

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Иркутский государственный университет путей сообщения»
Сибирский колледж транспорта и строительства

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по профессиональному модулю

ПМ. 02 Проектирование управляющих программ компьютерных систем и комплексов
МДК. 02.02. Программирование микроконтроллеров
для специальности
09.02.01 Компьютерные системы и комплексы
базовая подготовка среднего профессионального образования

Иркутск 2023

Электронный документ выгружен из ЕИС ФГБОУ ВО ИргУПС и соответствует оригиналу

Подписант ФГБОУ ВО ИргУПС Трофимов Ю.А.

00a73c5b7b623a969ccad43a81ab346d50 с 08.12.2022 14:32 по 02.03.2024 14:32 GMT+03:00

Подпись соответствует файлу документа



Фонд оценочных средств разработан в соответствии с ФГОС СПО по специальности 09.02.01 Компьютерные системы и комплексы, утвержденного приказом Министерства образования и науки Российской Федерации от 25 мая 2022 г. N 362, на основе рабочей программы дисциплины ПМ. 02 Проектирование управляющих программ компьютерных систем и комплексов МДК. 02.02. Программирование микроконтроллеров

РАССМОТРЕНО:

Цикловой методической
комиссией специальности
09.02.01 Компьютерные
системы и комплексы
Протокол № 9
«26» мая 2023 г.
Председатель ЦМК: Арефьева Н.В.

СОГЛАСОВАНО

Заместитель директора по УВР
Ресельс А.П.
«01» июня 2023г.

Согласовано:

Ведущий специалист «Института
информационных технологий и
кибербезопасности »

 С.В. Бахвалов

Разработчики: Фитисова Н.Н. преподаватели высшей категории Сибирского колледжа транспорта и строительства ФГБОУ ВО «Иркутский государственный университет путей сообщения».

СОДЕРЖАНИЕ

| | |
|--|----|
| 1.Формы контроля и оценивания элементов профессионального модуля | 4 |
| 2. Результаты освоения модуля..... | 5 |
| 3 Типовые задания для оценки освоения МДК.02.04 Администрирование сетей..... | 20 |

1.Формы контроля и оценивания элементов профессионального модуля

| Элемент модуля | Форма контроля и оценивания | |
|--|-----------------------------|--|
| | Промежуточная аттестация | Текущий контроль |
| МДК. 02.02. Программирование микроконтроллеров | Экзамен | оценка защиты практических; тестирование, опрос. |

2. Результаты освоения модуля

2.1. В результате аттестации по профессиональному модулю осуществляется комплексная проверка следующих знаний, умений, профессиональных и общих компетенций:

Контроль и оценка результатов освоения модуля осуществляется преподавателем в процессе проведения практических занятий, тестирования, учебной практики, экзамена квалификационного, а также при выполнении обучающимися индивидуальных заданий решения ситуационных задач, тестирования и различных видов опроса.

| | |
|--------------------------------|---|
| Иметь практический опыт | <ul style="list-style-type: none">составления формализованных описаний решений поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов;разработки алгоритмов решения поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов;оценки и согласования сроков выполнения поставленных задач;создания программного кода в соответствии с техническим заданием (готовыми спецификациями);оптимизации программного кода с использованием специализированных программных средств;приведения наименований переменных, функций, классов, структур данных и файлов в соответствие с установленными в организации требованиями;структурирования и форматирования исходного программного кода в соответствии с установленными в организации требованиями;комментирования и разметки программного кода в соответствии с установленными в организации требованиями;анализа и проверки исходного программного кода;отладки программного кода на уровне программных модулей;подготовки тестовых наборов данных в соответствии с выбранной методикой;регистрации изменений исходного текста программного кода в системе контроля версий;слияния, разделения и сравнения исходных текстов программного кода;сохранения сделанных изменений программного кода в соответ- |
|--------------------------------|---|

| | |
|--------------|--|
| | <p>ствии с регламентом контроля версий;</p> <p>выполнения процедур сборки программных модулей и компонент в программный продукт;</p> <p>подключения программного продукта к компонентам внешней среды;</p> <p>проверки работоспособности выпусков программного продукта;</p> <p>внесения изменений в процедуры сборки модулей и компонент программного обеспечения, развертывания программного обеспечения, миграции и преобразования данных;</p> <p>разработки и документирования программных интерфейсов;</p> <p>разработки процедур сборки модулей и компонент программного обеспечения;</p> <p>разработки процедур развертывания и обновления программного обеспечения;</p> <p>разработки процедур миграции и преобразования (конвертации) данных;</p> <p>подготовки тестовых сценариев и тестовых наборов данных в соответствии с выбранной методикой;</p> <p>тестирования и верификации управляющих программ;</p> <p>оформления отчетов о тестировании;</p> <p>запуска процедуры установки прикладного программного обеспечения на конечных устройствах пользователей и/или серверном оборудовании;</p> <p>контроля процедуры установки прикладного программного обеспечения;</p> <p>настройки установленного прикладного программного обеспечения;</p> <p>обновления установленного прикладного программного обеспечения.</p> |
| Уметь | <p>использовать методы и приемы формализации задач;</p> <p>использовать методы и приемы алгоритмизации поставленных задач;</p> <p>использовать программные продукты для графического отображения алгоритмов;</p> <p>применять стандартные алгоритмы в соответствующих областях;</p> <p>применять выбранные языки программирования для написания программного кода;</p> <p>использовать выбранную среду программирования и средства си-</p> |

| | |
|--|--|
| | <p>системы управления базами данных;</p> <p>использовать возможности имеющейся технической и/или программной архитектуры;</p> <p>применять нормативные документы, определяющие требования к оформлению программного кода;</p> <p>применять инструментарий для создания и актуализации исходных текстов программ.</p> <p>выявлять ошибки в программном коде;</p> <p>применять методы и приемы отладки программного кода;</p> <p>интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов;</p> <p>применять современные компиляторы, отладчики и оптимизаторы программного кода;</p> <p>документировать произведенные действия, выявленные проблемы и способы их устранения;</p> <p>проводить оценку работоспособности программного продукта;</p> <p>создавать резервные копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта и данных;</p> <p>использовать выбранную систему контроля версий;</p> <p>выполнять действия, соответствующие установленному регламенту используемой системы контроля версий;</p> <p>интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов;</p> <p>применять современные компиляторы, отладчики и оптимизаторы программного кода;</p> <p>документировать произведенные действия, выявленные проблемы и способы их устранения;</p> <p>создавать резервные копии программ и данных, выполнять восстановление, обеспечивать целостность программного продукта и данных;</p> <p>выполнять процедуры сборки программных модулей и компонент в программный продукт;</p> <p>производить настройки параметров программного продукта и осуществлять запуск процедур сборки;</p> <p>писать программный код процедур интеграции программных модулей;</p> <p>использовать выбранную среду программирования для разработ-</p> |
|--|--|

| | |
|--------------|--|
| | <p>ки процедур интеграции программных модулей;</p> <p>применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов;</p> <p>разрабатывать и оформлять контрольные примеры для проверки работоспособности программного обеспечения;</p> <p>разрабатывать процедуры генерации тестовых наборов данных с заданными характеристиками;</p> <p>подготавливать наборы данных, используемых в процессе проверки работоспособности программного обеспечения;</p> <p>выявлять соответствие требований заказчиков к существующим продуктам;</p> <p>соблюдать процедуру установки прикладного программного обеспечения в соответствии с требованиями организации-производителя;</p> <p>идентифицировать инциденты, возникающие при установке программного обеспечения, и принимать решение по изменению процедуры установки.</p> |
| Знать | <p>методы и приемы формализации и алгоритмизации задач;</p> <p>языки формализации функциональных спецификаций;</p> <p>нотации и программные продукты для графического отображения алгоритмов;</p> <p>алгоритмы решения типовых задач, области и способы их применения;</p> <p>синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования;</p> <p>методологии разработки программного обеспечения;</p> <p>методологии и технологии проектирования и использования баз данных;</p> <p>технологии программирования;</p> <p>особенности выбранной среды программирования и системы управления базами данных;</p> <p>компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними;</p> <p>инструментарий для создания и актуализации исходных текстов программ;</p> |

| | |
|--|--|
| | <p>методы повышения читаемости программного кода;</p> <p>системы кодировки символов, форматы хранения исходных текстов программ;</p> <p>нормативные документы, определяющие требования к оформлению программного кода;</p> <p>методы и приемы отладки программного кода;</p> <p>типы и форматы сообщений об ошибках, предупреждений;</p> <p>способы использования технологических журналов, форматы и типы записей журналов;</p> <p>современные компиляторы, отладчики и оптимизаторы программного кода;</p> <p>сообщения о состоянии аппаратных средств;</p> <p>методы и средства верификации работоспособности выпусков программных продуктов;</p> <p>языки, утилиты и среды программирования, средства пакетного выполнения процедур;</p> <p>возможности используемой системы контроля версий и вспомогательных инструментальных программных средств;</p> <p>установленный регламент использования системы контроля версий;</p> <p>методы и средства сборки и интеграции программных модулей и компонент;</p> <p>интерфейсы взаимодействия с внешней средой;</p> <p>интерфейсы взаимодействия внутренних модулей системы;</p> <p>методы и средства сборки модулей и компонент программного обеспечения;</p> <p>интерфейсы взаимодействия с внешней средой;</p> <p>интерфейсы взаимодействия внутренних модулей системы;</p> <p>методы и средства разработки процедур для развертывания программного обеспечения;</p> <p>методы и средства миграции и преобразования данных;</p> <p>методы создания и документирования контрольных примеров и тестовых наборов данных;</p> <p>правила, алгоритмы и технологии создания тестовых наборов данных;</p> <p>требования к структуре и форматам хранения тестовых наборов данных;</p> <p>основные понятия в области качества программных продуктов;</p> |
|--|--|

| | |
|--|--|
| | <p>лицензионные требования по настройке устанавливаемого программного обеспечения;</p> <p> типовые причины инцидентов, возникающих при установке программного обеспечения;</p> <p>основы архитектуры, устройства и функционирования вычислительных систем;</p> <p>принципы организации, состав и схемы работы операционных систем;</p> <p>стандарты информационного взаимодействия систем.</p> |
|--|--|

Результатом освоения программы междисциплинарного курса профессионального модуля является овладение обучающимися видом профессиональной деятельности Применение микропроцессорных систем, установка и настройка периферийного оборудования., в том числе овладение профессиональными (ПК) и общими (ОК) компетенциями:

| Код | Наименование результата обучения |
|--------|---|
| ПК 2.1 | Проектировать, разрабатывать и отлаживать программный код модулей управляющих программ. |
| ПК 2.2 | Владеть методами командной разработки программных продуктов. |
| ПК 2.3 | Выполнять интеграцию модулей в управляющую программу. |
| ПК 2.4 | Тестировать и верифицировать выпуски управляющих программ. |
| ПК 2.5 | Выполнять установку и обновление версий управляющих программ |

| | |
|------|---|
| ОК 1 | Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам; |
| ОК 2 | Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности; |
| ОК 3 | Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях; |
| ОК 4 | Эффективно взаимодействовать и работать в коллективе и команде; |
| ОК 5 | ОК 5 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста; |

| | |
|------|--|
| ОК 6 | Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения; |
| ОК 7 | ОК 7 Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях; |
| ОК 8 | Пользоваться профессиональной документацией на государственном и иностранном языках. |
| ОК 9 | Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам; |

