в микроконтроллерах STM32 делятся на базовые, общего назначения, расширенные и низкопотребляющие (LPTIM), отличаясь разрядностью (16/32 бит), количеством каналов и функционалом (ШИМ, захват, энкодер). Основные серии включают TIM1/TIM8 (расширенные), TIM2-TIM5 (общего назначения, 32-бит) и TIM6/TIM7 (базовые).

59. Таймеры STM32 называют многофункциональными, потому что они представляют собой сложные аппаратные блоки, способные работать автономно от ядра процессора. Они выполняют не только счет времени, но и генерируют ШИМ-сигналы, измеряют длительность импульсов, считают внешние события и синхронизируют работу других периферийных устройств (например, АЦП).

60. Функциональная схема таймера STM32 — это высокоточный аппаратный счетчик, который увеличивает или уменьшает свое значение с каждым тактом, поступающим через делитель частоты (Prescaler). Когда счетчик (CNT) достигает значения в регистре автоперезагрузки (ARR), генерируется событие обновления (прерывание/DMA), и счетчик сбрасывается или меняет направление.

61. Таймеры STM32 работают в нескольких основных режимах: базовый счетчик (временные задержки), входной захват (измерение сигналов), выходное сравнение (генерация событий), ШИМ (управление мощностью), режим энкодера и одиночного импульса. Таймеры поддерживают прямой, реверсивный и центрированный счета.

62. Настройка таймера в STM32CubeMX включает выбор таймера (TIMx), активацию внутреннего источника тактирования (Clock Source), настройку предделителя (Prescaler - PSC) и регистра автоперезагрузки (Counter Period - ARR) для задания частоты, а также включение прерываний (NVIC) при необходимости, завершаясь генерацией кода HAL.

63. Prescaler (PSC) в таймерах STM32 — это 16-битный аппаратный делитель частоты, который уменьшает входную частоту шины (например, МГц) до требуемой частоты счета основного регистра CNT. Он делит входную частоту на $(PSC + 1)$, обеспечивая гибкую настройку времени между тиками таймера.

64. Режим счетчика в таймерах STM32 — это базовая функция, определяющая направление счета (вверх, вниз или двунаправленный) 16/32-битного регистра CNT. Таймер увеличивает/уменьшает значение на основе тактового сигнала, деленного предделителем PSC

65. Регистр счетчика повторов (TIMx_RCR — Repetition Counter Register) в таймерах STM32 (преимущественно в продвинутых TIM1/TIM8) — это 8-битный регистр, позволяющий генерировать события обновления (прерывания, DMA) не каждый период таймера, а после $N + 1$ переполнений. Он используется для деления частоты прерываний, управления ШИМ и многофазными двигателями.

2.3.8 Конфигурация таймеров с помощью библиотек CMSIS, HAL и LL

66. Прерывания таймеров STM32 работают по принципу отсчета заданного количества импульсов (тактов) и вызова обработчика (функции) при переполнении счетчика. Логика строится на трех компонентах: предделителе (PSC) частоты, счетчике (CNT) и регистре автоперезагрузки (ARR). Когда счетчик достигает значения ARR, генерируется флаг обновления, вызывающий прерывание, после чего счетчик сбрасывается.

67. Регистры STM32 — это 32-битные ячейки памяти, управляющие периферией и логикой микроконтроллера (GPIO, таймеры, АЦП, UART) через отображение в

адресное пространство (MMIO). Они делятся на управляющие (настройка), регистры состояния (флаги) и регистры данных (чтение/запись значений). Основная работа идет с GPIO (MODER, OTYPER), RCC (тактование) и NVIC (прерывания).

68. Управление таймерами в STM32 через HAL (Hardware Abstraction Layer) осуществляется через структуру TIM_HandleTypeDef и набор функций HAL_TIM_xxx. Настройка обычно выполняется в STM32CubeMX, после чего используются функции для запуска/остановки (Base, IT, PWM, IC) и обработки прерываний (HAL_TIM_PeriodElapsedCallback).

69. Настройка таймера и его прерываний (ISR) через библиотеку LL (Low-Layer) в STM32 обеспечивает высокую производительность и малый объем кода. Процесс включает тактирование таймера, настройку делителя (PSC) и периода (ARR), включение прерывания по обновлению (UIE), настройку NVIC и запуск таймера функциями LL_TIM_EnableCounter и LL_TIM_EnableIT_UPDATE.

2.3.9 Работа с интерфейсом UART через регистры CMSIS. Использование прерывания UART

70. UART (Universal Asynchronous Receiver/Transmitter) — это асинхронный последовательный интерфейс, преобразующий параллельные данные от ЦП в последовательный поток для передачи по одному проводу (TX) и обратно (RX) без использования общей тактовой линии. Данные передаются «кадрами» (обычно 8 бит) со старт/стоп битами, требуя предварительной настройки одинаковой скорости передачи (баудрейт).

