

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ САРАТОВСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ САРАТОВСКОЙ ОБЛАСТИ
«МАРКСОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»**

**КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ
ОП.08 МИКРОПРОЦЕССОРНЫЕ СИСТЕМЫ**
специальность: 11.02.16 Монтаж, техническое обслуживание и
ремонт электронных приборов и устройств

г. Маркс, 2024 год



КОС для общепрофессиональной дисциплины ОП.08 Микропроцессорные системы разработан в соответствии с требованиями ФГОС СПО по специальности 11.02.16 Монтаж, техническое обслуживание и ремонт электронных приборов и устройств, утвержденного приказом Минпросвещения России от 04.10.2021г. №691.

РАССМОТРЕНО на заседании цикловой методической комиссии технического профиля
Протокол № 9, дата « 15/ » мая 2024 г.
Председатель /В. И. Гриднев/

СОГЛАСОВАНО с Методическим советом ГАПОУ СО «Марковский политехнический колледж»
Протокол № 10 от « 17 » мая 2024 г.
Председатель /Гостева И.Ю./

Составитель: Хлебникова Г.Н., преподаватель высшей квалификационной категории ГАПОУ СО «Марковский политехнический колледж»

Рецензенты:

Внутренний: Гриднев В. И., преподаватель высшей квалификационной категории ГАПОУ СО «Марковский политехнический колледж»

Внешний: Коваль Людмила Валентиновна, преподаватель Марковского сельскохозяйственного техникума.

СОДЕРЖАНИЕ

1.	ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ	4
2.	КРИТЕРИИ ОЦЕНИВАНИЯ УСПЕВАЕМОСТИ	10
3.	КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ	17
4.	КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ	44

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

1.1 Область применения фонда оценочных средств

Фонд оценочных средств (далее - ФОС) по учебной дисциплине представляет собой комплект методических и контрольных измерительных материалов, оценочных средств, предназначенных для аттестации обучающихся на соответствие их персональных достижений поэтапным требованиям программы подготовки специалистов среднего звена по специальности (текущий контроль успеваемости и промежуточная аттестация).

Фонд оценочных средств по дисциплине ОП.08 Микропроцессорные системы разработан согласно требованиям ФГОС СПО и является неотъемлемой частью реализации программы подготовки специалистов среднего звена по специальности 11.02.16 Монтаж, техническое обслуживание и ремонт электронных приборов и устройств.

Целью фонда оценочных средств является установление соответствия уровня подготовки обучающихся требованиям ФГОС СПО по специальности 11.02.16 Монтаж, техническое обслуживание и ремонт электронных приборов и устройств.

Задачи ФОС:

- контроль и управление процессом приобретения обучающимися необходимых знаний, умений, практического опыта и освоения компетенций, определенных ФГОС СПО;

- контроль и управление достижением целей программы, определенных как набор общих и профессиональных компетенций;

- оценка достижений обучающихся в процессе обучения с выделением положительных / отрицательных результатов и планирование предупреждающих / корректирующих мероприятий;

- обеспечение соответствия результатов обучения задачам будущей профессиональной деятельности через совершенствование традиционных и внедрение инновационных методов обучения;

- достижение такого уровня контроля и управления качеством образования, который обеспечил бы признание квалификаций выпускников работодателями отрасли.

Фонд оценочных средств включает в себя контрольно-оценочные средства (задания и критерии их оценки, а также описания форм и процедур) для проведения текущего контроля успеваемости и промежуточной аттестации (определения качества освоения обучающимися результатов освоения учебной дисциплины (умений, знаний, практического опыта, ПК и ОК).

ФОС обеспечивает поэтапную (текущий контроль) и интегральную (промежуточная аттестация) оценку умений и знаний обучающихся, приобретаемых при обучении по учебной дисциплине, направленных на

формирование компетенций.

1.1.1. Перечень общих компетенций

ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;

ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;

ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях;

ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях;

ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.

1.1.2. Перечень профессиональных компетенций

ПК 1.1. Осуществлять сборку, монтаж и демонтаж электронных приборов и устройств в соответствии с требованиями технической документации

ПК 1.2. Осуществлять сборку, монтаж и демонтаж электронных приборов и устройств и их настройку и регулировку в соответствии с требованиями технической документации и с учетом требований технических условий.

ПК 2.1. Производить диагностику работоспособности электронных приборов и устройств средней сложности

ПК 2.2. Осуществлять диагностику аналоговых, импульсных, цифровых и со встроенными микропроцессорными системами устройств средней сложности для выявления и устранения неисправностей и дефектов

ПК 2.3. Выполнять техническое обслуживание электронных приборов и устройств в соответствии с регламентом и правилами эксплуатации

ПК 3.1. Разрабатывать структурные, функциональные и принципиальные схемы простейших электронных приборов и устройств.

ПК 3.2. Разрабатывать проектно-конструкторскую документацию печатных узлов электронных приборов и устройств и микросборок средней сложности

Формой промежуточной аттестации по учебной дисциплине - **экзамен.**

1.2 Результаты освоения учебной дисциплины ОП.08 Микропроцессорные системы, подлежащие проверке.

В результате аттестации по учебной дисциплине осуществляется комплексная проверка следующих умений и знаний:

Коды и наименования результатов обучения (умения и знания)	Показатели оценки результата	Формы и методы контроля и оценки результатов обучения
студент должен уметь:		
<p>У 1 - читать электрические схемы, построенные на микросхемах микроконтроллеров ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>- проведение исследования работы цифровых устройств и проверку их работоспособности с использованием программ автоматизированного анализа электронных схем</p>	<p>Индивидуальный и фронтальный опрос. Оценка выполненной практической работы. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>
<p>У 2 - программировать встраиваемые системы: AVR- микро контроллеры с помощью специализированных языков; ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>- составление простейших программ и экспериментальное определение влияния команд на состояние микропроцессорной системы</p>	<p>Индивидуальный и фронтальный опрос. Оценка выполненной практической работы. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>
<p>У 3 - проводить программно-аппаратную отладку встраиваемых систем (микропроцессорных систем) ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>-анализировать ход выполнения программы, используя аппаратные и программные средства отладки</p>	<p>Индивидуальный и фронтальный опрос. Оценка выполненной практической работы. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>

<p>У 4- программировать микроконтроллеры ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 , ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>- составление простейших программ и экспериментальное определение влияния команд на состояние микропроцессорной системы</p>	<p>Индивидуальный и фронтальный опрос. Оценка выполненной практической работы. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен -экзамен</p>
<p>студент должен знать:</p>		
<p>З 1- типовые узлы и устройства микро-процессорных систем, ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 , ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>- описание принципа работы типовых узлов и устройств микропроцессорных систем, обоснование разделения по определенным признакам</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>
<p>З 2 - классификацию устройств памяти микроконтроллеров; ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 , ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>- понимание особенностей различных типов различных групп устройств памяти, обоснование разделения по определенным признакам</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>
<p>З 3 - архитектуру микропроцессоров и микроконтроллеров; ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 , ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>- понимание особенностей различных типов архитектур</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>

<p>З 4 - способы алгоритмизации и программирования микроконтроллеров ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>-понимание сущности этапов выбора методов решения задачи и разработки алгоритма реализации</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>
<p>З 5 -принципы взаимодействия аппаратного и программного обеспечения в работе микроконтроллеров ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>-понимание сущности взаимодействия аппаратного и программного обеспечения в работе микроконтроллеров</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>
<p>З 6- особенности архитектуры и функционирования, циклы выполнения команд микроконтроллеров ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>-понимание особенностей архитектуры, функционирования узлов в циклах выполнения команд микроконтроллеров</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>
<p>З 7- систему команд, способы адресации микроконтроллеров ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>-определение назначения и свойств основных групп команд и особенности программирования микроконтроллеров</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>