В результате аттестации по междисциплинарному курсу профессионального модуля осуществляется комплексная проверка следующих профессиональных компетенций:

| Профессиональные компетенции | Основные показатели оценки результата | Формы и методы контроля и оценки |
|---|---|---|
| ПК 2.1. Проектировать, разрабатывать и отлаживать программный код модулей управляющих программ. | Проектировать, разрабатывать и отлаживать программный код модулей управляющих программ. | Текущий контроль в форме: - защиты практических занятий. Зачеты по учебной практике и по каждому из разделов профессионального модуля. Экзамен по дисциплине. Оценка действий обучающихся в процессе практических занятий. |
| ПК 2.2. Владеть методами командной разработки программных продуктов. | Владеть методами командной разработки программных продуктов. | Текущий контроль в форме: - защиты практических занятий. Зачеты по учебной практике и по каждому из разделов профессионального модуля. Экзамен по дисциплине. Оценка действий обучающихся |

| | | |
|--|---|--|
| | | в процессе практических занятий |
| ПК 2.3. Выполнять интеграцию модулей в управляющую программу. | Выполнять интеграцию модулей в управляющую программу. | Текущий контроль в форме: - защиты практических занятий. Зачеты по учебной практике и по каждому из разделов профессионального модуля. Комплексный экзамен по профессиональному модулю. Оценка действий обучающихся в процессе практических занятий |
| ПК 2.4. Тестировать и верифицировать выпуски управляющих программ. | Тестировать и верифицировать выпуски управляющих программ. | Текущий контроль в форме: - защиты практических занятий. Зачеты по учебной практике и по каждому из разделов профессионального модуля. Комплексный экзамен по профессиональному модулю. Оценка действий обучающихся в процессе практических занятий |
| ПК 2.5. Выполнять установку и обновление версий управляющих программ | .Выполнять установку и обновление версий управляющих программ | Текущий контроль в форме: - защиты практических занятий. Зачеты по учебной практике и по каждому из разделов профессионального модуля. Комплексный экзамен по профессиональному модулю. Оценка действий обучающихся в процессе практических занятий |

Формы и методы контроля и оценки результатов обучения должны позволять проверять у обучающихся не только сформированность профессиональных компетенций,

но и развитие общих компетенций и обеспечивающих их умений.

| Результаты (освоенные общие компетенции) | Основные показатели оценки результата | Формы и методы контроля и оценки |
|---|---|--|
| ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес | <ul style="list-style-type: none"> - Демонстрация интереса к будущей профессии; - Участие в олимпиадах по специальности; - Составление портфолио студента. | Итоговый контроль знаний по дисциплинам профессионального модуля. Интерпретация результатов наблюдений за деятельностью обучающегося в процессе освоения образовательной программы |
| ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество | <ul style="list-style-type: none"> - Правильность выбора и аргументированность применения методов и способов решения профессиональных задач; - Точность оценки эффективности и качества выполнения профессиональных задач. - Рациональное распределение времени на все этапы решения задачи - совпадение результатов самоанализа и экспертного анализа разработанного плана - выбор и применение методов и способов решения профессиональных задач; - оценка эффективности и качества выполнения; | Экспертное наблюдение и оценка на практических и лабораторных занятиях при выполнении работ по учебной и производственной практике |
| ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность. | - Безошибочность принятия решения в стандартных и нестандартных ситуациях; | Текущий контроль знаний – индивидуальный опрос и практическая проверка. |

| | | |
|---|---|---|
| ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития. | <ul style="list-style-type: none"> - Быстрый и точный поиск необходимой информации; - Выполнение профессиональных задач с применением новых технологий для профессионального и личностного развития. | Экспертное наблюдение и оценка на практических и лабораторных занятиях. |
| ОК 5.Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности | <ul style="list-style-type: none"> - Решение нетиповых профессиональных задач с использованием различных источников информации; - Обоснованность использования информационно-коммуникационные технологии для совершенствования профессиональной деятельности. | Текущий контроль знаний – практическая проверка. Тестирование. |
| ОК 6.Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями | <ul style="list-style-type: none"> - Соблюдение мер конфиденциальности и информационной безопасности; - Использование приемов корректного межличностного общения. | Рубежный контроль знаний – практическая проверка. |
| ОК 7.Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за работу членов команды (подчиненных) и за результат выполнения заданий. | <ul style="list-style-type: none"> - Проведение контроля качества выполненной работы и нести ответственность в рамках профессиональной компетентности; | Рубежный контроль знаний – практическая проверка. |
| ОК 8.Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации | <ul style="list-style-type: none"> - Организация самостоятельных занятий при изучении профессиональных знаний и отечественного и зарубежного опыта; - Своевременность и осознанность планирования повышения квалификации. | Текущий контроль знаний – индивидуальный опрос. Защита творческих, проектных работ. |

| | | |
|--|--|--|
| ОК 9.Ориентироваться в условиях частой смены технологий в профессиональной деятельности. | <ul style="list-style-type: none"> - Анализ и использование инноваций в области профессиональной деятельности; - Результативность информационного поиска в условиях частой смены технологий в профессиональной деятельности. | Текущий контроль Практическая проверка. |
|--|--|--|

Программа воспитания в рабочей программе профессионального модуля отражается через содержание направлений воспитательной работы, разбитых на следующие воспитательные модули:

| Модули программы воспитания | Содержание модуля программы воспитания |
|--|---|
| Модуль 1 «Профессионально-личностное воспитание» | <p><i>Цель модуля:</i> создание условий для удовлетворения потребностей обучающихся в интеллектуальном, культурном и нравственном развитии в сфере трудовых и социально-экономических отношений посредством профессионального самоопределения.</p> <p><i>Задачи модуля:</i></p> <ul style="list-style-type: none"> – развитие общественной активности обучающихся, воспитание в них сознательного отношения к труду и народному достоянию; – формирование у обучающихся потребности трудиться, добросовестно, ответственно и творчески относиться к разным видам трудовой деятельности. – формирование профессиональных компетенций; – формирование осознания профессиональной идентичности (осознание своей принадлежности к определённой профессии и профессиональному сообществу); – формирование чувства социально-профессиональной ответственности, усвоение профессионально-этических норм; – осознанный выбор будущего профессионального развития и возможностей реализации собственных жизненных планов; – формирование отношения к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных про- |

| | |
|---|--|
| | блем. |
| <p>Модуль 2</p> <p>«Гражданско-патриотическое воспитание»</p> | <p><i>Цель модуля:</i> развитие личности обучающегося на основе формирования у обучающихся чувства патриотизма, гражданственности, уважения к памяти защитников Отечества и подвигам Героев Отечества, закону и правопорядку.</p> <p><i>Задачи модуля:</i></p> <ul style="list-style-type: none"> – формирование знаний обучающихся о символике России; – воспитание у обучающихся готовности к выполнению гражданского долга и конституционных обязанностей по защите Родины; – формирование у обучающихся патриотического сознания, чувства верности своему Отечеству; – развитие у обучающихся уважения к памяти защитников Отечества и подвигам Героев Отечества, историческим символам и памятникам Отечества; – формирование российской гражданской идентичности, гражданской позиции активного и ответственного члена российского общества, осознающего свои конституционные права и обязанности, уважающего закон и правопорядок, обладающего чувством собственного достоинства, осознанно принимающего традиционные национальные и общечеловеческие гуманистические и демократические ценности; – развитие правовой и политической культуры обучающихся, расширение конструктивного участия в принятии решений, затрагивающих их права и интересы, в том числе в различных формах общественной самоорганизации, самоуправления, общественно значимой деятельности; развитие в молодежной среде ответственности, принципов коллективизма и социальной солидарности; – формирование приверженности идеям интернационализма, дружбы, равенства, взаимопомощи народов; воспитание уважительного отношения к национальному достоинству людей, их чувствам, религиозным убеждениям; – формирование установок личности, позволяющих проти- |

| | |
|---|---|
| | <p>востоять идеологии экстремизма, национализма, ксенофобии, коррупции, дискриминации по социальным, религиозным, расовым, национальным признакам и другим негативным социальным явлениям;</p> <ul style="list-style-type: none"> – формирование антикоррупционного мировоззрения. |
| Модуль 3 «Физическая культура и здоровьесбережение» | <p><i>Цель модуля:</i> формирование у обучающихся чувства бережного отношения к культурному наследию и традициям многонационального народа России, культуры здоровья, безопасного поведения, стремления к здоровому образу жизни и занятиям спортом, воспитание психически здоровой, физически развитой и социально-адаптированной личности.</p> <p><i>Задачи модуля:</i></p> <ul style="list-style-type: none"> – формирование способности к духовному развитию, реализации творческого потенциала в учебной, профессиональной деятельности на основе нравственных установок и моральных норм, непрерывного образования, самовоспитания и универсальной духовно-нравственной компетенции - «становиться лучше»; – формирование у обучающихся ответственного отношения к своему здоровью и потребности в здоровом образе жизни, физическом самосовершенствовании, занятиях спортивно-оздоровительной деятельностью, развитие культуры безопасной жизнедеятельности, профилактику наркотической и алкогольной зависимости, табакокурения и других вредных привычек; – формирование бережного, ответственного и компетентного отношения к физическому и психологическому здоровью - как собственному, так и других людей, умение оказывать первую помощь, развитие культуры здорового питания. |
| Модуль 4 «Культурно-творческое воспитание» | <p><i>Цель модуля:</i> создание условий для самоопределения и социализации обучающихся на основе социокультурных, духовно-нравственных ценностей и принятых в российском обществе правил и норм поведения в интересах человека, семьи, общества и государства, формирование у обучающихся уважения к старшему поколению.</p> <p><i>Задачи модуля:</i></p> |

| | |
|--|---|
| | <ul style="list-style-type: none"> – воспитание здоровой, счастливой, свободной личности, формирование способности ставить цели и строить жизненные планы; – реализация обучающимися практик саморазвития и самовоспитания в соответствии с общечеловеческими ценностями и идеалами гражданского общества; – формирование позитивных жизненных ориентиров и планов; – формирование у обучающихся готовности и способности к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности; – формирование выраженной в поведении нравственной позиции, в том числе способности к сознательному выбору добра, нравственного сознания и поведения на основе усвоения общечеловеческих ценностей и нравственных чувств (чести, долга, справедливости, милосердия и дружелюбия); – развитие культуры межнационального общения; – формирование уважительного отношения к родителям и старшему поколению в целом, готовности понять их позицию, принять их заботу, готовности договариваться с родителями и членами семьи в решении вопросов ведения домашнего хозяйства, распределения семейных обязанностей; – воспитание ответственного отношения к созданию и сохранению семьи на основе осознанного принятия ценностей семейной жизни; – формирование толерантного сознания и поведения в поликультурном мире, готовности и способности вести диалог с другими людьми, достигать в нем взаимопонимания, находить общие цели и сотрудничать для их достижения. |
|--|---|

| | |
|--|--|
| <p>Модуль 5 «Экологическое воспитание»</p> | <p><i>Цель модуля:</i> формирование у обучающихся чувства бережного отношения к живой природе и окружающей среде, культурному наследию и традициям многонационального народа России.</p> <p><i>Задачи модуля:</i></p> <ul style="list-style-type: none"> – развитие у обучающихся экологической культуры, бережного отношения к родной земле, природным богатствам России и мира, понимание влияния социально-экономических процессов на состояние природной и социальной среды; – воспитание чувства ответственности за состояние природных ресурсов, формирование умений и навыков разумного природопользования, нетерпимого отношения к действиям, приносящим вред экологии; приобретение опыта эколого-направленной деятельности; – воспитание эстетического отношения к миру, включая эстетику быта, научного и технического творчества, спорта, общественных отношений; – формирование мировоззрения, соответствующего современному уровню развития науки и общественной практики, основанного на диалоге культур, а также на признании различных форм общественного сознания, предполагающего осознание своего места в поликультурном мире; – формирование чувства любви к Родине на основе изучения культурного наследия и традиций многонационального народа России. |
|--|--|