71. UART (Universal Asynchronous Receiver-Transmitter) в STM32 — это периферийный модуль для полнодуплексной асинхронной связи. Он поддерживает дробный делитель частоты для точной настройки скорости (Baud Rate), работу с DMA для передачи данных без нагрузки процессора и обнаруживает ошибки (переполнение, шум, фрейм, четность). Настройка обычно включает UART1-5, работу через CMSIS или библиотеку HAL.

72. Регистры UART/USART в STM32 (расположенные в структуре USART_TypeDef) управляют асинхронной и синхронной передачей данных. Основные регистры включают USART_CR1/2/3 (настройка), USART_BRR (скорость), USART_DR (данные), USART_SR (флаги состояния). Они позволяют задавать параметры кадров, делители частоты, управлять прерываниями и DMA.

73. Прерывания UART в STM32 обеспечивают асинхронную обработку данных, позволяя микроконтроллеру не блокировать выполнение основного кода (polling) при приеме/передаче байтов. Они активируются в NVIC, работают через вектор прерываний (USARTx_IRQn), вызывая функции обратного вызова HAL_UART_RxCpltCallback или HAL_UART_TxCpltCallback для обработки входящих данных или завершения передачи.

2.3.10 Работа с UART через библиотеку HAL. Инициализация интерфейса и передача данных в блокирующем режиме. Отладка программ с помощью UART. Функция `sprintf`

74. Инициализация UART в STM32 включает включение тактирования, настройку GPIO (Tx/Rx), конфигурацию параметров UART (скорость, длина данных, четность) и включение самого модуля, чаще всего через библиотеку HAL (Hardware Abstraction Layer). В STM32CubeMX это делается автоматически при настройке, а в коде требует инициализации дескриптора `UART_HandleTypeDef`.

75. Работа с UART через HAL-библиотеку в блокирующем режиме — это самый простой способ передачи/приема данных, при котором процессор «зависает» (блокируется) в функции до тех пор, пока данные полностью не отправятся или не примутся. Функции `HAL_UART_Transmit` и `HAL_UART_Receive` с заданным тайм-аутом используются для реализации этого механизма.

76. Отладка STM32 через последовательный порт (UART) выполняется путем отправки отладочных сообщений (`printf` в UART) на ПК и их просмотра в терминале (например, Tera Term, PuTTY). Это позволяет отслеживать значения переменных и выполнение кода в реальном времени, используя преобразователь USB-to-UART (например, CP2102 или FT232), подключенный к выводам TX/RX микроконтроллера.

77. Функция `sprintf` в STM32 (и в Си в целом) — это мощный инструмент форматирования строк, который вместо вывода текста на консоль (как `printf`) сохраняет его в символьный буфер (массив `char`). Она используется для преобразования чисел, переменных и текста в единую строку перед отправкой через UART, отображением на дисплее или сохранением.

78. Для вывода чисел с плавающей запятой (`float`, `double`) через `sprintf` в микроконтроллерах (AVR, STM32) используются спецификаторы `%f`, `%e`, `%g`. Основная проблема — высокая нагрузка на память, из-за чего в средах (IDE) поддержка `float` в `printf/sprintf` часто отключена по умолчанию.

2.3.11 Программирование последовательных портов SPI, I²C, USART

79. Программная модель SPI в STM32 основана на использовании периферийного модуля SPI, управляемого через регистры конфигурации (CR1, CR2), регистр данных (DR) и регистр статуса (SR). Он поддерживает полнодуплексный синхронный обмен, режимы Master/Slave, управление через DMA и прерывания, а также позволяет настраивать полярность/фазу, скорость (делитель) и порядок битов (MSB/LSB).

80. Программная модель I²C в STM32 базируется на аппаратном контроллере (I²C IP block), который управляет тактированием (SCL) и данными (SDA). Она включает Master/Slave режимы, 7/10-битную адресацию, режимы скоростей (Standard,

Fast, Fast+), DMA и обширную систему флагов событий (START, ADDR, TXE, RXNE).

81. Программная модель USART в STM32 основана на управлении регистрами периферийного модуля, который обеспечивает полнодуплексный (или полудуплексный) обмен данными

2.3.12 Прямой доступ к памяти в STM32. Контроллер DMA

82. Прямой доступ к памяти (DMA, Direct Memory Access) — это технология, позволяющая периферийным устройствам (дискам, сетевым картам) обмениваться данными напрямую с оперативной памятью (ОЗУ), минуя центральный процессор (ЦП). Это освобождает процессор от рутинной пересылки данных, значительно повышая общую производительность системы.

83. Контроллер прямого доступа к памяти (DMA) в STM32 — это аппаратный модуль, обеспечивающий автономную высокоскоростную передачу данных между периферией и памятью (или память-память) без участия центрального процессора. DMA освобождает процессор для выполнения задач, работая параллельно, и сообщает о завершении передачи через прерывания.

84. Каналы DMA в STM32 (Direct Memory Access) — это независимые аппаратные блоки, обеспечивающие передачу данных между памятью (RAM, Flash) и периферией (USART, SPI, ADC) без участия ЦП. Обычно DMA1 (7 каналов) и DMA2 (5 каналов) работают с разными шинами АНВ, имеют 4 уровня приоритета (Low to Very High) и поддерживают режимы циклической или однократной передачи.