<p>3 8- характерные особенности, принципы построения, основные типы, процессоров DSP ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>- описание принципов построения, знание особенностей и основных типов процессоров, их роль в реализации методов цифровой обработки сигналов</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>
<p>3 9- Архитектуру DSP семейства ADSP-21xx ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>-понимание особенностей архитектуры процессоров семейства ADSP-21xx</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>
<p>3 10- принципы программирования микропроцессорных систем на ассемблере ОК 1, ОК 2, ОК 3, ОК 7, ОК 9, ПК 1.1., ПК 1.2 ,ПК 2.1- 2.3, ПК 3.1., ПК 3.2.</p>	<p>-определение назначения и свойств основных групп команд микропроцессорных устройств и особенности программирования на языке ассемблер для современных микропроцессорных систем</p>	<p>Индивидуальный и фронтальный опрос. Тестирование с применением проблемных заданий. Устный и письменный контроль. Промежуточная аттестация в форме итогового контроля - экзамен</p>

2. КРИТЕРИИ ОЦЕНИВАНИЯ УСПЕВАЕМОСТИ

Контроль и оценка результатов освоения учебной дисциплины ОП.08 Микропроцессорные системы осуществляется преподавателем в процессе:

- проведения устного или письменного опроса по теме, разделу; круглого стола, деловой игры, семинара и др.
- выполнения и защиты лабораторных и практических работ;
- тестирования по отдельным темам и разделам;
- анализ выполнения типового задания и т.д.

Устный или письменный опрос проводится на практических занятиях и затрагивает как тематику предшествующих занятий, так и лекционный материал и позволяет выяснить объем знаний студента по определенной теме, разделу, проблеме. Устный опрос в форме собеседования - специальная беседа преподавателя с обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п.

Типовое задание - стандартные задания, позволяющие проверить умение решать как учебные, так и профессиональные задачи. Содержание заданий должно максимально соответствовать видам профессиональной деятельности.

Различают разноуровневые задачи и задания:

а) ознакомительного, позволяющие оценивать и диагностировать знание фактического материала (базовые понятия, алгоритмы, факты) и умение правильно использовать специальные термины и понятия, узнавание объектов изучения в рамках определенного раздела дисциплины;

б) репродуктивного уровня, позволяющие оценивать и диагностировать умения синтезировать, анализировать, обобщать фактический и теоретический материал с формулированием конкретных выводов, установлением причинно-следственных связей;

в) продуктивного уровня, позволяющие оценивать и диагностировать умения, интегрировать знания различных областей, аргументировать собственную точку зрения, выполнять проблемные задания.

Тестирование представляет собой систему стандартизированных заданий, позволяющую автоматизировать процедуру измерения уровня знаний и умений обучающегося, направлено на проверку владения терминологическим аппаратом и конкретными знаниями по дисциплине. Тестирование по теме, разделу занимает часть учебного занятия (10-30 минут), правильность решения разбирается на том же или следующем занятии; частота тестирования определяется преподавателем.

Практические занятия проводятся в часы, выделенные учебным планом для отработки практических навыков освоения компетенциями, и предполагают аттестацию всех обучающихся за каждое занятие.

В ходе практического занятия обучающиеся приобретают умения, предусмотренные рабочей программой дисциплины, учатся использовать формулы, и применять различные методики расчета, анализировать полученные результаты и делать выводы, опираясь на теоретические знания.

Содержание, этапы проведения конкретного практического занятия или лабораторной работы, критерии оценки представлены в методических указаниях

по выполнению практических работ.

Отчет по практической работе представляется в печатном виде в формате, предусмотренном шаблоном отчета по практической, лабораторной работе. Защита отчета проходит в форме доклада обучающегося по выполненной работе и ответов на вопросы преподавателя.

В случае невыполнения практических заданий в процессе обучения, их необходимо «отработать». Вид заданий, которые необходимо выполнить для ликвидации задолженности определяется в индивидуальном порядке, с учетом причин невыполнения.

Форма проведения текущего контроля успеваемости и промежуточной аттестации для обучающихся инвалидов и лиц с ограниченными возможностями здоровья выбирается с учетом индивидуальных психофизических особенностей (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.). При необходимости обучающимся инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на дифференцированном зачете.

2.1. Критерии оценивания теоретических знаний

Требования к устным ответам

Результатом проверки уровня усвоения учебного материала является отметка. При оценке знаний обучающихся предполагается обращать внимание на правильность, осознанность, логичность и доказательность в изложении материала, точность использования терминологии, самостоятельность ответа. Оценка знаний предполагает учёт индивидуальных особенностей обучающихся, дифференцированный подход к организации работы.

Критерии оценки устного ответа:

Оценка	Условия, при которых выставляется оценка
Оценка 5 («отлично»)	полно раскрыл содержание материала в объеме, предусмотренном программой и учебником; изложил материал грамотным языком в определенной логической последовательности, точно используя математическую терминологию и символику; правильно выполнил рисунки, чертежи, графики, сопутствующие ответу; показал умение иллюстрировать теоретические положения конкретными примерами, применять их в новой ситуации при выполнении практического задания; продемонстрировал усвоение ранее изученных сопутствующих вопросов, сформированность и устойчивость использованных при ответе умений и навыков; отвечал самостоятельно без наводящих вопросов преподавателя. Возможны одна – две неточности при освещении второстепенных вопросов или в выкладках, которые

	обучающийся легко исправил по замечанию преподавателя.
Оценка 4 («хорошо»)	В изложении допущены небольшие пробелы, не исказившие математического содержания ответа, исправленные по замечанию преподавателя. Допущены ошибки или более двух недочетов при освещении второстепенных вопросов или в выкладках, которые обучающийся легко исправил по замечанию преподавателя.
Оценка 3 («удовлетворительно»)	Неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса и продемонстрированы умения, достаточные для дальнейшего усвоения программного материала. Имелись затруднения или допущены ошибки в определении понятий и использовании терминологии, чертежах, выкладках, исправленные после нескольких наводящих вопросов преподавателя. Обучающийся не справился с применением теории в новой ситуации при выполнении практического задания, но выполнил задания обязательного уровня сложности по данной теме. При проверке теоретического материала выявлена недостаточная сформированность умений и навыков.
Оценка 2 («неудовлетворительно»)	Не раскрыто основное содержание учебного материала; обнаружено незнание или непонимание учеником большей или наиболее важной части учебного материала; допущены ошибки в определении понятий, при использовании математической терминологии, в рисунках, чертежах или графиках, в выкладках, которые не исправлены после нескольких наводящих вопросов преподавателя. Обучающийся обнаружил полное незнание и непонимание изучаемого материала или не смог ответить ни на один из поставленных вопросов по изучаемому материалу.

Требования к оформлению доклада

Доклад предоставляется в распечатанном виде, объёмом 3-5 страниц. Текст доклада должен быть представлен в текстовом редакторе Word, шрифт – Times New Roman 14, межстрочный интервал – 1.5 (полуторный). Поля: верхнее - 2, нижнее - 2, левое- 3, правое - 1,5.

Доклад должен включать в себя: введение, основную часть, заключение, список литературы (не менее 5 источников).

Критерии оценки доклада:

Оценка	Условия, при которых выставляется оценка
Оценка 5 («отлично»)	материал изложен в определенной логической последовательности. Тема доклада раскрыта полностью.
Оценка 4 («хорошо»)	тема раскрыта, но при этом допущены не существенные ошибки, исправленные по требованию преподавателя.
Оценка 3 («удовлетворительно»)	тема раскрыта не полностью, допущена существенная ошибка.
Оценка 2 («неудовлетворительно»)	содержании доклада не раскрывает рассматриваемую тему, обнаружено не понимание основного содержания учебного материала

Доклад может быть представлен как доклад-презентация. Необходимо представить 5-7 слайдов. Время доклада -5 минут. Критерии оценки доклада такие же. Дополнительно оценивается презентация.