3 Типовые задания для оценки освоения МДК. 02.02. Программирование микроконтроллеров

Промежуточная аттестация по МДК 02.04 и темам предусматривает экзамен

Перечень вопросов по МДК. 02.02. Программирование микроконтроллеров

Тема 1. Особенности программирования микроконтроллеров STM32 или аналогов

Контрольные вопросы по теме «Составление простейшего алгоритма программы для системы на основе МК»:

1. Какова особенность систем на основе микроконтроллеров STM32?
 - a) Высокая производительность
 - b) Малый размер
 - c) Долгий срок службы
 - d) Низкое энергопотребление

2. Какова функция основного блока программы в системах на основе микроконтроллеров STM32?
 - a) Запуск и настройка периферийных устройств
 - b) Обработка данных
 - c) Контроль выполнения алгоритма
 - d) Все вышеперечисленное

3. Какая программа необходима для разработки алгоритмов для систем на базе микроконтроллеров STM32?
 - a) Keil MDK
 - b) Arduino IDE
 - c) Visual Studio Code
 - d) CodeBlocks

4. Какие языки программирования наиболее часто используются для разработки программ для микроконтроллеров STM32?

- a) C и C++
- b) Java
- c) Python
- d) JavaScript

5. Какова типичная последовательность действий для создания алгоритма для системы на основе микроконтроллеров STM32?

- a) Определение целей и задач системы, выбор периферийных устройств, разработка алгоритма, программирование и отладка
- b) Программирование, отладка, выбор периферийных устройств, разработка алгоритма, определение целей и задач системы
- c) Выбор периферийных устройств, определение целей и задач системы, разработка алгоритма, программирование и отладка
- d) Разработка алгоритма, определение целей и задач системы, выбор периферийных устройств, программирование и отладка

| | | | | | |
|---------|---|---|---|---|---|
| Вопросы | 1 | 2 | 3 | 4 | 5 |
| Ответы | a | d | a | a | a |

Контрольные вопросы по теме «Составление графа конечного автомата сложного алгоритма для системы на основе МК»:

1. Какие компоненты следует использовать при составлении графа конечного автомата для системы на основе микроконтроллеров STM32?

- a) Порты ввода-вывода
- b) Шина данных
- c) АЦП
- d) ЦПУ

2. Как можно описать состояния в графе конечного автомата для системы на основе микроконтроллеров STM32?

- a) С помощью диаграммы состояний
- b) С помощью таблицы состояний
- c) С помощью блок-схемы
- d) С помощью матрицы переходов

3. Какие типы событий можно использовать при составлении графа конечного автомата для системы на основе микроконтроллеров STM32?

- a) Интерактивные события (нажатия кнопок, сигналы сенсоров)
- b) Внешние события (подача питания, сигналы от других устройств)
- c) Временные события (таймеры, задержки)
- d) Все перечисленные варианты

4. Какие действия могут выполняться при переходе между состояниями в графе конечного автомата для системы на основе микроконтроллеров STM32?

- a) Установка или сброс флагов
- b) Изменение значения переменных
- c) Запуск алгоритмов или задач
- d) Все перечисленные варианты

5. Какие инструменты можно использовать для визуализации графа конечного автомата для системы на основе микроконтроллеров STM32?

- a) UML-диаграммы
- b) Блок-схемы
- c) Графические редакторы
- d) Все перечисленные варианты

| | | | | | |
|---------|---|---|---|---|---|
| Вопросы | 1 | 2 | 3 | 4 | 5 |
|---------|---|---|---|---|---|

| | | | | | |
|--------|---|---|---|---|---|
| Ответы | a | a | d | d | d |
|--------|---|---|---|---|---|

Итоговый тест по теме «Особенности программирования микроконтроллеров STM32 или аналогов»

1. Что такое микроконтроллер STM32?
 - a) Микроконтроллер от компании STMicroelectronics
 - b) Микроконтроллер для систем автоматизации
 - c) Микроконтроллер для встраиваемых систем

2. Какие языки программирования поддерживает STM32?
 - a) C и C++
 - b) Java и Python
 - c) Assembler и Pascal

3. Какой тип ядра используется в микроконтроллерах STM32?
 - a) ARM Cortex-R
 - b) ARM Cortex-M
 - c) ARM Cortex-A

4. Какая особенность микроконтроллеров STM32 позволяет уменьшить потребление энергии?
 - a) Возможность работы в спящем режиме
 - b) Использование батарей с низким энергопотреблением
 - c) Отсутствие потерь памяти при перезагрузке

5. Какая разрядность используется в микроконтроллерах STM32?
 - a) 8 бит
 - b) 16 бит
 - c) 32 бита

6. Какие интерфейсы связи поддерживают микроконтроллеры STM32?

- a) SPI, I2C, UART
- b) USB, Ethernet, HDMI
- c) VGA, RS-232, CAN

7. Какие периферийные устройства встречаются в микроконтроллерах STM32?

- a) Таймеры, АЦП, ЦАП
- b) Принтер, сканер, клавиатура
- c) Микрофон, динамик, камера

8. Какой инструмент используется для программирования микроконтроллеров STM32?

- a) Keil MDK
- b) Visual Studio
- c) Eclipse

9. Какой тип памяти применяется в микроконтроллерах STM32?

- a) Flash-память
- b) Оперативная память
- c) Жесткий диск

10. Какие средства отладки поддерживают микроконтроллеры STM32?

- a) JTAG и SWD
- b) USB и Bluetooth
- c) Ethernet и Wi-Fi

11. Содержат ли микроконтроллеры STM32 встроенные компиляторы?

- a) Нет, требуется использование внешних компиляторов

- b) Да, встроенный компилятор GCC
- c) Да, встроенный компилятор Keil

12. Какой максимальный тактовый сигнал поддерживают микроконтроллеры STM32?

- a) 32 МГц
- b) 72 МГц
- c) 120 МГц

13. Какой тип архитектуры используется в микроконтроллерах STM32?

- a) RISC
- b) CISC
- c) MIPS

14. Какие входы-выходы имеют микроконтроллеры STM32?

- a) Цифровые и аналоговые
- b) Только цифровые
- c) Только аналоговые

15. Какая режимная система работает в микроконтроллерах STM32?

- a) Nucleo
- b) CubeSat
- c) CubeMx

16. Какие типы корпусов микроконтроллеров STM32 существуют?

- a) LQFP, BGA, TSSOP
- b) CD, DVD, Blu-ray
- c) TV, монитор, проектор

| | | | | | | | | | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| Ответы | a | a | b | a | c | a | a | a | a | a | a | c | a | a | c | a |

Тема 2. Модульное программирование микроконтроллеров STM32 или аналогов

Контрольные вопросы по теме «Работа с памятью МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. На каком языке программирования можно работать с памятью микроконтроллеров STM32?

- a) C/C++
- b) Java
- c) Python
- d) JavaScript

2. Какие функции микроконтроллера STM32 необходимо использовать для работы с памятью?

- a) malloc() и free()
- b) write() и read()
- c) allocate() и deallocate()
- d) memory() и access()

3. Какой тип памяти использовать для хранения постоянных данных в микроконтроллере STM32?

- a) SRAM
- b) DRAM
- c) Flash
- d) Cache

4. Какие макросы можно использовать для определения расположения данных в памяти микроконтроллера STM32?

- a) __attribute__
- b) #define

- c) malloc()
- d) sizeof()

5. Какая функция используется для записи данных в память микроконтроллера STM32?

- a) memcpu()
- b) malloc()
- c) allocate()
- d) write()

6. Какие функции используются для чтения данных из памяти микроконтроллера STM32?

- a) memcpu() и read()
- b) malloc() и access()
- c) read() и write()
- d) allocate() и deallocate()

7. Какой размер памяти доступен для программиста при работе с микроконтроллером STM32?

- a) От 1 КБ до нескольких МБ
- b) От 1 МБ до нескольких ГБ
- c) От 1 КБ до нескольких ГБ
- d) От 1 МБ до нескольких ТБ

8. Какие методы существуют для оптимизации использования памяти микроконтроллера STM32?

- a) Удаление неиспользуемого кода и данных
- b) Увеличение размера памяти микроконтроллера
- c) Использование дополнительных устройств хранения данных
- d) Запись данных во внешнюю память

9. Какие особенности работы с памятью нужно учитывать при программировании микроконтроллеров STM32?

- a) Ограниченный объем доступной памяти
- b) Ограниченные скорости чтения и записи
- c) Возможность потери данных при сбое питания
- d) Возможность переполнения стека при неверном управлении памятью

| | | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|-----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Ответы | a | b | c | a | d | a | a | a | abd |

Контрольные вопросы по теме «Работа с подсистемой ввода/вывода МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. В какой подсистеме микроконтроллеров STM32 осуществляется работа с вводом/выводом?

- A. Подсистема GPIO
- B. Подсистема USART
- C. Подсистема ADC
- D. Подсистема SPI

2. Какая функция используется для настройки вывода микроконтроллера STM32 на вход или выход?

- A. GPIO_Init()
- B. USART_Init()
- C. ADC_Init()
- D. SPI_Init()

3. Какая функция используется для установки значения вывода микроконтроллера STM32 в высокий или низкий уровень?

- A. GPIO_Set()
- B. GPIO_Clear()
- C. GPIO_Toggle()
- D. GPIO_Read()

4. Какая функция используется для настройки и передачи данных по последовательному интерфейсу USART?