85. Регистры каналов DMA (1-7) в STM32 (например, серии F1/F3/F4) обеспечивают независимую настройку передачи данных. Основные регистры: DMA_CPARx (адрес периферии), DMA_CMARx (адрес памяти), DMA_CNDTRx (количество данных), DMA_CCRx (управление/конфигурация) и регистры статуса/очистки прерываний (ISR/IFCR).

86. Конфигурация DMA в STM32 включает включение тактирования, настройку источника/приемника (периферия-память), размера данных, инкремента адресов, режима работы (циклический/одиночный) и включение канала. Процесс выполняется через библиотеку HAL (функции HAL_DMA_Init) или низкоуровневую LL.

87. Доступ к периферии в STM32 осуществляется через многоуровневую матрицу шин (АНВ/APB), где высокоскоростные блоки (DMA, Flash) подключены к АНВ, а низкоскоростные — к APB1/APB2. Специфика включает необходимость включения тактирования (RCC), учет разности скоростей шин, использования DMA для АНВ и соблюдение 32-битного выравнивания при доступе.

88. В контроллерах STM32 DMA-каналы (потoki) генерируют три основных типа прерываний для управления передачей данных: окончание передачи (ТС - Transfer

Complete), половина передачи (HT - Half Transfer) и ошибка передачи (TE - Transfer Error).

2.3.13 Работа с контроллером DMA через CMSIS регистры

89. DMA (Direct Memory Access) в STM32 позволяет передавать данные из памяти (RAM/Flash) в порт ввода-вывода (GPIO) без участия процессора, освобождая его ресурсы. Это происходит путем настройки канала DMA на копирование массива данных по адресу регистра GPIOx_ODR при поступлении запроса, обычно от таймера, обеспечивая высокую скорость передачи.

90. Программное управление генератором импульсов (ШИМ/PWM) в STM32 осуществляется через периферийные модули таймеров (TIMx) с использованием библиотеки HAL. Ключевые функции включают инициализацию (HAL_TIM_PWM_ConfigChannel), запуск (HAL_TIM_PWM_Start), динамическое изменение скважности (__HAL_TIM_SET_COMPARE) и частоты (__HAL_TIM_SET_PRESCALER).

91. Управление генератором сигналов (ШИМ/DAC) на STM32 с компьютера реализуется через интерфейс связи (USB, UART) с использованием таймеров. Компьютер отправляет параметры (частота, скважность), STM32 принимает их, обновляет регистры таймера (CCR) или ЦАП, меняя форму сигнала на пинах GPIO. Используется CubeMX для настройки и HAL для программирования.

92. Программа верхнего уровня для генератора импульсов на STM32 — это прикладной код (часто на C/C++ с использованием HAL), который управляет периферией таймеров, задавая параметры частоты, скважности и количества импульсов без прямого обращения к регистрам. Она обеспечивает удобный интерфейс (например, через UART или структуру данных) для изменения характеристик ШИМ/генерации "на лету".

2.3.14 Программный модуль контроллера ПДП (DMA1)

93. Регистры модуля DMA1 в STM32 управляют прямым доступом к памяти, позволяя периферии обмениваться данными с памятью без участия ядра. Основные регистры включают ISR (статус), IFCR (очистка флагов), CCRx (конфигурация канала), CNDTRx (счетчик данных), CPARx (адрес периферии) и CMARx (адрес памяти) для каждого из каналов (обычно 7 каналов).

2.3.15 Подключение SD-карты к микроконтроллеру

94. Подключение SD-карты к микроконтроллеру (MCU) чаще всего осуществляется через интерфейс SPI (4 провода + питание), так как он поддерживается большинством МК (AVR, STM32, ESP32). SD-карта работает от 3,3V, поэтому при

подключении к 5V-микроконтроллерам (например, Arduino Uno) необходим преобразователь уровней для линий данных.

2.3.16 Файловая система FAT на внешней SD-карте

95. Использование файловой системы FAT на SD-карте в STM32 обычно реализуется через библиотеку FatFs, разработанную Elm Chan. Она обеспечивает удобный API для чтения/записи файлов (FAT12/16/32, exFAT), изолируя прикладной код от низкоуровневых драйверов SD-карты (SPI или SDIO).

2.3.17 Настройка FatFs в STM32CubeM

96. Подключение карты памяти (SD/microSD) к STM32 осуществляется через интерфейс SPI (проще) или SDIO (быстрее). Схема включает питание 3.3В, линии SPI (SCK, MISO, MOSI, CS) и резисторы подтяжки (pull-up) 10-100 кОм на линиях данных/CS. Важно обеспечить стабильное питание и качественное соединение, особенно для SDIO.

2.3.18 Режимы пониженного энергопотребления. SLEEP

97. STM32 предлагает несколько режимов энергосбережения для оптимизации питания: Sleep (стоп ядра, периферия работает), Stop (остановка всех тактовых частот, сохранение ОЗУ) и Standby/Shutdown (минимальное потребление, отключение питания, сброс). Также существуют промежуточные режимы, например, Low-power Run/Sleep.