Оформление слайдов	Параметры
Стиль	Соблюдать единого стиля оформления.
Фон	Фон не должен быть слишком темным или ярким, чтобы не отвлекать внимания от содержания слайдов.
Использование цвета	Слайд не должен содержать более трех цветов. Фон и текст должны быть оформлены контрастными цветами.
Анимационные эффекты	При оформлении слайда использовать возможности анимации. Анимационные эффекты не должны отвлекать внимание от содержания слайдов.
Представление информации	Параметры
Содержание информации	Слайд должен содержать минимум информации. Информация должна быть изложена доступным языком. Содержание текста должно точно отражать этапы выполненной работы. Текст должен быть расположен на слайде так, чтобы его удобно было читать. В содержании текста должны быть ответы на проблемные вопросы. Текст должен соответствовать теме презентации.
Расположение информации на странице	Предпочтительно горизонтальное расположение информации. Наиболее важная информация должна располагаться в центре. Надпись должна располагаться под картинкой.

Размер шрифта	Для заголовка – не менее 24. Для информации не менее – 18. Лучше использовать один тип шрифта. Важную информацию лучше выделять жирным шрифтом, курсивом, подчеркиванием На слайде не должно быть много текста, оформленного прописными буквами.
Выделения информации	На слайде не должно быть много выделенного текста (заголовки, важная информация).
Объем информации	Слайд не должен содержать большого количества информации. Лучше ключевые пункты располагать по одному на слайде.
Виды слайдов	Для обеспечения разнообразия следует использовать разные виды слайдов: <ul style="list-style-type: none"> – с таблицами – с текстом – с диаграммами

Критерии оценивания презентаций:

Оценка	Условия, при которых выставляется оценка
Оценка 5 («отлично»)	выполненная презентация отвечает всем требованиям критериев
Оценка 4 («хорошо»)	в презентации имеются незначительные нарушения или отсутствуют какие-либо параметры
Оценка 3 («удовлетворительно»)	при оценивании половина критериев отсутствует

Требования к оформлению реферата

Реферат предоставляется в распечатанном виде, объемом 10-15 страниц. Текст реферата должен быть представлен в текстовом редакторе Word, шрифт - TimesNewRoman 14, межстрочный интервал – 1.5 (полуторный), в таблицах возможен межстрочный интервал – 1(одинарный), поля: верхнее - 2, нижнее - 2, левое- -3, правое - 1,5.

Реферат должен включать в себя: содержание, введение, основную часть, заключение, список литературы (не менее 5 источников).

Время на защиту реферата: 5 минут.

Критерии оценивания реферата:

Оценка	Условия, при которых выставляется оценка
Оценка 5 («отлично»)	материал изложен в определенной логической последовательности. Тема реферата раскрыта полностью.
Оценка 4 («хорошо»)	тема реферата раскрыта, при этом допущены не существенные ошибки, исправленные по требованию преподавателя

Оценка 3 («удовлетворительно»)	тема раскрыта не полностью, допущена существенная ошибка
Оценка 2 («неудовлетворительно»)	при защите реферата обнаружено не понимание основного содержания учебного материала

Выполнение тестирования

Критерии оценивания:

Оценка	Условия, при которых выставляется оценка
Оценка 5 («отлично»)	если студент при тестировании дал 85-100% правильных ответов
Оценка 4 («хорошо»)	если студент при тестировании дал 69-84% правильных ответов
Оценка 3 («удовлетворительно»)	если студент при тестировании дал 51-68% правильных ответов
Оценка 2 («неудовлетворительно»)	если студент при тестировании дал менее 50% правильных ответов

2.2. Критерии оценивания практических знаний

Оценка	Критерии оценивания
Оценка 5 («отлично»)	<ul style="list-style-type: none"> - практическая работа выполнена в установленные сроки (при отсутствии уважительных причин для несвоевременного выполнения работы); - все расчеты выполнены в соответствии с методикой и в полном объеме, обозначены единицы измерения всех рассчитываемых показателей; - сделан развернутый вывод по итогам выполненных расчетов; - работа оформлена аккуратно.
Оценка 4 («хорошо»)	<ul style="list-style-type: none"> - практическая работа выполнена в установленные сроки (при отсутствии уважительных причин для несвоевременного выполнения работы); - расчеты выполнены в полном объеме, но были допущены одна - две негрубые ошибки при выполнении математических действий или не обозначены единицы измерения рассчитываемых показателей; - сделан развернутый вывод по итогам выполненных расчетов, однако не все выводы носят аргументированный и доказательный характер; - работа оформлена аккуратно
Оценка 3 («удовлетворительно»)	<ul style="list-style-type: none"> - практическая работа выполнена в неустановленные сроки (при отсутствии уважительных причин для несвоевременного выполнения работы); - расчеты выполнены в полном объеме, но при этом были допущены одна – две грубые или три – четыре негрубые ошибки при выполнении математических действий, не обозначены единицы измерения рассчитываемых показателей или работа оформлена неаккуратно, с большим количеством исправлений;

	<ul style="list-style-type: none">- не сделан развернутый вывод по итогам выполненных расчетов.- работа оформлена неаккуратно.
Оценка 2 («неудовлетворительно»)	<ul style="list-style-type: none">- работа не выполнена;- при выполнении расчетов обучающийся допускает более двух грубых ошибок или более четырех негрубых, не обозначены единицы измерения рассчитываемых показателей или обозначены неправильно;- не сделан вывод по итогам выполненных расчетов. <p>В случае получения оценки «неудовлетворительно» студент обязан выполнить работу заново.</p>

3. КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ

Раздел 1. Микропроцессорные системы. Основные понятия

Тема 1.1. Микропроцессорные системы (МПС). Виды и характеристики.

Устный фронтальный опрос

1. Основные виды МПС и их особенности.
2. Обобщенная структура МПС.
3. Основные характеристики и параметры МПС.
4. Краткая характеристика возможностей и применений микропроцессорных систем

Тема 1.2. Организация функционирования МПС

Письменный опрос

1. Обобщенная структурная схема МПС.
2. Опишите алгоритм работы. Механизмы прерываний.

Письменный опрос проводится во время урока. На письменный опрос отводится 10 мин.

Тема 1.3. Микропроцессоры (МП)

Устный фронтальный опрос

1. Классификация и характеристики МП.
2. Понятие об архитектуре микропроцессора.
3. Основные элементы архитектуры.
4. Поколения МП.

Тема 1.4. Микроконтроллеры (МК). Общие сведения

Устный опрос

1. Классификация. Архитектура.
2. Обобщенная структурная схема микроконтроллера серии AVR
3. Основные элементы структурной схемы. Назначение. Характеристика.
4. Логические основы построения микроконтроллеров.
5. Классификация устройств памяти системы команд.

Тема 1.5. Микроконтроллеры семейства серии AVR

Устный опрос

1. Какие особенности имеет AVR-архитектура?
2. Какая память программ используется в микроконтроллерах с архитектурой AVR?
3. Какой объем памяти программ имеет микроконтроллер ATmega16?
4. Для чего в памяти программ предусматривают раздел начальной загрузки?
5. Как загружается программа пользователя в память программ?
6. Память какого типа используется для хранения данных в микроконтроллерах с AVR-архитектурой?
7. Для чего в структуре микроконтроллера предусматривается память типа EEPROM?
8. Почему память типа EEPROM не может быть использована непосредственно при выполнении инструкций процессорного ядра?
9. Какой объем статической памяти имеет микроконтроллер типа ATmega16?
10. Какой адрес имеет старшая ячейка статической памяти микроконтроллера ATmega16?