- A. USART_Config()
- B. USART_Send()
- C. USART_Receive()
- D. USART_Transmit()

5. Какая функция используется для настройки и чтения аналогового сигнала с помощью АЦП микроконтроллера STM32?

- A. ADC_Config()
- B. ADC_Start()
- C. ADC_Read()
- D. ADC_Stop()

6. Какая функция используется для настройки и передачи данных по последовательному интерфейсу SPI микроконтроллера STM32?

- A. SPI_Config()
- B. SPI_Transmit()
- C. SPI_Receive()
- D. SPI_SetMode()

| | | | | | | |
|---------|---|---|---|---|---|---|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 |
| Ответы | a | a | b | d | c | a |

Контрольные вопросы по теме «Работа с последовательным интерфейсом МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. Какие языки программирования подходят для работы с последовательным интерфейсом микроконтроллеров STM32 - C, C++, или оба варианта?

- А) C
- Б) C++
- В) Оба

2. Какой высокоуровневый язык программирования предпочтительнее использовать для работы с последовательным интерфейсом микроконтроллеров STM32?

- А) C
- Б) C++
- В) Нет предпочтений

3. Каким образом можно осуществить связь микроконтроллера STM32 с другими устройствами через последовательный интерфейс?

- А) SPI
- Б) I2C
- В) UART
- Г) Все вышеперечисленное

4. Какой тип передачи данных поддерживается последовательным интерфейсом микроконтроллера STM32?

- А) Полудуплексная
- Б) Полная дуплексная
- В) Полудуплексная и полная дуплексная

5. Какая основная функция используется для работы с последовательным интерфейсом микроконтроллера STM32 в языке C/C++?

- А) HAL_UART_Transmit()
- Б) HAL_I2C_Master_Transmit()
- В) HAL_SPI_Transmit()
- Г) HAL_GPIO_WritePin()

6. Какой библиотека предоставляет функциональность для работы с последовательным интерфейсом микроконтроллера STM32 в языке C/C++?

- А) STM32Cube HAL
- Б) Arduino
- В) Keil
- Г) MPLAB X IDE

7. Как включить поддержку последовательного интерфейса в среде разработки для микроконтроллера STM32?

- А) Подключить соответствующую библиотеку
- Б) Выбрать необходимые пины для последовательного интерфейса
- В) Включить конфигурацию в настройках проекта
- Г) Все вышеперечисленное

| | | | | | | | |
|---------|---|---|---|---|---|---|---|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Ответы | в | в | г | в | а | а | г |

Контрольные вопросы по теме «Работа с системой прерываний МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. Какие микроконтроллеры поддерживают систему прерываний в STM32?
 - a. STM32F0
 - b. STM32F4
 - c. STM32L1

2. Какая функция используется для инициализации системы прерываний в STM32?
 - a. NVIC_Init()
 - b. EXTI_Init()
 - c. SysTick_Init()

3. Как можно обработать прерывание в STM32?
 - a. Используя функцию обработчика прерывания
 - b. Отключить систему прерываний
 - c. Использовать таймер

4. Какой регистр используется для установки приоритета прерывания в STM32?
 - a. NVIC_PriorityGroupConfig()
 - b. NVIC_SetPriority()
 - c. NVIC_GetPriority()

5. Какой регистр используется для разрешения прерывания в STM32?
 - a. NVIC_EnableIRQ()
 - b. NVIC_DisableIRQ()
 - c. EXTI_Init()

6. Какое прерывание используется для обработки таймера в STM32?
- TIM1_IRQn
 - USART1_IRQn
 - EXTI0_IRQn
7. Как передать данные из прерывания в основную программу в STM32?
- Используя глобальные переменные
 - Используя функцию обработчика прерывания
 - Используя DMA
8. Как определить источник прерывания в STM32?
- Чтение регистра EXTI_PR
 - Чтение регистра GPIOA_IDR
 - Чтение регистра TIM_SR
9. Какова максимальная задержка между поступлением запроса на прерывание и его обработкой в STM32?
- 1 такт процессора
 - 10 тактов процессора
 - 100 тактов процессора
10. Как можно изменить приоритет прерывания во время выполнения программы в STM32?
- Изменить значение в таблице векторов прерываний
 - Использовать функцию NVIC_SetPriority()
 - Использовать функцию NVIC_PriorityGroupConfig()

| | | | | | | | | | | |
|---------|-----|---|---|---|---|---|---|---|---|----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Ответы | abc | a | a | b | a | a | a | a | b | b |

Контрольные вопросы по теме «Работа с таймерами счетчиками МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. Каким образом можно работать с таймерами и счетчиками микроконтроллеров STM32 на языке C/C++?

- а) Используя библиотеку HAL
- б) Написав собственные функции управления таймерами и счетчиками
- в) Используя стандартные функции языка C/C++

2. Какая функция используется для инициализации таймера в STM32 microcontroller?

- а) HAL_TIM_Base_Init()
- б) TIM_Init()
- в) TIM_Configuration()

3. Какой режим работы таймера позволяет генерировать прерывания с определенной периодичностью?

- а) Режим захвата
- б) Режим сравнения
- в) Режим основного счетчика

4. Какой регистр нужно настроить для задания периода работы таймера?

- а) TIM_Period
- б) TIM_Prescaler
- в) TIM_Counter

5. Какую функцию следует использовать для включения прерывания от таймера?

- а) HAL_TIM_Base_Start_IT()

- b) TIM_ITConfig()
- c) NVIC_EnableIRQ()

6. Какая функция позволяет обработать прерывание от таймера?

- a) HAL_TIM_PeriodElapsedCallback()
- b) TIM_IRQHandler()
- c) TIM_UpdateRequest()

7. Какой регистр нужно изменить для изменения текущего значения счетчика таймера?

- a) TIM_Prescaler
- b) TIM_Counter
- c) TIM_Period

8. Какая функция нужна для остановки работы таймера?

- a) HAL_TIM_Base_Stop()
- b) TIM_Stop()
- c) TIM_ClearFlag()

9. Какой режим работы счетчика позволяет измерить время между двумя событиями?

- a) Режим захвата
- b) Режим сравнения
- c) Режим основного счетчика

10. Какой регистр содержит текущее значение счетчика таймера?

- a) TIM_Counter
- b) TIM_Period
- c) TIM_Prescaler

| | | | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|---|----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Ответы | a | a | c | a | a | a | b | a | a | a |

Контрольные вопросы по теме «Работа с модулем DMA на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. Какой модуль используется для работы с DMA в микроконтроллерах STM32?

- a) USART
- b) GPIO
- c) ADC
- d) DMA

2. Для чего используется модуль DMA?

- a) Для работы с периферийными устройствами
- b) Для работы с внешней памятью
- c) Для работы с прерываниями
- d) Для работы с таймером

3. Каким образом осуществляется передача данных с помощью DMA?

- a) Через прерывания
- b) Через USART
- c) Через SPI
- d) Через DMA-каналы

4. Какие операции можно выполнять с помощью DMA?

- a) Чтение данных из периферийного устройства
- b) Запись данных в периферийное устройство
- c) Передача данных из памяти в периферийное устройство
- d) Все вышеперечисленные операции

5. Какой язык программирования используется для работы с модулем DMA?

- a) C++
- b) Python
- c) Java
- d) C

6. Можно ли работать с модулем DMA на высокоуровневом языке (C/C++)?

- a) Да
- b) Нет

7. Возможно ли использовать несколько DMA-каналов одновременно?

- a) Да
- b) Нет

8. Каким образом происходит настройка DMA-канала?

- a) Через конфигурационные регистры
- b) Через специальные команды
- c) Через прерывания
- d) Через модуль USART

| | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Ответы | d | a | d | d | d | a | a | a |

Контрольные вопросы по теме «Работа с синхронными интерфейсами МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1) Какие языки программирования используются для работы с синхронными интерфейсами микроконтроллеров STM32?

- a) C/C++
- b) Java
- c) Python
- d) JavaScript

2) Какой из следующих языков программирования чаще всего используется для работы с синхронными интерфейсами микроконтроллеров STM32?

- a) C/C++
- b) Java
- c) Python
- d) JavaScript

3) Какой интерфейс является синхронным интерфейсом в микроконтроллерах STM32?

- a) UART
- b) SPI
- c) I2C
- d) GPIO

4) Какие функции API используются для работы с синхронными интерфейсами микроконтроллеров STM32 в C/C++?

- a) USART_Init()
- b) SPI_Init()
- c) I2C_Init()
- d) GPIO_Init()

5) Какие параметры можно настроить при инициализации синхронного интерфейса SPI в микроконтроллерах STM32?

- a) Скорость передачи данных
- b) Режим передачи данных (полудуплекс, полный дуплекс)

- c) Размер данных в пакете
- d) Все вышеперечисленное

6) Какие функции API используются для передачи данных по синхронному интерфейсу SPI в микроконтроллерах STM32 в C/C++?

- a) USART_SendData()
- b) SPI_SendData()
- c) I2C_SendData()
- d) GPIO_SetBits()

7) Какие функции API используются для приема данных по синхронному интерфейсу SPI в микроконтроллерах STM32 в C/C++?

- a) USART_ReceiveData()
- b) SPI_ReceiveData()
- c) I2C_ReceiveData()
- d) GPIO_ReadInputData()

8) Какие типы прерываний можно использовать при работе с синхронными интерфейсами микроконтроллеров STM32 в C/C++?

- a) Прерывания по приему данных
- b) Прерывания по передаче данных
- c) Прерывания по ошибкам передачи данных
- d) Все вышеперечисленное

| | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Ответы | a | a | b | b | d | b | b | d |

Контрольные вопросы по теме «Работа с режимами потребления МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. Какие режимы потребления доступны для микроконтроллеров STM32?
 - a) Операционный режим
 - b) Режим сна
 - c) Режим ожидания
 - d) Режим остановки
2. Какой режим потребления следует выбрать для минимизации энергопотребления микроконтроллера в активном состоянии?
 - a) Операционный режим
 - b) Режим сна
 - c) Режим ожидания
 - d) Режим остановки
3. Что происходит с микроконтроллером в режиме сна?
 - a) Происходит остановка работы ядра и переход в низкопотребляющий режим
 - b) Микроконтроллер продолжает работу в полной мощности
 - c) Микроконтроллер переходит в режим ожидания
 - d) Микроконтроллер полностью останавливается
4. В чем отличие режима сна от режима ожидания?
 - a) В режиме сна микроконтроллер продолжает работать с низким энергопотреблением, в режиме ожидания - останавливается
 - b) В режиме сна микроконтроллер полностью останавливается, в режиме ожидания продолжает работу с низким энергопотреблением
 - c) Режимы сна и ожидания эквивалентны и означают одно и то же
 - d) Режимы сна и ожидания отличаются только названиями, но выполняют одинаковые функции
5. Какие действия требуются для выхода из режима сна или ожидания?