98. Режим SLEEP в STM32 — это режим пониженного энергопотребления, при котором останавливается тактирование ядра Cortex-M, но периферия и память (RAM/Flash) остаются активными. Он обеспечивает быстрый выход из спячки (несколько мкс), сохраняя работоспособность таймеров, АЦП и коммуникационных интерфейсов, потребляя меньше энергии, чем в активном режиме.

2.3.19 АЦП STM32. Общие сведения, режимы. Установка конфигурации через регистры CMSIS

99. Аналого-цифровой преобразователь (АЦП) превращает непрерывный аналоговый сигнал (например, напряжение с датчика) в последовательность цифровых чисел. Процесс включает дискретизацию по времени (измерение через интервалы) и квантование по уровню (присвоение числового значения), что позволяет микропроцессорам обрабатывать сигналы.

100. АЦП (ADC) в микроконтроллерах STM32 — это высокопроизводительные 12-битные (в ряде моделей 14/16-битные) преобразователи последовательного приближения (SAR), отличающиеся наличием регулярных и инжектированных каналов, поддержкой DMA, встроенной калибровкой и высокой скоростью опроса (до нескольких миллионов выборок в секунду, MSPS).

101. АЦП в STM32 использует два типа каналов: регулярные (для фонового, последовательного опроса множества датчиков) и инжектированные (для мгновенного прерывания текущих измерений и опроса критически важных сигналов). Регулярные каналы обычно работают с DMA, тогда как инжектированные каналы используются для задач реального времени, "впрыскиваясь" в очередь преобразований.

102. Конфигурация АЦП через CMSIS (на примере STM32) включает включение тактирования RCC, настройку GPIO на аналоговый вход, сброс/калибровку АЦП и установку регистров управления (CR1, CR2, SMPR, SQR). Основные шаги: включение питания, выбор канала, времени выборки (SMPR), режима работы (одиночный/непрерывный) и запуск через ADC_CR2_ADON.

2.3.20 Работа с АЦП через регистры CMSIS. Основные режимы преобразования

103. Для преобразования аналогового сигнала в цифровую форму на STM32 (АЦП) необходимо: настроить тактирование периферии, сконфигурировать GPIO-пин как аналоговый вход, настроить параметры АЦП (разрешение, время выборки), выбрать режим работы (одиночный, сканирование, DMA) и запустить процесс преобразования.

104. АЦП в микроконтроллерах STM32 имеет гибкую настройку, обеспечивая режимы однократного (single) и непрерывного (continuous) преобразования, работу с регулярными и инжектированными каналами, а также расширенные режимы сканирования, совместной работы нескольких АЦП (сдвоенные/строенные) и использования DMA для передачи данных.

105. Режим одноканального однократного преобразования АЦП в STM32 — это работа преобразователя, при которой по программному или аппаратному триггеру выполняется одно измерение аналогового сигнала на одном конкретном канале, после чего АЦП автоматически останавливается. Результат записывается в регистр данных (DR), и для следующего измерения требуется новый запуск.

106. Режим непрерывного преобразования одного канала (Single Channel Continuous Conversion) в STM32 позволяет АЦП автоматически перезапускать измерение сразу после завершения предыдущего, обеспечивая постоянный поток данных с одного входа. АЦП работает без участия процессора, обновляя регистр данных DR.

107. Режим сканирования нескольких каналов (Scan Mode) с однократным преобразованием (Single Conversion) в STM32 позволяет по одному запросу последовательно оцифровать набор аналоговых входов. АЦП проходит по списку каналов, преобразует каждый, и останавливается, обычно используя DMA для передачи данных.

108. Режим сканирования нескольких каналов (Scan Mode) с непрерывным преобразованием (Continuous Conversion) в STM32 позволяет АЦП автоматически последовательно опрашивать группу каналов (например, IN0, IN1, IN2) и после завершения последнего сразу начинать заново с первого. В связке с DMA это обеспечивает постоянное обновление значений в памяти без участия процессора.

2.3.21 Работа с АЦП в различных режимах. Запуск от таймера, чтение результата с использованием прерываний.

109. Внешний запуск инжектированной группы АЦП в STM32 осуществляется через настройку триггера (EXTSEL) в регистре ADC_JSQR, который реагирует на события таймеров или GPIO. Настройка включает выбор источника триггера, полярности (EXTEN), настройку каналов и активацию через HAL_ADCEx_InjectedStart_IT() или запись в регистр.

110. Автоматический запуск инжектированных каналов АЦП в STM32 (injected channels) — это режим, при котором измерения происходят без участия ядра, обычно по сигналу таймера. В отличие от регулярных каналов, инжектированные «вклиниваются» в процесс преобразования (приостанавливая регулярные). Настройка включает привязку триггера (EXTSEL/JEXTSEL) и включение сканирования (Scan mode).

2.3.22 Режим сдвоенных преобразований АЦП STM32

111. Режим сдвоенных преобразований (Dual Mode) в АЦП STM32 — это режим работы, при котором два модуля АЦП (обычно ADC1 и ADC2) синхронизируются для одновременного или последовательного измерения сигналов. Это позволяет удвоить частоту дискретизации, сканировать больше каналов или одновременно измерять сигналы в паре, объединяя 12-битные данные в один 32-битный регистр, что удобно для DMA-передачи.