Практическая работа №1. Выполнение сравнительного анализа микросхем микроконтроллеров серии AVR

1. Цель работы: Приобретение практических навыков сравнительного анализа микросхем микроконтроллеров серии AVR.

2. Время выполнения работы-2 часа

3. Краткие теоретические сведения

Микроконтроллер (МК) характеризуется:

1) тактовой частотой, определяющей максимальное время выполнения переключения элементов микропроцессорной системы;

2) разрядностью, т.е. максимальным числом одновременно обрабатываемых двоичных разрядов, существенно влияет на быстродействие.

Разрядность МК обозначается m/n/k/ и включает:

m - разрядность внутренних регистров, определяет принадлежность к тому или иному классу процессоров;

n - разрядность шины данных, определяет скорость передачи информации;

k - разрядность шины адреса, определяет размер адресного пространства.

Например, МП i8088 характеризуется значениями m/n/k=16/8/20;

3) архитектурой. Понятие архитектуры МК включает в себя систему команд и способы адресации, возможность совмещения выполнения команд во времени, наличие дополнительных устройств в составе микропроцессора, принципы и режимы его работы.

В процессе развития архитектура МК претерпела существенные

изменения. Первые МК строились по так называемой принстонской архитектуре (архитектуре фон Неймана) в которой память для команд и данных является общей. Эта архитектура имеет свои достоинства – простоту, возможность оперативного перераспределения памяти между областями хранения команд и данных и др. Недостаток – последовательная во времени выборка из памяти команд и данных, передаваемых по одной и той же системной шине, что ограничивает производительность МК. Тем не менее, в силу своих достоинств, принстонская архитектура не только длительное время доминировала в микропроцессорной технике, но и сохранила свои позиции до настоящего времени.

В системах гарвардской архитектуры память команд и память данных разделены, причем каждая из них имеет собственную шину для обмена с процессором. При этом во время передач данных для выполнения текущей команды можно производить выборку и расшифровку следующей, что повышает производительность МК системы. Реализация системы по сравнению с Принстонской усложняется (в системе больше шин), ниже коэффициент использования памяти. Но в системах высокой производительности и внутренних структурах высокопроизводительных МК гарвардская архитектура находит широкое применение.

По другому архитектурному признаку, связанному с характером системы команд, МК делятся на:

- с CISC процессорами
- с RISC процессорами
- с VLIW процессорами
- с ARM процессорами

МК CISC имеют так называемую полную систему команд (ComplexInstructionSetComputer), т. е. большой набор разноформатных команд при использовании многих способов адресации. Архитектура CISC силу многообразия команд (общее число команд составляет 100...200) позволяет применять эффективные методы решения задач, но, в тоже время, усложняет схему процессора и увеличивает его стоимость и в общем случае не обеспечивает максимального быстродействия. Они характеризуются следующими особенностями:

- нефиксированным значением длины команды.
- исполнение операций, таких как, загрузка в память, арифметические действия кодируется в одной инструкции.
- небольшим числом регистров, каждый из которых выполняет строго определенную функцию.

МК RISC имеют сокращенную систему команд (ReduceInstructionSetComputer), из которой исключены редко применяемые команды. Общее число команд находится в пределах 50...100.. Простая архитектура позволяет, как удешевить МК, так и увеличить тактовую частоту.

Характерные особенности МК RISC:

- фиксированная длина машинных инструкций (например, 32 бита) и

простой формат команды.

- одна инструкция выполняет только одну операцию с памятью — чтение или запись. Операции вида «прочитать-изменить-записать» отсутствуют.
- большое количество регистров общего назначения (32 и более).

Форматы команд, в большинстве своем, идентичны (например, все команды имеют длину 4 байта), резко уменьшено число используемых способов адресации. Данные, как правило, обрабатываются только с регистровой или непосредственной адресацией. Значительно увеличено число регистров процессора, что позволяет редко обращаться к внешней памяти, что повышает быстродействие. Идентичность временных циклов выполнения команд позволяет проще организовывать конвейерные методы обработки информации

МК VLIW- МК с системой команд сверхбольшой разрядности. Команда сверхбольшой разрядности состоит из группы команд, которые могут выполняться параллельно. Команды сверхбольшой разрядности формируются специальным компилятором планирования перед выполнением прикладной программы ARM-(Advanced RISC machine) усовершенствованная RISC архитектура, 32-битная микропроцессорная архитектура с сокращённым набором команд (RISC), разрабатываемая британской корпорацией ARM Limited.

AVR — семейство восьмибитных микроконтроллеров фирмы Atmel. Микроконтроллеры AVR имеют **гарвардскую** архитектуру и систему команд, близкую к идеологии **RISC**. Процессор AVR имеет 32 8-битных регистра общего назначения, объединённых в регистровый файл. Система команд микроконтроллеров AVR весьма развита и насчитывает в различных моделях от 90 до 133 различных инструкций. Большинство команд занимает только 1 ячейку памяти (16 бит) и выполняется за 1 такт. Управление периферийными устройствами осуществляется через адресное пространство данных. Для удобства существуют «сокращённые команды» IN/OUT.

Маркировка микроконтроллера - это сгруппированный набор буквы цифр, обозначающих расшифровку производителя чипа, его модели и дополнительных параметров

AT - с данного префикса начинается название микроконтроллера и оно обозначает изготовителя ATMEL.

Слова "tiny", "mega", "xmega" - следует после AT и обозначает семейство микроконтроллеров. Например, маркировка "ATmega" - означает чип от фирмы ATMEL семейства "mega".



В большинстве случаев после АТ(tiny, mega, xmega) следуют цифры 1, 2, 4, 8, 16, 32, которые указывают на размер встроенной Flash-памяти в кБ, необходимой для сохранения рабочей программы.

Следующие цифры после размера Flash-памяти обозначает версию(модификацию).

После цифр может присутствовать буква, обозначающая режим питания микроконтроллера:

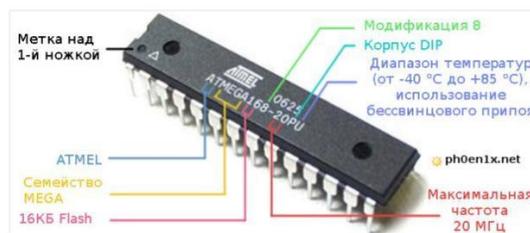
L - работа при пониженном (Low) напряжении питания 2,7 вольт (2,7-5,5В), но при этом максимальная частота может достигать 8 МГц;

V - возможна работа при напряжении питания 1,8 вольт (1,8-5,5В);

U - возможна работа при напряжении питания 0,7 вольт (0,7-5,5В);

P - энергоэкономичная версия с потреблением тока до 100 нА в режиме Power-down, изготовлена по технологии picoPower;

A - специальная версия с заниженным энергопотреблением, напряжение питания 1,8-5,5В. *Если же буквы после цифр нет, то это означает что стандартное питание чипа - 5В, а точнее диапазон 4,5-5,5В.*



Следующая буква обозначает температурные характеристики микросхемы, а также дополнительные свойства:

A - диапазон температур (от -20 °C до +85 °C),

C - коммерческий (Commercial) диапазон температур (от 0 °C до 70 °C)

D - расширенный автомобильный диапазон температур (от -40 °C до +150 °C)

F - расширенный диапазон температур (от -40 °C до +125 °C)

H - промышленный диапазон температур (от -40 °C до +85 °C), с использованием NiPdAu (Nickel-Palladium-Gold)

I - промышленный (Industrial) (от -40 °C до +85 °C)

N - расширенный диапазон температур (от -40 °C до +105 °C), использование бессвинцового припоя

R - упаковка в ленты для автоматизированных систем сборки.