- a) Изменение состояния входов/выходов
- b) Внешнее прерывание
- c) Работа по таймеру или другому событию
- d) Все перечисленные варианты ответов

| | | | | | |
|---------|---|---|---|---|---|
| Вопросы | 1 | 2 | 3 | 4 | 5 |
| Ответы | a | a | a | a | d |

Контрольные вопросы по теме «Работа с внешней памятью в МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. Какое преимущество имеет работа с внешней памятью в микроконтроллерах STM32?

- a) Увеличение доступного объема памяти
- b) Ускорение работы микроконтроллера
- c) Улучшение энергоэффективности

2. Какие типы внешней памяти поддерживаются микроконтроллерами STM32?

- a) SPI Flash
- b) EEPROM
- c) SD Card

3. Как осуществляется подключение внешней памяти к микроконтроллеру STM32?

- a) Через шину I2C
- b) Через шину SPI
- c) Через шину UART

4. Какие функции необходимо использовать для работы с внешней памятью в STM32?

- a) memcpu()
- b) read()
- c) HAL_FLASH_Program()

5. Как происходит чтение данных из внешней памяти в STM32?

- a) Через функцию HAL_FLASH_Read()
- b) Через функцию HAL_SPI_Receive()
- c) Через функцию HAL_UART_Receive()

6. Как происходит запись данных во внешнюю память в STM32?

- a) Через функцию HAL_FLASH_Program()
- b) Через функцию HAL_SPI_Transmit()
- c) Через функцию HAL_UART_Transmit()

7. Как проверить успешность операции записи данных во внешнюю память в STM32?

- a) Проверить значение возвращаемое функцией HAL_FLASH_Program()
- b) Проверить значение флага по окончании операции записи
- c) Проверить значение регистра статуса внешней памяти

8. Как удалить данные из внешней памяти в STM32?

- a) Через функцию HAL_FLASH_Erase()
- b) Через функцию HAL_SPI_Erase()
- c) Через функцию HAL_UART_Erase()

9. Какие ограничения есть при работе с внешней памятью в STM32?

- a) Ограничения по объему памяти
- b) Ограничения по допустимой частоте работы

с) Ограничения по длительности операций чтения/записи

10. Какие дополнительные настройки необходимо выполнить для работы с внешней памятью в STM32?

- а) Настройка тактирования
- б) Настройка прерываний
- с) Настройка DMA

| | | | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|---|----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Ответы | a | a | b | c | b | a | a | a | a | a |

Контрольные вопросы по теме «Работа с АЦП/ЦАП МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. Какой микроконтроллер STM32 поддерживает работу с АЦП/ЦАП?

- а) STM32F030
- б) STM32F407
- с) STM32L476

2. Какая библиотека используется для работы с АЦП/ЦАП на микроконтроллерах STM32?

- а) HAL
- б) CMSIS
- с) SPL

3. Какой функцией можно настроить режим работы АЦП в микроконтроллере STM32?

- а) ADC_Init()
- б) ADC_Config()

c) ADC_Setup()

4. Каким образом происходит запись в регистр АЦП на микроконтроллере STM32?

- a) Запись напрямую в регистр
- b) Использование специальных функций библиотеки
- c) Использование прерываний

5. Какой функцией можно считать значение, преобразованное АЦП на микроконтроллере STM32?

- a) ADC_Read()
- b) ADC_GetValue()
- c) ADC_ReadValue()

6. Какая функция используется для настройки режима работы ЦАП на микроконтроллере STM32?

- a) DAC_Init()
- b) DAC_Config()
- c) DAC_Setup()

7. Какие каналы ЦАП могут быть использованы на микроконтроллере STM32?

- a) Один
- b) Два
- c) Четыре

8. Каким образом происходит запись значения в регистр ЦАП на микроконтроллере STM32?

- a) Запись напрямую в регистр
- b) Использование специальных функций библиотеки

c) Использование прерываний

9. Какой функцией можно установить значение, которое будет преобразовано в аналоговый сигнал на микроконтроллере STM32?

- a) DAC_SetValue()
- b) DAC_Write()
- c) DAC_SendValue()

10. Какая функция используется для запуска преобразования в АЦП на микроконтроллере STM32?

- a) ADC_StartConversion()
- b) ADC_Convert()
- c) ADC_BeginConversion()

| | | | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|---|----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Ответы | b | a | a | b | b | a | c | b | a | a |

Контрольные вопросы по теме «Работа с USB в МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. Какой высокоуровневый язык программирования можно использовать для работы с USB в микроконтроллерах STM32?

- a) C++
- b) C
- c) Java
- d) Python

2. Какой API следует использовать для работы с USB в микроконтроллерах STM32?

- a) USB-IF API

- b) USB to UART API
- c) USBж API
- d) USB OTG API

3. Какой интерфейс обеспечивает связь между микроконтроллером STM32 и персональным компьютером через USB?

- a) USB to UART
- b) USB to SPI
- c) USB to I2C
- d) USB HID

4. Какие функции отвечают за инициализацию USB в микроконтроллерах STM32?

- a) usb_init()
- b) usb_start()
- c) usb_enable()
- d) USB_Init()

5. Какой тип передачи данных можно использовать при работе с USB в микроконтроллерах STM32?

- a) Bulk
- b) Serial
- c) Parallel
- d) Interrupt

6. Какое преимущество имеет использование USB в микроконтроллерах STM32?

- a) Высокая скорость передачи данных
- b) Низкое энергопотребление
- c) Простота подключения к персональному компьютеру

d) Возможность подключения множества устройств через один порт USB

7. Какой функцией можно отправить данные через USB в микроконтроллерах STM32?

- a) `usb_send()`
- b) `usb_write()`
- c) `USB_sendData()`
- d) `USB_WriteData()`

8. Каким образом можно обработать прерывание от USB в микроконтроллерах STM32?

- a) Используя функцию `interrupt_handler()`
- b) Настройкой прерываний через USB API
- c) Используя функцию `USB_IRQHandler()`
- d) Нет возможности обработать прерывание от USB в микроконтроллерах STM32

9. Какие дополнительные компоненты требуются для работы с USB в микроконтроллерах STM32?

- a) USB драйвер
- b) USB конвертер
- c) USB кабель
- d) USB хаб

10. Какие функции отвечают за чтение данных из USB в микроконтроллерах STM32?

- a) `usb_read()`
- b) `usb_receive()`
- c) `USB_ReadData()`
- d) `USB_ReceiveData()`

| | | | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|---|----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Ответы | b | d | a | d | a | d | c | c | a | c |

Контрольные вопросы по теме «Работа с высокоуровневыми стеками в МК на высокоуровневом языке (C/C++). Типовые алгоритмы и программные модули»:

1. Какое преимущество представляет работа с высокоуровневыми стеками в микроконтроллерах STM32 на высокоуровневом языке (C/C++)?
 - a. Упрощение разработки и отладки кода
 - b. Увеличение производительности и эффективности работы контроллера
 - c. Облегчение интеграции с другими устройствами и программным обеспечением
 - d. Все вышеперечисленное
2. Какой язык программирования используется для работы с высокоуровневыми стеками в микроконтроллерах STM32?
 - a. C
 - b. C++
 - c. Оба варианта языка программирования могут быть использованы
3. Какие высокоуровневые стеки доступны для работы с микроконтроллерами STM32?
 - a. TCP/IP
 - b. USB
 - c. CAN
 - d. Ethernet
 - e. Все вышеперечисленные

4. Какие протоколы поддерживаются высокоуровневыми стеками в микроконтроллерах STM32?

- a. MQTT
- b. HTTP
- c. FTP
- d. MODBUS
- e. Все вышеперечисленные

5. Какая функциональность предоставляется высокоуровневыми стеками в микроконтроллерах STM32?

- a. Обработка сенсорных данных
- b. Работа с файлами и хранение данных
- c. Контроль и обработка сетевых соединений
- d. Все вышеперечисленное

6. Можно ли использовать высокоуровневые стеки в микроконтроллерах STM32 для разработки интернета вещей (IoT) приложений?

- a. Да
- b. Нет

7. Возможно ли использовать высокоуровневые стеки в микроконтроллерах STM32 для создания многопоточных приложений?

- a. Да
- b. Нет

8. Какие средства разработки подходят для работы с высокоуровневыми стеками в микроконтроллерах STM32?

- a. Keil
- b. STM32CubeIDE
- c. PlatformIO

d. Arduino IDE

е. Все вышеперечисленные

| | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Ответы | d | c | e | e | d | a | a | e |

**Итоговый тест по теме «Модульное программирование
микроконтроллеров STM32 или аналогов»**

1. Что такое модульное программирование?

- a) Техника программирования, которая позволяет разделять код на небольшие функциональные блоки
- b) Техника программирования, которая используется только для микроконтроллеров STM32
- c) Техника программирования, которая применяется только для аналоговых устройств

2. Какие преимущества дает модульное программирование?

- a) Упрощение отладки кода
- b) Улучшение повторного использования кода
- c) Облегчение разработки сложных проектов

3. Какие инструменты используются для разработки модульного программного обеспечения на STM32?

- a) STM32CubeMX
- b) STM32CubeIDE
- c) Компиляторы для STM32

4. Что такое HAL (Hardware Abstraction Layer) в STM32?

- a) Библиотека программных функций, которая обеспечивает абстракцию от аппаратного уровня
- b) Библиотека, которая позволяет программировать аналоговые устройства на STM32
- c) Инструмент для создания периферийных драйверов на STM32

5. Что такое CMSIS (Cortex Microcontroller Software Interface Standard)?

- a) Стандартный интерфейс программирования для микроконтроллеров Cortex-M
- b) Библиотека, которая используется только для модульного программирования на STM32
- c) Инструмент для разработки аналоговых устройств

6. Какие функции выполняет startup файл при запуске микроконтроллера?

- a) Настройка системных тактовых генераторов и сторожевого таймера
- b) Инициализация периферийных устройств
- c) Загрузка программного кода в память и запуск исполнения

7. Какие периферийные устройства можно программировать с использованием модульного программирования на STM32?

- a) GPIO (General Purpose Input/Output)
- b) UART (Universal Asynchronous Receiver/Transmitter)
- c) ADC (Analog-to-Digital Converter)

8. Какая функция используется для настройки пинов GPIO на STM32?

- a) HAL_GPIO_Init()
- b) HAL_ADC_Init()
- c) HAL_UART_Init()

9. Какая функция используется для отправки данных через UART на STM32?

- a) HAL_UART_Transmit()
- b) HAL_ADC_Start()
- c) HAL_GPIO_WritePin()

10. Какая функция используется для настройки ADC на STM32?

- a) HAL_ADC_Init()
- b) HAL_UART_Transmit()
- c) HAL_GPIO_Init()

11. Как можно прочитать значение с ADC на STM32?

- a) HAL_ADC_Start()
- b) HAL_ADC_GetValue()
- c) HAL_UART_Receive()

12. Что такое прерывания (interrupts) в STM32?