112. Режим одновременных преобразований инжектированных каналов (Injected Simultaneous Mode) в АЦП STM32 — это режим, при котором ADC1 (master) и ADC2 (slave) одновременно запускают и выполняют преобразование заранее заданных инжектированных групп каналов. Это позволяет синхронно оцифровать до 4 сигналов (по 4 в каждом АЦП) с высокой точностью в один момент времени, что используется для многоканальных измерений.

113. Режим одновременных преобразований регулярных каналов (Dual ADC Regular Simultaneous Mode) в STM32 — это режим сдвоенной работы двух АЦП (ADC1 и ADC2), при котором регулярные каналы обоих модулей преобразуются одновременно по одному триггеру. Это удваивает частоту дискретизации и позволяет синхронно оцифровывать два сигнала, например, для измерения тока и напряжения.

114. Режим «быстрых преобразований со смещением во времени» (Fast Interleaved Mode) в АЦП STM32 — это режим работы сдвоенных АЦП (dual ADC), при котором они синхронно преобразуют один и тот же канал с задержкой друг относительно друга, удваивая частоту дискретизации. АЦП1 запускается с небольшой задержкой после АЦП2, что позволяет одному каналу достичь максимальной скорости выборки, превышающей возможности одного АЦП.

115. Режим медленных преобразований со смещением во времени (часто понимаемый как режим сдвоенных/совмещенных АЦП — Dual Mode/Interleaved) в STM32 — это режим работы, при котором преобразования регулярных каналов приостанавливаются для запуска инжектированных (приоритетных) каналов. Это обеспечивает чередование: регулярные каналы опрашиваются, затем выполнение переключается на инжектированные, и затем снова возвращается к регулярным.

116. Режим поочередного запуска АЦП (Alternate Trigger Mode) — это режим работы сдвоенных АЦП (например, в STM32), при котором один триггер (сигнал запуска) попеременно запускает преобразование инжектированных каналов на разных АЦП. При первом запуске работает АЦП1, при втором — АЦП2, и так далее, что используется для высокоскоростного сбора данных.

117. Комбинированный режим одновременного преобразования (Combined Regular/Injected Simultaneous Mode) в АЦП STM32 — это режим работы двух АЦП (ADC1 и ADC2) в режиме сдвоенного преобразования (dual mode), который синхронно запускает инжектированные (Injected) и регулярные (Regular) группы каналов. Он обеспечивает максимально близкое по времени измерение до 8 каналов (4 регулярных + 4 инжектированных) за счет синхронизации ADC1 и ADC2.

118. Комбинированный режим в АЦП STM32 (обычно Dual ADC mode) сочетает одновременное преобразование нескольких каналов двумя АЦП для увеличения скорости и поочередный (прерывистый/discontinuous) запуск для управления выборкой. Это позволяет синхронно измерять пары сигналов, разбивая длинную последовательность на короткие этапы.

119. Комбинированный режим в АЦП STM32 (Dual/Triple mode) объединяет преимущества одновременного сэмплирования инжектированных каналов (для высокого приоритета) и сдвига времени (Interleaved/Delay) для регулярных каналов. Это позволяет синхронно измерять токи/напряжения и одновременно повышать общую частоту дискретизации, эффективно используя несколько АЦП.

2.3.23 Работа АЦП в режиме оконного компаратора

120. Режим оконного компаратора в АЦП STM32 — это функция аппаратного мониторинга, которая проверяет, попадает ли преобразованное напряжение в заданный диапазон (окно) между высоким (HTR) и низким (LTR) порогами. Если сигнал выходит за границы окна, АЦП автоматически генерирует прерывание, позволяя мгновенно реагировать без участия CPU.

121. Проверка режима оконного компаратора АЦП STM32 осуществляется путем установки пороговых значений (LTR, HTR) и контроля флага AWD (Analog Watchdog) в регистре статуса ADC_SR, который срабатывает, когда значение канала выходит за заданные пределы. В оконном режиме AWD активируется, если сигнал вне диапазона [LTR; HTR].

122. Внутренний источник опорного напряжения (VREFINT) в АЦП STM32 — это встроенный в микроконтроллер прецизионный источник стабильного напряжения (обычно около 1.2В – 1.25В), не зависящий от напряжения питания. Он используется для калибровки АЦП, точного измерения напряжения питания (V_{DDA}) и преобразования цифровых кодов в абсолютные величины напряжения.

123. АЦП в микроконтроллерах STM32 — это преимущественно 12-битные последовательного приближения (SAR), отличающиеся высокой скоростью (до нескольких Мвыб/с), наличием самокалибровки для снижения смещения нуля, встроенным источником опорного напряжения (ИОН) и возможностью работы с DMA, обеспечивая точность порядка нескольких МЗР.