U - промышленный (indUstrial) диапазон температур (от -40 °C до +85 °C), использование бессвинцового припоя

Z - автомобильный диапазон температур (от -40 °C до +125 °C)

Задание:

Выполнить сравнительный анализ микросхем микроконтроллеров серии AVR.

5. Порядок выполнения

5.1. Изучите краткие теоретические сведения

5.2. Получите вариант задания у преподавателя

5.3. Пользуясь справочной литературой, составьте таблицу справочных

данных для заданных ИМС

5.4. Сделать выводы

	ИМС:	ИМС:	ИМС:
Fmax, МГц			
Ucc., В			
размер Flash-памяти в кБ,			
Температурный диапазон			
Тип корпуса			

6. Указания к выполнению отчета

Отчет должен содержать

- тему и цель работы,
- таблицу справочных данных для заданных ИМС
- выводы по проведенным исследованиям
- ответы на контрольные вопросы

7. Контрольные вопросы

1. Что такое архитектура микроконтроллера?
2. Основные характеристики микроконтроллера
3. Преимущества и недостатки прынстонской архитектуры
4. Преимущества и недостатки гарвардской архитектуры
5. Характерные особенности МК RISC:

Раздел 2. Алгоритмизация и программирование микроконтроллеров

Тема 2.1. Языки программирования

Устный опрос.

1. Опишите этап разработки алгоритма и структуры программы;
2. Что такое алгоритм

3. Опишите этап написания исходного текста программы;
4. Опишите этап получения выполняемой программы
5. Что такое транслятор?
6. Что такое компилятор?
7. Особенности ассемблера
8. Опишите этап тестирования и отладки программы
9. Опишите этап получения загрузочной программы
10. Что включает процедура кросс-разработки

Тема 2.2. Трансляция программы

Устный опрос.

1. Транслятор.
2. Трансляция программы и получение файла прошивки для микроконтроллера.
3. Краткий обзор содержимого файла прошивки.
4. Разбор файла описаний и листинга программы.
5. Размещение программы в памяти микроконтроллера.

Тема 2.3. Краткий обзор программаторов

Письменный опрос.

1. Программаторы.
2. Последовательные и параллельные программаторы.
3. Внутрисхемное программирование

Письменный опрос проводится во время урока. На письменный опрос отводится 10 мин.

Тема 2.4. Программирование микроконтроллеров

Устный опрос.

1. Программирование в машинных кодах.
2. Подробный разбор файлов проекта и разбор содержимого файла прошивки.
3. Редактирование кодов команд в файле прошивки
4. Приемы программирования. Этапы программирования.
5. Операции начальной настройки. Операции, составляющие тело цикла.
6. Алгоритм создания программы. Форма записи. Директивы. Операторы.
7. Мастер Программ и его свойства.

8. Настройка портов.
9. Работа программа на языке Си. Описание. Комментарии.

Тема 2.5. Среда разработки AVR Studio

Письменный опрос.

Сделать детальное описание режима отладки программы.

Письменный опрос проводится во время урока. На письменный опрос отводится 10 мин.

Тема 2.6. Отладка программ

Практическая работа № 2 Разработка программы устройства управления светодиодным индикатором при помощи одной кнопки

Практическая работа № 3 Создание программы на языке Си устройства с мигающим светодиодом

Цель работы: Получить практические навыки по использованию возможностей среды разработки ArduinoIDE для составления программ;

2. Время выполнения работы-2 часа

3.Краткие теоретические сведения

Микроконтроллер программируется с помощью компьютера и может работать самостоятельно или в сочетании с ПК. К микроконтроллеру могут присоединяться различные аналоговые и цифровые датчики, которые регистрируют состояние окружающей среды и передают данные в микроконтроллер. После обработки данных программа микроконтроллера может вывести информацию на монитор или, например, осуществить управление приводом. Поддержка проектов разработчиков осуществляется за счет готовых программ и библиотек функций среды программирования Arduino.

Микроконтроллер ATmega328 имеет 32 Кбайт программной памяти (из которых 2 Кбайт используются для хранения загрузчика). Кроме того, микросхема имеет 2 Кбайт ОЗУ (SRAM), 1 Кбайт байт долговременной памяти данных (EEPROM).

Задание

Составьте программу управления включением и выключением светодиода, подключенного к 13-му ПИНу микроконтроллера с параметрами согласно варианту, указанному в таблице 1.

Таблица 1 – Варианты для практической работы

№	ПИН	Время горения светодиода, с	Время паузы, с
1	2	1	3
2	4	3	5
3	6	8	7
4	3	9,3	9,5
5	5	4	2,2
6	9	1,2	4
7	12	2	6
8	10	11,5	5
9	13	5	3,5
10	11	1	2
11	15	15	9
12	16	3,5	8
13	24	4,8	7,5
14	27	6	5,5
15	28	8,8	6,4
16	25	4	4
17	18	6	5
18	19	2,5	2

4. Порядок выполнения.

4.1. Изучить описание примера программы

Первая строка объявляет обязательную функцию **setup()**.

Функция **setup()** вызывается, когда стартует скетч (программа для микроконтроллера). Используется для инициализации переменных, определения режимов работы выводов, запуска используемых библиотек и т.д. Она производит вызов стандартной функции **pinMode**, которая устанавливает режим работы заданного вход/выхода (ПИН) как входа или как выхода.

С помощью данной функции 13-й ПИН микроконтроллера переводится в режим **вывода**, чтобы затем можно было управлять напряжением на этом контакте.

4.2. Написать и выполнить программу для управления работой светодиодом с периодом включения и выключения, указанным в варианте.

5. Указания к выполнению отчета

Отчет должен содержать:

- наименование и цель работы.
- код программы, написанной в соответствии с вариантом.
- выводы по проделанной работе

6. Контрольные вопросы

1. Перечислите основные компоненты среды разработки Arduino IDE
2. Назначение функции setup().
3. С помощью какой функции можно обеспечить подачу HIGH или LOW на выход микроконтроллера?
4. С помощью какой функции задается пауза?

Практическая работа № 4 Разработка автомата «бегущие огни»

1. Цель работы: Получить практическое представление о реализации конечных автоматов с использованием возможностей среды разработки ArduinoIDE

2. **Время выполнения работы- 2 часа**
3. **Краткие теоретические сведения**

Задание

Составьте программу с различными вариантами управления светодиодами.

4. Порядок выполнения.