- a) Специальные сигналы, которые возникают при определенных событиях и прерывают нормальное выполнение программы
- b) Библиотека, которая используется только для модульного программирования на STM32
- c) Инструмент для разработки аналоговых устройств

13. Какая функция используется для настройки прерываний на STM32?

- a) HAL_NVIC_SetPriority()
- b) HAL_GPIO_Init()
- c) HAL_UART_Init()

14. Как можно обработать прерывание на STM32?

- a) Создать функцию-обработчик и зарегистрировать ее с помощью HAL_NVIC_SetPriority()
- b) Использовать HAL_ADC_Init()

с) Использовать HAL_GPIO_Init()

| | | | | | | | | | | | | | | |
|---------|---|-----|-----|---|---|----|-----|---|---|----|----|----|----|----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| Ответы | a | abc | abc | a | a | ac | abc | a | a | a | b | a | a | a |

Тема 3. Автоматизация процессов на основе систем с микроконтроллерами STM32 или аналогов

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Дисплей символьный» на основе МК»:

1. Какой язык программирования используется при создании алгоритма и программы для системы «Дисплей символьный» на основе микроконтроллеров STM32?
 - a) C/C++
 - b) Python
 - c) Java
 - d) JavaScript

2. Какая система используется в качестве основы для разработки программы для системы «Дисплей символьный» на микроконтроллерах STM32?
 - a) STM32Cube
 - b) Arduino
 - c) Raspberry Pi
 - d) PIC

3. Какая функция отвечает за вывод текста на дисплей в программе для системы «Дисплей символьный» на микроконтроллерах STM32?
 - a) lcd.print()
 - b) displayText()
 - c) printText()
 - d) showText()

4. Какая команда используется для очистки дисплея в программе для системы «Дисплей символьный» на микроконтроллерах STM32?

- a) lcd.clear()
- b) clearScreen()
- c) cleanDisplay()
- d) displayClear()

5. Каким образом подключается дисплей к микроконтроллеру STM32?

- a) По USB
- b) По HDMI
- c) По I2C
- d) По Ethernet

6. Какая функция отвечает за установку курсора на дисплее в программе для системы «Дисплей символьный» на микроконтроллерах STM32?

- a) lcd.setCursor()
- b) setCursorPosition()
- c) moveCursor()
- d) setCursorPos()

7. Какая библиотека используется при программировании системы «Дисплей символьный» на микроконтроллерах STM32?

- a) HAL
- b) Standard
- c) LiquidCrystal
- d) Wire

8. Какой микроконтроллер является основой для системы «Дисплей символьный»?

- a) STM32F103C8T6
- b) ATmega328P
- c) ESP8266

d) Raspberry Pi 4

9. Какой тип дисплея используется в системе «Дисплей символьный»?

- a) Жидкокристаллический (LCD)
- b) OLED
- c) TFT
- d) Электrolюминесцентный (EL)

10. Каким образом загружается программа на микроконтроллер STM32?

- a) Через USB
- b) Через Wi-Fi
- c) Через Ethernet
- d) Через SPI

| | | | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|---|----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Ответы | a | a | a | a | c | a | c | a | a | a |

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Дисплей графический» на основе МК»:

1)Что такое микроконтроллер?

Микроконтроллер - это компактное электронное устройство, интегрированная микросхема, которая объединяет в себе процессор, память и периферийные устройства. Он предназначен для управления различными устройствами или системами.

2)Для чего используется графический дисплей?

Графический дисплей используется для отображения информации в виде графики или текста. Он может быть использован в различных устройствах,

таких как мобильные телефоны, компьютеры, телевизоры, навигационные системы и т.д.

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Дисплей 7-сегментный» на основе МК»:

1. Что представляет собой система семисегментного дисплея?

Система семисегментного дисплея представляет собой устройство, состоящее из семи сегментов, которые могут быть включены или выключены для отображения цифр или букв. Она обычно используется для отображения цифровой информации.

2. Какие типы микроконтроллеров подходят для управления 7-сегментным дисплеем?

Для управления 7-сегментным дисплеем подходят различные типы микроконтроллеров, такие как AVR, PIC, ARM, Arduino и многие другие. Возможностей для управления 7-сегментным дисплеем предоставляют много типов микроконтроллеров.

3. В чем заключается принцип работы алгоритма управления 7-сегментным дисплеем?

Принцип работы алгоритма управления 7-сегментным дисплеем заключается в последовательном включении и выключении сегментов для отображения определенных символов или цифр. Алгоритм может использовать данные, сохраненные в памяти микроконтроллера или полученные от внешних источников, для определения, какие сегменты должны быть включены или выключены.

4. Какие возможности по управлению отображением информации на 7-сегментном дисплее предоставляются выбранным микроконтроллером?

Многие микроконтроллеры предоставляют возможность управлять отображением информации на 7-сегментном дисплее с помощью программирования или ввода с внешних источников. Они могут определять, какие символы или цифры должны быть отображены, а также контролировать яркость и освещение дисплея.

5. Каковы применения 7-сегментного дисплея в различных устройствах и системах?

7-сегментный дисплей может быть использован в различных устройствах и системах для отображения цифровой информации, такой как часы, термометры, счетчики, приборы измерения и многие другие. Он обеспечивает простой и понятный способ отображения чисел или символов.

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Кнопки управления» на основе МК»:

1) Что такое система "Кнопки управления"?

Система "Кнопки управления" представляет собой устройство, состоящее из клавиш или кнопок, с помощью которых пользователь может управлять различными функциями или операциями.

2) Как работает алгоритм системы "Кнопки управления"?

Алгоритм системы "Кнопки управления" обычно работает следующим образом:

- Считывает состояние клавиш или кнопок с помощью датчиков
- Обработывает полученные данные
- Принимает решение о необходимых действиях на основе полученной информации
- Выполняет соответствующие действия, такие как регулировка громкости, переключение каналов, передача команд и прочее.

3) Как создать алгоритм и программу для системы "Кнопки управления"?

Для создания алгоритма и программы для системы "Кнопки управления" можно использовать различные языки программирования, такие как C, C++, Python, Java и другие. Выбор языка будет зависеть от конкретной задачи и возможностей программиста.

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Матрица клавиатуры» на основе МК»:

1) Какой функционал должна обеспечивать программа для системы «Матрица клавиатуры»?

- Считывание нажатых кнопок или клавиш и определение соответствующей комбинации
- Обработка и интерпретация введенных команд или символов
- Взаимодействие с другими устройствами или программами для выполнения соответствующих действий.

2) Какие языки программирования можно использовать для разработки алгоритма и программы «Матрица клавиатуры»?

C, C++, Python, Java и другие. Выбор языка будет зависеть от требований проекта, возможностей программиста и доступных инструментов.

3) Какие шаги необходимо выполнить для разработки алгоритма и программы «Матрица клавиатуры»?

- Анализ требований и определение необходимого функционала
- Проектирование структуры программы и алгоритма
- Написание и отладка кода
- Тестирование и оптимизация программы
- Документирование и поддержка программы

4) Какие возможные проблемы могут возникнуть при разработке программы для системы «Матрица клавиатуры»?

- Необходимость обработки большого количества различных комбинаций клавиш

- Ошибки в считывании клавиш или кнопок, такие какдребезг контактов
- Конфликты между различными функциями или комбинациями клавиш
- Сложность взаимодействия с другими устройствами или программами

5) Какие факторы необходимо учесть при выборе микроконтроллера для системы «Матрица клавиатуры»?

- Необходимое количество и тип клавиш или кнопок

- Возможности обработки и считывания сигналов с клавиатуры
- Наличие достаточного числа входов или портов для работы с матрицей

клавиш

- Интеграция с другими устройствами или программами

6) Какие методы можно использовать для определения координаты нажатой клавиши в матрице клавиш?

- Метод сканирования матрицы клавиш последовательно или одновременно
- Использование алгоритма поиска определенной комбинации кнопок или

клавиш

- Таблица соответствия координат клавиш и символов

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Энкодер» на основе МК»:

1) Какие функциональные требования должны быть учтены при создании алгоритма и программы для системы "Энкодер"?

- Установление точной позиции и скорости вращающегося объекта
- Определение направления вращения

- Предоставление выходных данных в реальном времени
- Возможность настройки и калибровки системы
- Обработка и фильтрация шумов и помех
- Интерфейс с другими системами или устройствами для передачи данных

2) Какие аппаратные компоненты потребуются для реализации системы "Энкодер"?

- Вращающийся объект с энкодером
- Микроконтроллер или PLC (программируемый логический контроллер) для обработки сигналов энкодера
- Интерфейсные платы или модули для связи между энкодером и микроконтроллером/PLC
- Дисплей или монитор для отображения данных
- Питание для системы

3) Какие языки программирования можете использоваться для написания программы для системы "Энкодер"?

- C/C++
- Python
- Java
- MATLAB
- LabVIEW
- Assembler

4) Какие алгоритмы могут использоваться для определения позиции и скорости вращающегося объекта с помощью энкодера?

- Алгоритм счета импульсов: подсчет числа импульсов, отправляемых энкодером, для определения позиции и скорости
- Интерполяционные алгоритмы: используются для увеличения точности путем добавления промежуточных значений между импульсами

- Фильтрация сигнала: применение фильтров для удаления шумов и помех из сигнала энкодера

5) Какие технические ограничения нужно учитывать при создании программы для системы "Энкодер"?

- Частота обновления данных: необходимо определить, в каком временном интервале должны обновляться данные в программе
- Точность и разрешение энкодера: нужно учесть максимальную точность и разрешение энкодера при обработке данных
- Возможность обработки большого объема данных в реальном времени: программа должна быть способна обрабатывать данные от энкодера в реальном времени без задержек

6) Какая информация может быть получена с помощью системы "Энкодер"?

- Точную позицию вращающегося объекта
- Текущую скорость вращения
- Направление вращения (по часовой или против часовой стрелки)
- Период вращения
- Максимальную и минимальную скорость вращения
- Абсолютное положение вращающегося объекта (если используется абсолютный энкодер)

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Тачскрин» на основе МК»:

1. Какие этапы включает в себя процесс создания алгоритма для системы "Тачскрин" на основе микроконтроллера?

- Определение требований к функциональности и производительности системы "Тачскрин".
- Проектирование аппаратной части системы, включая выбор микроконтроллера и сенсорного экрана.

- Разработка алгоритмов обработки входных данных от сенсорного экрана, включая определение координат касания и распознавание жестов.
- Реализация алгоритмов на микроконтроллере.
- Тестирование и отладка системы "Тачскрин".
- Оптимизация алгоритмов обработки данных для повышения производительности и энергоэффективности.
- Внедрение алгоритмов в конечное устройство с сенсорным экраном.

2. Какие технологии и инструменты используются при разработке программы для системы "Тачскрин" на базе микроконтроллера?