2.3.24 Работа с АЦП через функции библиотеки HAL

124. Работа с АЦП (ADC) в STM32 через HAL-библиотеку строится на инициализации структуры ADC_HandleTypeDef, запуске преобразования (HAL_ADC_Start) и чтении значения (HAL_ADC_GetValue). Основные режимы включают опросы (Polling), прерывания (IT) и DMA, обеспечивая преобразование аналоговых сигналов (0–3.3В) в 12-битные цифровые значения (0–4095).

2.3.25 USB HID (human interface device) – класс устройств для взаимодействия с человеком

125 USB HID (Human Interface Device) — это класс USB-устройств, предназначенных для прямого взаимодействия человека с компьютером, обеспечивающий обмен данными ввода-вывода (мышь, клавиатура, геймпад) без необходимости установки специфических драйверов. Это стандартизированный протокол, позволяющий системе «plug-and-play» мгновенно распознавать устройства, использующие стандартные дескрипторы для передачи команд, таких как нажатие клавиш или движение курсора. Приведите пример.

126. Choice Group— это элемент интерфейса (UI-компонент), предназначенный для выбора одного варианта из группы предложенных. Он позволяет пользователю выбрать только один элемент (радиокнопки), что делает его идеальным для взаимоисключающих опций.

127. Custom Item (обычно Custom HID) — это пользовательское USB-устройство, созданное на базе микроконтроллера STM32, которое не соответствует стандартным классам (как мышь или клавиатура). Оно используется для обмена произ-

вольными данными между ПК и контроллером, позволяя задавать уникальные параметры передачи.

Тема 3 Автоматизация процессов на основе систем с МК

3.1 Структура управляющих систем. Понятие обратной связи и принципы управления. Принципы автоматического управления

128. Две основные схемы управляющих систем — это разомкнутая и замкнутая (с обратной связью). Разомкнутая система работает по заранее заданной программе без учета результатов, а замкнутая использует датчики для постоянной корректировки действий на основе фактического результата. Они различаются принципом использования обратной связи.

129. Обратная связь (ОС) в управляющих системах — это передача информации о фактическом состоянии выхода системы обратно на вход для корректировки управляющего воздействия. Она обеспечивает стабилизацию параметров, повышение точности, подавление возмущений и адаптацию к изменениям. Ключевые свойства: формирование замкнутого контура, компенсация ошибок и возможность достижения цели.

130. Автоматическое управление — это поддержание заданных параметров работы оборудования без участия человека. Основные принципы включают управление по отклонению (обратная связь), компенсацию возмущений и комбинированное регулирование. Главная цель — обнаружить отклонение регулируемой величины от нормы и устранить его.

3.2 Основы создания алгоритмов и программ для взаимодействия систем на основе МК с пользователем

131. Проектирование UI/UX — это последовательный процесс создания удобного (UX) и привлекательного (UI) интерфейса, включающий исследования, создание структуры, прототипирование и тестирование. Этапы включают: анализ аудитории, разработку сценариев (user flow), создание черно-белых каркасов (wireframes), дизайн интерфейса и юзабилити-тестирование, обеспечивая интуитивно понятное взаимодействие пользователя с системой.

132. Разработка алгоритмов — это создание четкой последовательности действий для решения поставленной задачи, базирующаяся на трех структурах: следование (линейный порядок), ветвление (условия if-then-else) и циклы (while, for). Процесс включает анализ, проектирование (блок-схемы/псевдокод), реализацию и тестирование на эффективность (время/память).

133. Программная реализация алгоритмов в микроконтроллерах (МК) включает написание кода (обычно на C/C++), его компиляцию в машинные коды, прошивку в память МК с помощью программатора и отладку на реальном «железе». Процесс

базируется на модульности (разбиении на блоки), использовании прерываний для реального времени и прямом управлении регистрами периферии.

134. Пользовательский интерфейс (UI — User Interface) — это внешняя среда, инструменты и способы, с помощью которых человек взаимодействует с цифровым продуктом (сайтом, приложением) или устройством. Он включает визуальные элементы — кнопки, меню, шрифты, иконки и формы ввода, — обеспечивая интуитивно понятное и удобное управление.

135. Формы представления пользовательского интерфейса (UI) определяют, как человек взаимодействует с цифровой системой. Основные формы включают графический (GUI), текстовый (CLI/TUI), голосовой (VUI), жестовый и смешанный (VR/AR) интерфейсы, выбор которых зависит от устройства, цели и целевой аудитории для обеспечения удобства и эффективности.

136. Пользовательский интерфейс (UI) — это способ взаимодействия человека с цифровым устройством, включающий визуальные элементы, звуки и реакции. Основные виды включают графические (GUI), командные (CLI), голосовые (VUI), жестовые (Gesture UI) и текстовые (TUI) интерфейсы, которые подбираются под задачи устройства и удобство пользователя.

137. Компоненты пользовательского интерфейса (UI) — это интерактивные и визуальные элементы, через которые человек взаимодействует с цифровым продуктом. Они делятся на элементы ввода (кнопки, поля), навигации (меню, хлебные крошки), информирования (иконки, прогресс-бары) и контейнеры (окна, панели), обеспечивая интуитивное управление.