4.1. Изучить описание примера программы

Программа предполагает три варианта горения четырех светодиодов. Вариант будет выбираться пользователем командой с ПК. При отправке «1» светодиоды будут последовательно загораться через каждые четверть секунды, при отправке «2» светодиоды будут последовательно загораться в обратном порядке через каждые четверть секунды, при отправке «3» светодиоды будут загораться в порядке 1 – 4 – 3 – 2 с интервалом времени в 0.5 секунды. Для ввода команд необходимо открыть **Serial Monitor (Инструменты – Монитор порта)** и выставить скорость передачи данных, как в написанной программе.

```
int value =
0; int led1
= 2;
int led2 = 4; int led3 = 6;
int led4 = 3;
void setup()
{
Serial.begin(9600);
pinMode(led1,
OUTPUT);
pinMode(led2,
OUTPUT);
pinMode(led3,
```

```
OUTPUT);
void loop()
{
if (Serial.available())
{
value = Serial.read();
}
switch(value)
{
case(1): {
digitalWrite(led1,
HIGH); delay(250);
digitalWrite(led2,
HIGH); delay(250);
digitalWrite(led3,
HIGH); delay(250);
digitalWrite(led4,
HIGH); delay(250);
break;
}
case(2):
{
digitalWrite(led4, HIGH);
delay(250);
digitalWrite(led3, HIGH);
delay(250);
digitalWrite(led2, HIGH);
delay(250);
digitalWrite(led1, HIGH);
delay(250);
break;

}case(3):
{
digitalWrite(led1, HIGH);
delay(500);
digitalWrite(led2, HIGH);
delay(500);
digitalWrite(led3, HIGH);
delay(500);
digitalWrite(led4, HIGH);
delay(500);
```

```

    break;
  }
}
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
digitalWrite(led
4, LOW);
}

```

Переменная **value**, будет содержать значение варианта горения светодиодов, и переменные ПИНов светодиодов. Далее инициализируем UART со скоростью 9600 и настраиваем ПИНЫ светодиодов на выход. В бесконечном цикле мы считываем данные из UART и записываем их в переменную value. Далее, в зависимости от значения **value**, выполняется определенный вариант горения светодиодов.

4.2. Написать и выполнить программу для управления работой четырех светодиодов, варианты горения светодиодов, выдержки времени и номера ПИН выбрать самостоятельно.

5. Указания к выполнению отчета

Отчет должен содержать:

- наименование и цель работы.
- код программы, написанной в соответствии с вариантом.
- выводы по проделанной работе

6. Контрольные вопросы

1. В каких случаях целесообразно применение конечного автомата?
2. Назначение переменной value
3. Назначение переменной case

Практическая работа № 5 Разработка программы «бегущие огни» с использованием прерываний по таймеру.

Цель работы: Получить практическое представление о реализации прерываний по таймеру в микроконтроллерах ATmega328 автоматов с использованием возможностей среды разработки ArduinoIDE

1. Время выполнения работы- 2 часа

2. Краткие теоретические сведения

Прерывание это событие, при котором происходит приостановка основной программы и переход на выполнение подпрограммы прерывания.

При этом в стек записывается содержимое счетчика команд, а в сам счетчик записывается вектор прерывания, по которому находится команда безусловного перехода к подпрограмме обработки прерывания. После выполнения подпрограммы обработки прерывания, в счетчик команд загружается сохраненное значение из стека и выполнение основной программы продолжается с того места, где оно остановилось.

Таблица векторов прерывания располагается в памяти программ, начиная с адреса 0x0002, либо в области загрузчика. Приоритет прерывания зависит также от его расположения в таблице векторов прерывания: чем меньше адрес, тем выше приоритет прерывания.

Задание

Составьте программу последовательного включения каждого из четырех светодиодов с циклическим повторением

3. Порядок выполнения.

3.1. Изучить описание примера программы `#include<avr/io.h>`

```
#include <avr/interrupt.h>
```

```
void InitTimer()
```

```
{
```

```
TCCR1A|=(1<<COM0A1)|(0<<COM0A0)|(0<<WGM00)| (0<<WGM01);
```

```
TCCR1B|= (0<<CS02) | (1<<CS01)| (0<<CS00);
```

```
TIMSK0|= (1<< OCIE1A);
```

```
TCNT1= 0;
```

```
OCR1A = 1999;
```

```
}
```

```
void setup() {
```

```
long counter = 1;
```

```
int led1 =2;
```

```
int led2 =4;
```

```
int led3 =6;
```

```
int led4 =3;
```

```
InitTimer();
```

```
pinMode(led1,
```

```
OUTPUT);
```

```
pinMode(led2,
OUTPUT);
pinMode(led3,
OUTPUT);
pinMode(led4,
OUTPUT);
}
ISR(TIMER0_COMPA_vec
t)
{
if(counter <= 4000)
{
counter++; }
else {
counter = 0;
}
}
void loop()
{
switch(counter)
{
case(1000):
{
digitalWrite(led1,
HIGH);
digitalWrite(led2,
LOW);
digitalWrite(led3,
LOW);
digitalWrite(led4,
LOW); break;
}
}
```

```
case(2000):
{
digitalWrite(led1, LOW);
digitalWrite(led2, HIGH);
digitalWrite(led3, LOW);
digitalWrite(led4, LOW);
break;
}
case(3000):
{
digitalWrite(led1,
LOW);
digitalWrite(led2,
LOW);
digitalWrite(led3,
HIGH);
digitalWrite(led4,
LOW); break;
}
case(4000):
{
digitalWrite(led1,
LOW);
digitalWrite(led2,
LOW);
digitalWrite(led3,
LOW);
digitalWrite(led4,
HIGH); break;
}
case(0):
{
```

```
digitalWrite(led1,  
LOW);  
digitalWrite(led2,  
LOW);  
digitalWrite(led3,  
LOW);  
digitalWrite(led4,  
LOW);  
    break;  
}  
}  
}
```

Timer1-функция инициализации 16-битного таймера, который будет работать с прерыванием каждую 1 мс.

3.2. Написать и выполнить программу для последовательного включения каждого из четырех светодиодов с циклическим повторением, варианты горения светодиодов, выдержки времени и номера ПИН выбрать самостоятельно.

4. Указания к выполнению отчета

Отчет должен содержать:

- наименование и цель работы.
- код программы, написанной в соответствии с вариантом.
- выводы по проделанной работе

5. Контрольные вопросы

1. Опишите процедуру обработки прерываний
2. От чего зависит приоритет прерывания?
3. Какими способами можно производить прерывания по таймеру в микроконтроллере ATmega328?

Практическая работа № 6 Создание программы сигнального устройства с звуковым выходом

1. ЦЕЛЬ ЗАНЯТИЯ

1. Изучить необходимые для написания программы функции;
2. Написать программу для работы с пьезоэлементом (динамиком).
2. **КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

В качестве звукового вывода будет использоваться пьезоэлемент, подключенный к микроконтроллеру. Пьезоэлемент — электромеханический преобразователь, одним из разновидностей которого является пьезоизлучатель звука, который также называют пьезодинамиком, просто звонком или английским buzzer. Пьезодинамик переводит электрическое напряжение в колебание мембраны. Эти колебания и создают звук (звуковую волну).

Управлять звуковым выводом можно двумя способами:

- 1) Использовать функцию `analogWrite(pin, value)`;

Выдает аналоговую величину (ШИМ волну) на порт вход/выхода. Функция может быть полезна для управления яркостью подключенного светодиода или скоростью электродвигателя. После вызова `analogWrite()` на выходе будет генерироваться постоянная прямоугольная волна с заданной шириной импульса до следующего вызова `analogWrite` (или вызова `digitalWrite` или `digitalRead` на том же порту вход/выхода). Частота ШИМ сигнала приблизительно 490 Hz.

Параметры:

- `pin`: порт вход/выхода на который подаем ШИМ сигнал;
- `value`: период рабочего цикла значение между 0 (полностью выключено) и 255 (сигнал подан постоянно).

Частота звука будет зависеть от заданного значения `value`.

- 2) Использовать функции `tone(pin, frequency, duration)` и `noTone(pin)`.

Функция `tone()` генерирует на порту вход/выхода сигнал – прямоугольную «волну», заданной частоты и с 50% рабочим циклом. Длительность может быть задана параметром `duration`, в противном случае сигнал генерируется пока не будет вызвана функция `noTone()`. К порту вход/выхода может быть подключен пьезо или другой динамик для воспроизведения сигнала.