- Язык программирования, поддерживаемый микроконтроллером, например, С или C++.
- Различные интегрированные среды разработки (IDE), такие как Arduino IDE или MPLAB X.
- Библиотеки и фреймворки, предназначенные для работы с сенсорными экранами и обработки входных данных от них.
- Средства отладки и профилирования для выявления и исправления ошибок и улучшения производительности.
- Различные симуляторы и эмуляторы для проверки работы программного обеспечения без фактического наличия микроконтроллера.

3. Какие особенности нужно учитывать при выборе алгоритма для обработки сенсорного ввода в системе "Тачскрин"?

- Точность и стабильность определения координат касания.
- Скорость обработки входных данных для плавной и отзывчивой реакции сенсорного экрана на действия пользователя.
- Учет многокасаний и распознавание жестов для поддержки различных сценариев использования.
- Оптимальное использование ресурсов микроконтроллера, таких как память и процессорное время.

- Адаптация алгоритма к особенностям конкретного сенсорного экрана и его техническим характеристикам.
- Устойчивость к помехам и шумам, которые могут возникать при работе с сенсорным экраном.

4. Какие методы и подходы рекомендуется использовать для оптимизации алгоритма обработки входных данных от сенсорного экрана?

- Использование аппаратных возможностей микроконтроллера, таких как аппаратный ускоритель для обработки сенсорного ввода.
- Разделение обработки данных на параллельные задачи для повышения производительности.
- Уменьшение количества вычислений и операций для снижения нагрузки на микроконтроллер.
- Оптимизация алгоритмов для более эффективного использования доступных ресурсов.
- Применение алгоритмов компрессии данных для сокращения объема информации, передаваемой от сенсорного экрана.
- Использование кэширования результатов вычислений для сокращения времени обработки повторяющихся операций.

5. Какие аспекты безопасности и надежности должны быть учтены при разработке программы для системы "Тачскрин" на микроконтроллере?

- Защита от несанкционированного доступа и изменения программного обеспечения.
- Соответствие стандартам и протоколам безопасности, таким как SSL/TLS для защищенной связи.
- Отказоустойчивость системы и восстановление после сбоев или ошибок.
- Тестирование и отладка программы для выявления и устранения уязвимостей и ошибок.

- Резервное копирование и обновление программного обеспечения системы для обеспечения актуальности и безопасности.
- Соответствие требованиям и рекомендациям по безопасности для конкретного применения системы "Тачскрин".

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Мультиметр» на основе МК»:

1. Каковы основные шаги при создании алгоритма для системы "Мультиметр" на основе микроконтроллера?

- Определение требований и функциональности мультиметра.
- Разработка блок-схемы алгоритма измерения и обработки данных.
- Написание программного кода на языке программирования, поддерживаемом микроконтроллером.
- Тестирование и отладка алгоритма на микроконтроллере.
- Оптимизация и улучшение алгоритма при необходимости.

2. Какие параметры и функции необходимо учесть при разработке программы для мультиметра на основе микроконтроллера?

- Измерение напряжения, тока, сопротивления и других электрических величин.
- Обработка измеренных данных (расчет среднего, минимального, максимального значения и др.).
- Управление дисплеем для отображения результатов измерений.
- Управление кнопками и интерфейсом пользователя.
- Обработка ошибок и исключительных ситуаций.
- Реализация интерфейса для связи с компьютером или другими устройствами.

3. Какие особенности аппаратной составляющей системы "Мультиметр" необходимо учитывать при создании алгоритма?

- Особенности работы и возможности микроконтроллера.
- Калибровка и компенсация погрешностей измерительных каналов.
- Взаимодействие с различными типами датчиков и измерительных преобразователей.
- Безопасность пользователя при работе с высокими напряжениями или токами.

4. Какие методы и инструменты используются для тестирования и отладки алгоритма для мультиметра на основе микроконтроллера?

- Симуляция работы мультиметра на компьютере с использованием специализированного программного обеспечения.
- Использование эмуляторов и отладочных платформ для микроконтроллера.
- Вывод отладочной информации на дисплей или порт компьютера.
- Анализ и сравнение результатов измерений с эталонными значениями.
- Использование протоколов связи для передачи данных с микроконтроллера на компьютер.

5. Каким образом происходит интеграция алгоритма с программой для системы "Мультиметр" на основе микроконтроллера?

- Включение функций и вызов необходимых процедур алгоритма в программный код мультиметра.
- Установка соответствующих прерываний или таймеров для активации и вызова алгоритма измерения по времени или по запросу пользователя.
- Интеграция алгоритма в цикл обработки данных мультиметра и взаимодействия с пользователем.
- Тестирование интеграции алгоритма с программой для обеспечения правильной работы и соответствия требованиям.

Контрольные вопросы по теме «Создание алгоритма и программы для системы «UART с PC» на основе МК»:

1. Каковы основные этапы создания алгоритма и программы для системы «UART с PC» на основе микроконтроллера?

- а. Определение требований и функциональности системы «UART с PC».
- б. Проектирование алгоритма обмена данными между микроконтроллером и ПК через UART.
- с. Написание и отладка программного кода для микроконтроллера, реализующего алгоритм обмена данными.
- д. Тестирование и оптимизация программы на микроконтроллере.
- е. Подключение и настройка программного обеспечения на ПК для взаимодействия с микроконтроллером посредством UART.

2. Какие аспекты необходимо учитывать при разработке алгоритма для взаимодействия системы «UART с PC» через микроконтроллер?

- а. Определение протокола обмена данными между микроконтроллером и ПК.
- б. Управление потоком данных - контроль скорости передачи данных и обработка прерываний.
- с. Обработка ошибок при передаче данных.
- д. Буферизация данных для оптимизации производительности системы.
- е. Разработка интерфейса для обеспечения управления и конфигурации системы через ПК.

3. Какой язык программирования наиболее подходит для разработки программы, обеспечивающей коммуникацию через UART между микроконтроллером и PC?

наиболее подходящим языком программирования может быть C или C++.

Эти языки широко используются для программирования микроконтроллеров

и обладают достаточной гибкостью и эффективностью для работы с данными через UART.

4. Какие возможные проблемы могут возникнуть при создании алгоритма и программы для системы «UART с PC» на основе микроконтроллера, и как их можно предотвратить?

- a. Ошибки в алгоритме передачи данных.
- b. Проблемы синхронизации между микроконтроллером и ПК.
- c. Неправильная обработка прерываний и ошибок при передаче данных.
- d. Недостаточная производительность системы при большом объеме данных.
- e. Неполадки или несовместимость оборудования.

5. Какие технические характеристики микроконтроллера нужно учитывать при разработке программы для обмена данными через UART с PC?

- a. Наличие и характеристики UART-интерфейса (скорость передачи данных, количество доступных портов UART).
- b. Объем доступной памяти для буферизации и обработки данных.
- c. Возможности микроконтроллера по обработке прерываний и ошибок при передаче данных.
- d. Количество и тип доступных аппаратных и программных последовательных портов.
- e. Совместимость микроконтроллера с программным обеспечением на ПК для взаимодействия посредством UART.

Контрольные вопросы по теме «Создание алгоритма и программы для системы «LAN с PC» на основе МК»:

1. Какие шаги нужно выполнить для создания алгоритма для системы "LAN с PC" на базе микроконтроллера?

- Определение целей и требований системы.
- Анализ возможностей микроконтроллера и выбор необходимых компонентов.
- Проектирование сетевого протокола и коммуникационного интерфейса.
- Разработка алгоритма обработки и передачи данных.
- Написание и отладка программного кода для микроконтроллера.
- Тестирование и оптимизация алгоритма.

2. Какие основные функции должен выполнять алгоритм для системы "LAN с PC"?

- Установление соединения с компьютером по сети.
- Передача данных между микроконтроллером и компьютером.
- Обработка и анализ полученных данных.
- Управление периферийными устройствами на основе принятых команд.

3. Какое программное обеспечение и языки программирования можно использовать для написания программы для системы "LAN с PC" на основе микроконтроллера?

- Arduino IDE с использованием языка C/C++ для платформы Arduino.
- MPLAB X IDE с использованием языка C/C++ для микроконтроллеров PIC.
- PlatformIO с использованием языков C/C++ для различных платформ микроконтроллеров.

4. Какие аппаратные компоненты необходимы для реализации системы "LAN с PC" на базе микроконтроллера?

- Микроконтроллер со встроенным Ethernet-контроллером или с возможностью подключения Ethernet-модуля.
- RJ45 разъем для подключения сетевого кабеля.

- Дополнительные периферийные устройства (например, сенсоры или актуаторы), если необходимо управлять внешними устройствами.

5. Как можно обеспечить безопасность данных при работе системы "LAN с РС" на основе микроконтроллера?

- Использование сетевых протоколов и шифрования данных для безопасной передачи информации.

- Установка паролей и других методов аутентификации для защиты от не-санкционированного доступа к системе.

- Регулярное обновление программного обеспечения микроконтроллера для исправления возможных уязвимостей.

- Физическая защита микроконтроллера и сетевого оборудования от не-санкционированного доступа.

Контрольные вопросы по теме «Создание алгоритма и программы для системы «CAN» на основе МК»:

1. Каковы основные шаги при создании алгоритма для системы "CAN" на основе микроконтроллера?

- Определение функциональных требований системы и анализ возможных вариантов решения;

- Выбор подходящего микроконтроллера с поддержкой "CAN" интерфейса;

- Разработка алгоритма обмена данными по протоколу "CAN";

- Разработка и реализация программного кода для микроконтроллера, включающего обработку и передачу сообщений по "CAN" интерфейсу;

- Тестирование и отладка разработанного алгоритма и программы;

- Оптимизация и оптимизация производительности системы "CAN".

2. Какие специфические требования нужно учитывать при разработке программы для системы "CAN" на основе микроконтроллера?

- Поддержка протокола "CAN" и возможность работы с сообщениями по этому протоколу;
- Управление скоростью передачи данных по "CAN" интерфейсу;
- Обработка ошибок передачи данных;
- Реализация механизмов приоритета доступа к "CAN" шине;
- Обеспечение синхронизации и временной согласованности передачи данных.

3. Какие инструменты и языки программирования наиболее подходят для создания алгоритма и программы для "CAN" системы на основе микроконтроллера?

- Язык программирования C или C++;
- Среды разработки, такие как Keil, IAR Embedded Workbench или STM32CubeIDE;
- Утилиты для моделирования и анализа системы "CAN", такие как CANalyzer или CANoe.

4. Какие возможные проблемы могут возникнуть при разработке алгоритма и программы для "CAN" системы на основе микроконтроллера, и какие стратегии и методы их решения существуют?

- Проблемы синхронизации передачи данных между разными узлами системы "CAN" - использование правильных алгоритмов и технических решений для обеспечения согласованности данных;
- Проблемы с обнаружением и обработкой ошибок передачи данных по "CAN" интерфейсу - использование механизмов контроля ошибок и алгоритмов обработки ошибок;
- Проблемы с производительностью системы "CAN" при большом объеме и высокой скорости передачи данных - оптимизация алгоритмов и программы, использование аппаратного ускорения, улучшение архитектуры системы.