138. Разработка пользовательского интерфейса (UI) — это итерационный процесс, включающий исследование аудитории, создание структуры (UX), визуальный дизайн, прототипирование, тестирование и передачу в разработку. Основные этапы — анализ требований, проектирование макетов, создание дизайн-концепции, тестирование и подготовка материалов для верстки.

139. Хороший интерфейс (UI/UX) — это невидимый помощник, который делает взаимодействие с продуктом интуитивным, быстрым и приятным, следуя принципу «не заставляйте меня думать». Ключевые правила включают: простоту (минимализм), предсказуемость, единый стиль (последовательность), мгновенную обратную связь, адаптивность под устройства и высокую визуальную четкость.

3.3 Особенности работы с внешними устройствами

140. На низком уровне микроконтроллер (МК) общается с миром через физические электрические сигналы, которые преобразуются в логические нули и единицы.

- Определение: прямое считывание сигналов с физических датчиков.
- Примеры:
 - Терморезистор для измерения температуры.

- Фоторезистор для определения освещенности.
- Кнопки для получения информации о нажатии.
- Датчик Холла для определения магнитного поля.
- Как работает: микроконтроллер читает аналоговые или цифровые сигналы напрямую с датчиков через свои порты ввода-вывода (I/O) и использует их для принятия решений или управления другими устройствами.

141. Высокоуровневое взаимодействие

- Определение: использование сенсоров, которые сами выполняют предварительную обработку данных перед передачей микроконтроллеру.
- Примеры:
 - Камеры (например, OV7670) с встроенным блоком обработки изображений, которые передают на микроконтроллер уже обработанные данные, например, информацию о распознанном объекте.
 - Модули Wi-Fi/Bluetooth (например, ESP8266/ESP32) для беспроводной связи с другими устройствами, которые принимают и отправляют данные.
 - Датчики расстояния (например, радары), которые обрабатывают данные для определения точного расстояния до объектов.

Как работает: микроконтроллер получает уже обработанные и более сложные данные, что позволяет ему сосредоточиться на основной задаче, а не на низкоуровневой обработке сигналов.

142. Преимущества и недостатки

- Низкоуровневые:
 - Преимущества: дешевле, меньше, проще в использовании.
 - Недостатки: требуют более сложного программирования для обработки всех данных.
- Высокоуровневые:
 - Преимущества: освобождают микроконтроллер от рутинной обработки данных, упрощают разработку.
 - Недостатки: дороже, более сложные в настройке, могут быть менее гибкими.

143. Подключение LCD-дисплея на контроллере HD44780 (например, 1602) к STM32F103C8T6 (Blue Pill) чаще всего выполняется в 4-битном режиме для экономии пинов. Используйте 6 линий GPIO (RS, E, D4-D7), потенциометр 10 кОм для контраста, 220 Ом для подсветки и запитайте дисплей от 5В.

144. Подключение разъема USB 2.0 (обычно Type-A или Micro-B) к микроконтроллеру требует соединения четырех основных линий: питания (+5В, GND) и дифференциальных данных (D+, D-). Линии D+/D- подключаются напрямую к USB-контроллеру микроконтроллера (например, в STM32), часто через защитные резисторы 22 – 33 Ом.

145. Сигналы инкрементального энкодера (каналы А и В) обрабатываются микроконтроллером путем фиксации фронтов импульсов. Направление вращения определяется по состоянию сигнала В в момент изменения сигнала А. Основные методы включают использование прерываний для быстрого реагирования, простой

опрос пинов (polling) или аппаратные счетчики (таймеры) для точного подсчета на высокой скорости.

3.4 Взаимодействие систем на основе МК по телекоммуникационным сетям с другими вычислительными системами

146. Взаимодействие микроконтроллеров (МК) с вычислительными системами (ПК, серверы, другие МК) осуществляется через проводные и беспроводные интерфейсы, обеспечивающие обмен данными. Основные способы включают последовательные протоколы (UART/RS-232, SPI, I2C, CAN), USB, Ethernet и беспроводные технологии (Wi-Fi, Bluetooth, ZigBee), позволяя МК передавать телеметрию, получать команды и работать в сети.

147. Основные компоненты вычислительной системы (включая микроконтроллеры — МК) при взаимодействии с другими системами делятся на аппаратные и функциональные блоки: центральный процессор (CPU), ОЗУ (RAM) и ПЗУ (ROM), система шин, периферийные устройства ввода/вывода (USB, UART, GPIO) и интерфейсы связи (Ethernet, CAN, Wi-Fi).

148. Подключение микроконтроллера к локальной сети (LAN/Ethernet) организуется через внешний сетевой контроллер (например, ENC28J60, W5500) по интерфейсу SPI или встроенный Ethernet-адаптер. Модуль преобразует данные МК в пакеты Ethernet, работая с TCP/IP стеком.

Пример: Подключение Arduino к сети через модуль ENC28J60.

149. Модель OSI (Open Systems Interconnection) — это 7-уровневая эталонная модель взаимодействия сетевых протоколов, где каждый уровень (от физического до прикладного) отвечает за определенный этап передачи данных. Она описывает, как информация превращается в сигналы и передается между системами, обеспечивая совместимость различного оборудования.