Воспроизводиться одновременно может только один сигнал. Если сигнал уже воспроизводится на одном порту, то вызов `tone()` с номером другого порта в качестве параметра ни к чему не приведет, если же `tone()` будет вызвана с тем же номером порта, то будет установлена новая частота сигнала.

Параметры:

- `pin`: номер порта вход/выхода, на котором будет генерироваться сигнал;
- `frequency`: частота сигнала в Герцах;
- `duration`: длительность сигнала в миллисекундах.

3. **ПОРЯДОК ВЫПОЛНЕНИЯ ПРАКТИЧЕСКОЙ РАБОТЫ**

Напишем программу, которая будет сигнализировать звуковым сигналом и горящим светодиодом о нажатии кнопки.

```

int button = 6; int led = 11;
int buzzer = 12; void setup()
{
pinMode(led, OUTPUT); pinMode(button, INPUT); pinMode(buzzer, OUTPUT);
}
void loop()
{
if (digitalRead(button) == HIGH)

{
digitalWrite(led, HIGH); tone(buzzer, 100);
}
else
{
digitalWrite(led, LOW); noTone();
}
}

```

Сначала объявляем переменные со значениями ПИНов кнопки, светодиода и динамика. Далее настраиваем выходы светодиода и динамика на выход, а вывод кнопки – на вход. В бесконечном цикле, при нажатой кнопке будет гореть светодиод и генерироваться звуковой сигнал с частотой 100 Гц.

4. ВАРИАНТЫ

ПИНЫ кнопки, светодиода, динамика и частоту звукового сигнала выбрать самостоятельно.

5. ОТЧЕТ

Отчет по практической работе должен содержать:

1. Наименование и цель работы.
2. Краткие теоретические данные по практической работе.
3. Код программы, написанной в соответствии с вариантом.
4. Вывод по проделанной работе.

Практическое занятие №7 Разработка (проектирование) устройства «музыкальная шкатулка»

1. ЦЕЛЬ ЗАНЯТИЯ

1. Изучить схему и принцип работы устройства;

2. Написать программу с различными простейшими музыкальными

ком - позициями.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1. ФУНКЦИЯ ДЛЯ РАБОТЫ С ВНЕШНИМИ ПРЕРЫВАНИЯМИ

Для написания программы для «Музыкальной шкатулки» будут использоваться внешние прерывания. Прерывание (англ. interrupt) — сигнал, сообщающий процессору о наступлении какого-либо события. При этом выполнение текущей последовательности команд приостанавливается, и управление передаётся обработчику прерывания, который выполняет работу по обработке события и возвращает управление в прерванный код. Для работы с ними используется функция `attachInterrupt(interrupt, function, mode)`.

Параметры функции:

1) `interrupt`: номер прерывания или `pin`: номер цифрового порта (только для Arduino Due)

2) `function`: функция, вызываемая прерыванием, функция должна быть без параметров и не возвращать значений.

3) `mode` задает режим обработки прерывания. Допустимо использование следующих констант:

- `LOW` вызывает прерывание, когда на порту `LOW`;
- `CHANGE` прерывание вызывается при смене значения на порту, с `LOW` на `HIGH` и наоборот;
- `RISING` прерывание вызывается только при смене значения на порту с `LOW` на `HIGH`;
- `FALLING` прерывание вызывается только при смене значения на порту с `HIGH` на `LOW`;

ATmega328 может обрабатывать до двух внешних прерываний на ПИНе 4 (`INT0`) и ПИНе 5 (`INT1`). В скобках указан номер прерывания.

2.2. СТРУКТУРНАЯ СХЕМА И ПРИНЦИП РАБОТЫ УСТРОЙСТВА

На рисунке 1 представлена структурная схема электрической части устройства.

Плата с компонентами помещаются в шкатулку. Кнопка закрепляется в месте соприкосновения крышки и основной части шкатулки. При закрытой крышке кнопка нажата и на выводе 4 установлен сигнал `HIGH` и музыка не проигрывается. При открытии шкатулки кнопка отжимается, значение на выводе 4 сменяется на `LOW`, срабатывает прерывание и вызывается функция проигрывания музыкальной композиции.

Для проигрывания музыки будут использоваться функции `tone()` и `noTone()`, изученные на Практическом занятии №6. В таблице 1 представлены значения частот нот 1 и 2 октавы и обозначения для каждой ноты.

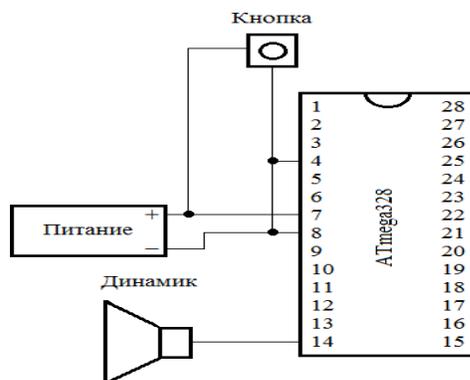


Рисунок 1. Структурная схема электрической части устройства

Таблица 1 – Частоты и обозначения нот

1 октава	Обозначение	Частота, Гц	2 октава	Обозначение	Частота, Гц
до	C	261	до	c	523
до-диез	C1*	277	до-диез	c1*	554
ре	D	293	ре	d	587
ре-диез	D1*	311	ре-диез	d1*	622
ми	E	329	ми	e	659
фа	F	349	фа	f	698
фа-диез	F1*	370	фа-диез	f1*	740
соль	G	392	соль	g	784
соль-диез	G1*	415	соль-диез	g1*	830
ля	A	440	ля	a	880
ля-диез	B	466	ля-диез	b	932
си	H	494	си	h	988

*см. – C1 означает C# и т.д. для остальных нот.

В качестве воспроизводимой мелодии будет использоваться «В траве сидел кузнечик». Ниже представлены используемые для мелодии ноты.

E-C-E-C-E-F-F – В траве сидел кузнечик; F-C-F-C-F-E-E – Совсем как огуречик;

E-C-E-F-C-F-C-F-E-E – В траве сидел кузнечик; C-E-F – Зелененьким он был;

F-G-G-G-G-G-G#-G#-G#-G# – Представьте себе, представьте себе; G#-G#-G-F-E-F-F – Совсем как огуречик;

F-G-G-G-G-G-G#-G#-G#-G# – Представьте себе, представьте себе; G#-G#-G-F-E-F-F – Совсем как огуречик;

3. ПОРЯДОК ВЫПОЛНЕНИЯ ПРАКТИЧЕСКОЙ РАБОТЫ

Ниже представлен код программы для данного устройства.

```
// МЕЛОДИЯ – массив нот и массив длительностей char melody[]={ 'E', 'C', 'E ',
'C', 'E', 'F', 'F',
  'F', 'C', 'F', 'C', 'F', 'E', 'E',
  'E', 'C', 'E', 'F', 'C', 'F', 'C', 'F', 'E', 'E', 'C', 'E', 'F',
  'F', 'G', 'G', 'G', 'G', 'G', 'G', 'G#', 'G#', 'G#', "G#", 'G#', 'G#', 'G', 'F', 'E', 'F', 'F',
  'F', 'G', 'G', 'G', 'G', 'G', 'G#', 'G#', 'G#', "G#",
  'G#', 'G#', 'G', 'F', 'E', 'F', 'F', '*'}; int bb[]={4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 8, 8,
4, 4, 4, 2, 2, 4, 4, 4, 2, 2,
4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 2, 2, 4, 4, 4, 2, 2,
4, 4, 4, 4, 4, 4, 64};
// массив для наименований нот в пределах двух октав char
names[]={ 'C', 'C1', 'D', 'D1', 'E', 'F', 'F1', 'G', 'G1', 'A', 'B',
  'H', 'c', 'c1', 'd', 'd1', 'e', 'f', 'f1', 'g', 'g1', 'a', 'b', 'h'};
// массив частот нот
int tones[]={261,277,293,311,329,349,370,392,415,440,466, 494,
523,554,587,622,659,698,740,784,830,880,932,988};
int buzzer = 14; int button = 4; int notes, beats;

void setup()
{
  pinMode(buzzer, OUTPUT); pinMode (button, INPUT); attachInterrupt(0, play,
LOW)
}

void loop()
{
}

void play()
{
  for (int i=0; i < sizeof(melody); i++)
  {
    notes = melody[i]; beats = bb[i];
    if (notes == '*')
    {
```

```

}
else
{
}

tone(buzzer, 0 , beats); // пауза

playNote(notes, beats); // воспроизвести ноту

// пауза между нотами delay(beats);
}
}

// функция проигрыша ноты
void playNote(char note, int duration)
{
for (int i = 0; i < sizeof(tones); i++)
{
if (names[i] == note)
{
tone(buzzer, tones[i], duration);
}
}
}
}