5. Какие основные функции и возможности должны быть включены в алгоритм и программу для "CAN" системы на основе микроконтроллера, чтобы обеспечить надежную и эффективную работу системы?

- Инициализация и настройка "CAN" интерфейса и его параметров;
- Прием и обработку сообщений от других узлов системы "CAN";
- Формирование и передача сообщений другим узлам системы "CAN";
- Обработка ошибок передачи и контроль целостности данных;
- Управление и синхронизация доступа к "CAN" шине между разными узлами системы;
- Мониторинг и отладка работы системы "CAN" через специализированные инструменты и интерфейсы.

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Электропривод» на основе МК»:

1. Каковы основные шаги при создании алгоритма для системы "Электропривод" на основе микроконтроллера?

- Определение требований и функциональности системы "Электропривод".
 - Разработка структуры алгоритма, который будет управлять электроприводом.
 - Проектирование алгоритма с учетом различных условий и случаев работы системы.
 - Реализация алгоритма на языке программирования, совместимом с микроконтроллером.
 - Тестирование и отладка алгоритма для проверки его правильной работы.

2. Какие критерии необходимо учитывать при разработке программы для системы "Электропривод" на основе микроконтроллера?

- Эффективность и быстродействие программы.
- Надежность и безопасность работы системы.

- Удобство и простота использования для пользователя.
- Гибкость и возможность обновления программы.
- Совместимость с микроконтроллером и его техническими характеристиками.

3. Какие технические характеристики микроконтроллера необходимо учитывать при создании программы для системы "Электропривод"?

- Вычислительную мощность и скорость работы.
- Объем оперативной и постоянной памяти, необходимый для работы программы.
- Наличие аппаратных модулей и интерфейсов для управления электроприводом.
- Поддержку нужных коммуникационных протоколов (например, UART, SPI, I2C).
- Входы-выходы для взаимодействия с внешними устройствами.

4. Какие инструменты и языки программирования можно использовать для создания алгоритма и программы для системы "Электропривод" на основе микроконтроллера?

- Интегрированные среды разработки (IDE) для конкретного микроконтроллера, например, Arduino IDE, MPLAB X, Keil uVision.
- Языки программирования, такие как C, C++, ассемблер (для оптимизации производительности).
- Библиотеки и фреймворки, специально разработанные для работы с микроконтроллером и электроприводом, например, Arduino библиотеки.

5. Какие этапы включает в себя процесс тестирования созданного алгоритма и программы для системы "Электропривод" на основе микроконтроллера?

- Модульное тестирование алгоритма, проверка корректности работы каждой отдельной части программы.

- Интеграционное тестирование, проверка взаимодействия различных модулей и компонентов программы.
- Системное тестирование, проверка работы системы "Электропривод" в различных реальных условиях.
- Проверка на соответствие требованиям и функциональности системы.
- Отладка и исправление ошибок, выявленных во время тестирования.

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Нагреватель» на основе МК»:

1. Какой алгоритм нужно разработать для системы "Нагреватель" на основе микроконтроллера?

Для системы "Нагреватель" на основе микроконтроллера нужно разработать алгоритм управления нагревом. Алгоритм должен определять, когда включать и выключать нагреватель, какую температуру поддерживать и какие дополнительные функции предусмотреть.

2. Какие этапы включает в себя создание программы для системы "Нагреватель" на основе микроконтроллера?

- Определение требований к функциональности и возможностям системы;
- Разработка алгоритма управления нагревом;
- Выбор микроконтроллера и необходимых компонентов;
- Написание программного кода на языке программирования, соответствующем микроконтроллеру;
- Тестирование программы и отладка ошибок;
- Внедрение программы в микроконтроллер и проведение окончательного тестирования.

3. Какие основные задачи должен выполнять алгоритм для системы "Нагреватель" на основе микроконтроллера?

- Регулирование температуры в зависимости от заданных параметров;
- Определение требуемого времени работы нагревателя для достижения заданной температуры;
- Управление включением и выключением нагревателя;
- Обработка ошибок и предотвращение перегрева или недостаточного нагрева.

4. Какие технические характеристики микроконтроллера нужно учитывать при создании программы для системы "Нагреватель"?

- Вычислительная мощность и скорость работы;
- Наличие аналоговых и цифровых входов/выходов для подключения датчиков и управления нагревателем;
- Поддержка необходимых коммуникационных интерфейсов для связи с другими устройствами или сетями;
- Возможность программной настройки и обновления;
- Интерфейс программирования для разработки и отладки.

5. Какие методы и средства тестирования программы для системы "Нагреватель" на основе микроконтроллера будут использоваться?

- Модельное тестирование с использованием эмуляторов микроконтроллера;
- Юнит-тестирование отдельных функций программы;
- Имитация работы с помощью специальных программных средств или симуляторов;
- Функциональное тестирование системы в реальных условиях с использованием реального нагревателя и датчиков;
- Тестирование нагрузки и проверка стабильности работы программы в различных условиях

Контрольные вопросы по теме «Создание алгоритма и программы для системы «Матобработка данных (DSP)» на основе МК»:

1) Какая роль требуется от МК в алгоритме системы «Матобработка данных (DSP)»?

В алгоритме системы «Матобработка данных (DSP)» требуется от МК роль центрального процессора, который выполняет вычисления и управляет операциями обработки данных.

2) Какие типы данных можно обрабатывать в алгоритме системы «Матобработка данных (DSP)» на основе МК?

В алгоритме системы «Матобработка данных (DSP)» на основе МК можно обрабатывать различные типы данных, включая аналоговые и цифровые сигналы, аудио и видео данные, изображения и другие форматы данных, используемые в области обработки сигналов и данных.

3) Какие преимущества имеет использование МК для системы «Матобработка данных (DSP)»?

- Высокая производительность: МК может обеспечить высокую скорость обработки данных и выполнение сложных алгоритмов.
- Гибкость: МК можно программировать для различных алгоритмов обработки данных и легко вносить изменения в алгоритмы по мере необходимости.
- Низкая задержка: МК обеспечивает быстрый доступ к данным и минимальную задержку при обработке, что особенно важно при работе с реальным временем и потоковыми данными.
- Экономическая эффективность: МК являются относительно недорогими и доступными для использования в системах обработки данных, что позволяет снизить стоимость реализации системы «Матобработка данных (DSP)».

4) Каким образом алгоритм в системе «Матобработка данных (DSP)» на основе МК может обрабатывать потоковые данные?

Алгоритм в системе «Матобработка данных (DSP)» на основе МК может обрабатывать потоковые данные путем создания буферов для приема и передачи данных, использования прерываний для обработки данных в реальном времени и оптимизации алгоритмов для обработки данных в пакетном режиме.

5) Какие компоненты алгоритма необходимо использовать для обработки данных в системе «Матобработка данных (DSP)» на основе МК?

- АЦП (аналого-цифровой преобразователь) для преобразования аналоговых сигналов в цифровой формат.
- ЦП-ЦП приемник и передатчик для связи с другими устройствами или сетями.
- Буферы для приема и передачи данных.
- Алгоритмы обработки сигналов и данных, которые выполняются МК.
- Оперативная память (ОЗУ) для временного хранения данных и промежуточных результатов.
- Устройства вывода данных, такие как монитор, акустическая система или другие устройства для визуализации или воспроизведения обработанных данных.

Итоговый тест по теме «Автоматизация процессов на основе систем с микроконтроллерами STM32 или аналогов»

1. Какой микроконтроллер наиболее часто используется для автоматизации процессов?

- a) STM32
- b) Arduino
- c) Raspberry Pi
- d) PIC

2. Какие языки программирования поддерживает микроконтроллер STM32?
- a) C
 - b) C++
 - c) Python
 - d) Все вышеперечисленные
3. Что такое GPIO на микроконтроллере STM32?
- a) Графический интерфейс
 - b) Вход/выход общего назначения
 - c) Внутренняя память
 - d) Генератор частоты
4. Какое преимущество имеет использование микроконтроллера STM32 вместо Raspberry Pi для автоматизации процессов?
- a) Более высокая производительность
 - b) Большое количество GPIO-пинов
 - c) Ниже стоимость
 - d) Встроенные варианты коммуникационных интерфейсов
5. Какого типа данные можно использовать для программирования микроконтроллера STM32?
- a) Целые числа
 - b) Символьные строки
 - c) Действительные числа
 - d) Все вышеперечисленные
6. Какое преимущество имеет использование аналогового входа на микроконтроллере STM32?
- a) Измерение аналоговых величин
 - b) Передача аналоговых данных

- c) Для подключения цифровых датчиков
- d) Нет преимущества

7. Какая шина связи используется в микроконтроллерах STM32 для обмена данными?

- a) I2C
- b) SPI
- c) UART
- d) Все вышеперечисленные

8. Для чего используются таймеры на микроконтроллере STM32?

- a) Для отсчета времени
- b) Для генерации прерываний
- c) Для генерации PWM-сигналов
- d) Все вышеперечисленные

9. Какова стандартная частота работы микроконтроллера STM32?

- a) 1 MHz
- b) 8 MHz
- c) 16 MHz
- d) 72 MHz

10. Какие периферийные устройства можно подключить к микроконтроллеру STM32?

- a) LED-индикаторы
- b) Клавиатура
- c) Жидкокристаллический дисплей
- d) Все вышеперечисленные

11. Что такое прерывание на микроконтроллере STM32?

- a) Ошибка в программе
- b) Сигнал о необходимости прерывания текущей операции
- c) Остановка работы микроконтроллера
- d) Потеря соединения со внешним устройством

12. Какую библиотеку можно использовать для программирования микроконтроллера STM32?

- a) Standard Peripheral Library (SPL)
- b) CMSIS
- c) HAL (Hardware Abstraction Layer)
- d) Все вышеперечисленные

13. Какова типичная стоимость микроконтроллера STM32?

- a) 1-5 долларов США
- б) 10-20 долларов США
- в) 50-100 долларов США

14. Какие ресурсы доступны для изучения программирования на микроконтроллере STM32?

- a) Онлайн курсы
- б) Учебники
- в) Вебинары

15. Какое напряжение требуется для питания микроконтроллера STM32?

- a) 5 Вольт
- б) 3.3 Вольта
- в) 12 Вольт

16. Какая среда разработки может использоваться для программирования микроконтроллера STM32?

- a) Keil MDK

б) Microsoft Excel

в) Adobe Illustrator

17.Какой язык описание аппаратуры (HDL) может использоваться для разработки на микроконтроллере STM32?

а) VHDL

б) Verilog

в) Assembly

| | | | | | | | | | | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|---|----|----|----|----|-----|----|----|-----|
| Вопросы | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| Ответы | б | д | б | д | д | а | д | д | д | д | б | д | а | абв | б | а | абв |