Уровни модели OSI (снизу-вверх):

7. Прикладной (Application Layer): Уровень пользовательских приложений (браузеры, почтовые клиенты). Взаимодействует с сетью, определяя тип приложения (HTTP, SMTP, FTP, DNS).

6. Представительный (Presentation Layer): Обеспечивает преобразование форматов данных (кодирование, сжатие, шифрование) в понятный для прикладного уровня вид (например, JPEG, ASCII).

5. Сеансовый (Session Layer): Управляет сессиями связи, устанавливая, поддерживая и завершая диалог между двумя приложениями.

4. Транспортный (Transport Layer): Отвечает за доставку данных между хостами, разделяет данные на части, контролирует целостность и порядок следования (протоколы TCP, UDP).

3. Сетевой (Network Layer): Определяет маршрутизацию данных, IP-адресацию и пересылку пакетов между сетями (маршрутизаторы).

2. Канальный (Data Link Layer): Обеспечивает передачу данных через физическую среду, проверяет ошибки и управляет доступом к среде (MAC-адреса, коммутаторы).

1. Физический (Physical Layer): Работает с физической средой передачи данных — биты превращаются в электрические сигналы, свет или радиоволны (кабели, сетевые адаптеры).

Принцип работы: При отправке данные инкапсулируются (проходят от 7 до 1 уровня), превращаясь в биты. При получении происходит декапсуляция (от 1 к 7 уровню), трансформируя биты обратно в понятную информацию.

150. Структура кадра Ethernet (стандарт II) — это последовательность байт, используемая для передачи данных в локальных сетях, состоящая из преамбулы (8 байт), MAC-адресов получателя и отправителя (по 6 байт), типа протокола (2 байта), полезных данных (46-1500 байт) и контрольной суммы (4 байта). Она обеспечивает синхронизацию, адресацию и проверку целостности данных.

151. Протокол ARP (Address Resolution Protocol) — это сетевой протокол, используемый для сопоставления IP-адреса (уровень 3) с физическим MAC-адресом (уровень 2) в локальных сетях (Ethernet). Он работает по принципу «запрос-ответ», рассылая широковещательные сообщения для поиска нужного устройства, чтобы отправить ему данные, и кэширует результаты для ускорения работы.

152. ARP-запрос и ARP-ответ — это фундаментальный механизм протокола ARP (Address Resolution Protocol), работающий в локальных сетях (Ethernet) для сопоставления IP-адреса (логического) с физическим MAC-адресом.

153. ICMP (Internet Control Message Protocol) — это сетевой протокол (уровень 3 модели OSI), предназначенный для обмена диагностическими сообщениями и уведомления об ошибках при передаче IP-пакетов. Он не используется для передачи данных пользователями, а работает "внутри" IP-пакетов для информирования о доступности узлов, маршрутах и проблемах сети.

154. Протокол IP (Internet Protocol) — это основной сетевой протокол, отвечающий за адресацию и маршрутизацию пакетов данных (дейтаграмм) между устройствами в составных сетях. Он обеспечивает доставку данных от источника к получателю, прокладывая маршруты через промежуточные узлы, но не гарантирует надежность доставки или целостность данных, перекладывая эти функции на протоколы верхнего уровня (например, TCP).

155. Заголовок IPv4 (обычно 20–60 байт) — это обязательный префикс IP-дейтаграммы, обеспечивающий маршрутизацию. Ключевые поля включают Версию (4), Длину заголовка (IHL), Общую длину, Идентификатор (для фрагментации), TTL (время жизни), Протокол (TCP/UDP), контрольную сумму и 32-битные адреса отправителя/получателя.

156. Заголовок ICMP (Internet Control Message Protocol) — это служебная часть пакета, используемая для диагностики и передачи сообщений об ошибках, которая

инкапсулируется непосредственно в IP-пакет. Заголовок имеет фиксированный 8-байтовый (64 бита) размер для первых трех полей, за которыми следует специфичная для типа сообщения информация.

3.5 Операционные системы реального времени

157. С точки зрения критериев использования и эффективности, многозадачные операционные системы (ОС) делятся на три основных типа: системы пакетной обработки (для пакетного решения задач), системы разделения времени (для интерактивной работы пользователей) и системы реального времени (для жесткого контроля за процессами).

158. Операционные системы реального времени (ОСРВ/RTOS) гарантируют выполнение задач за строго определенное время, что критично для встраиваемых систем, авиации и промышленных роботов. Основные примеры: FreeRTOS, QNX Neutrino, VxWorks, Zephyr, RTLinux и российская ЗОСРВ «Нейтрино».

159. Граф состояния задач в ОСРВ (Операционных Системах Реального Времени) описывает жизненный цикл задачи, обеспечивая предсказуемость исполнения. Основные состояния: «Запуск» (Ready), «Выполнение» (Running), «Ожидание» (Blocked/Waiting). Переходы между ними осуществляются планировщиком на основе приоритетов и событий, гарантируя реакцию на внешние запросы.