```

Сначала объявляются массивы нот и их длительностей для воспроизводимой мелодии. Далее объявляются массивы нот двух октав и их частоты. Далее объявляются переменные ПИНов кнопки и динамика, а также переменные нот и длительностей. В функции setup() настраиваются параметры выводов, к которым присоединены кнопка и динамик, а также задается функция прерывания вывода 4 (INT0), которые при значении LOW на выводе 4 (шкатулка открыта) будет вызывать функцию play(). Функция loop() в программе не используется, так как работа происходит через обработчик прерываний. В функции play() выполняется бесконечный цикл, в котором при каждом шаге цикла воспроизводится нота номера этого шага до того момента, пока ноты в мелодии не закончатся и воспроизведение не начнется заново.

4. ВАРИАНТЫ

Варианты выводов кнопки (4 или 5), динамика выбрать самостоятельно. Выбор воспроизводимой мелодии (нот и их длительности) производить при помощи

интернета.

5. ОТЧЕТ

Отчет по практической работе должен содержать:

1. Наименование и цель работы.
2. Краткие теоретические данные по практической работе.
3. Код программы, написанной в соответствии с вариантом.
4. Вывод по проделанной работе.

Практическое занятие №8 Разработка кодового замка

1. ЦЕЛЬ ЗАНЯТИЯ

1. Ознакомиться с принципами работы и использованием последовательного интерфейса UART;
2. Изучить необходимые функции по работе с UART;
3. Написать программу по управлению светодиодом при помощи кнопки, с возможностью блокировки управления с ПК.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1. ПОСЛЕДОВАТЕЛЬНЫЙ ИНТЕРФЕЙС UART

Интерфейс UART наиболее распространённый последовательный интерфейс современных микроконтроллеров. UART – это универсальный приемопередатчик. Данные передаются последовательным кодом в формате, представленным на рисунке 1.



Рисунок 1. Структура передаваемой информации по UART

Передача каждого байта происходит за равные промежутки времени. Время передачи одного бита определяется скоростью передачи. Она указывается в бодах (бит в секунду). Кроме битов информации, UART вставляет в поток биты синхронизации (стартовый и стоповый). Таким образом, для передачи байта информации необходимо не 8, а 10 бит. В таблице 1 представлены стандартные скорости передачи интерфейса UART.

Таблица 1 – Скорости передачи данных по UART

Скорость передачи, бод	Время передачи одного бита, мкс	Время передачи байта, мкс
4800	208	2083
9600	104	1042
19200	52	521
38400	26	260
57600	17	174
115200	8,7	87

Обмен информацией через UART происходит в дуплексном режиме, т.е. передача данных может происходить одновременно с приемом. Для этого в UART есть два сигнала:

- TX – выход для передачи данных;
- RX – вход для приема данных.

При соединении двух UART устройств выход TX одного устройства соединяется с входом RX другого, а вывод TX второго UART подключается к входу RX первого.

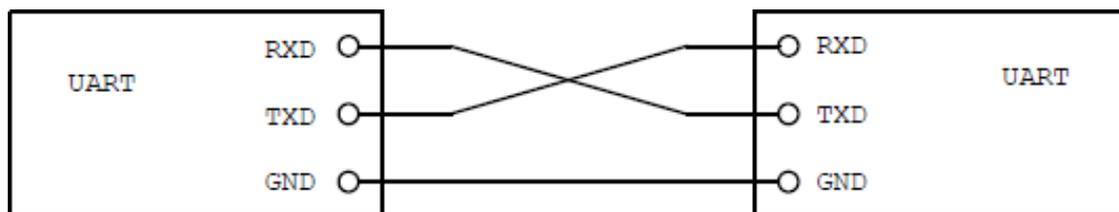


Рисунок 2. Схема подключения двух устройств по UART

2.2. UART В МИКРОКОНТРОЛЛЕРЕ АТМЕГА328

Микроконтроллер ATmega328 имеет один порт UART, сигналы которого подключены к выводам 2 (PD0) (сигнал RX) и 3 (PD1) (сигнал TX). Через эти выводы можно подключить к микроконтроллеру другое устройство имеющее интерфейс UART.

Кроме связи с другими микроконтроллерами, порт UART используется для загрузки в контроллер программы с компьютера. Для этого к выводам RX и TX подключаются соответствующие выводы преобразователя USB/UART. Преобразователь позволяет подключать контроллер к компьютеру по USB. На компьютер устанавливается специальный драйвер, создающий виртуальный COM – порт, через который происходит обмен. Тем самым можно производить управление микроконтроллером с компьютера.

3. ПОРЯДОК ВЫПОЛНЕНИЯ ПРАКТИЧЕСКОЙ РАБОТЫ

Напишем программу по управлению светодиодами с помощью кнопки. Управление будет доступно, после введения определенного кодового значения с ПК. При отправке «1345» управление светодиодами будет разрешено и при нажатии кнопки светодиод будет загораться. Чтобы запретить управление, с ПК необходимо отправить любую другую команду, отличную от разрешающей. После этого сообщение светодиод не будет реагировать на нажатие кнопки. Для ввода команд необходимо открыть Serial Monitor (Инструменты – Монитор порта). Также в мониторе необходимо выставить скорость передачи данных как в написанной программе.

Сначала объявляем переменную value, в которую будет записываться кодовое значение с компьютера, и переменные ПИНов светодиода и кнопки. Далее инициализируем UART со скоростью 9600 и настраиваем ПИНЫ светодиода на выход, а кнопки на вход. В бесконечном цикле мы считываем данные из UART и записываем их в переменную value. После, в зависимости от значения value, управление светодиодом с кнопки разрешено либо запрещено.

4. ВАРИАНТЫ

Кодовое значение, номера ПИНов светодиода и кнопки выбрать самостоятельно.

5. ОТЧЕТ

Отчет по практической работе должен содержать:

1. Наименование и цель работы.
2. Краткие теоретические данные по практической работе.
3. Код программы, написанной в соответствии с вариантом.
4. Вывод по проделанной работе.

Практическое занятие №9-10 Разработка кодового устройства с музыкальным звонком

1. ЦЕЛЬ ЗАНЯТИЯ

1. Изучить схему и принцип работы устройства;
2. Написать программу с различными вариантами кодовых комбинаций и музыкального звонка.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

В данной практической работе представлена вариация кодового замка с использованием кнопок, подключенных непосредственно к микроконтроллеру, в отличие от, где кодовая комбинация вводилась при помощи ПК. На рисунке 1 представлена структурная схема программируемого устройства. В таблице 1 приводятся используемые ПИНЫ микроконтроллера и их назначение.

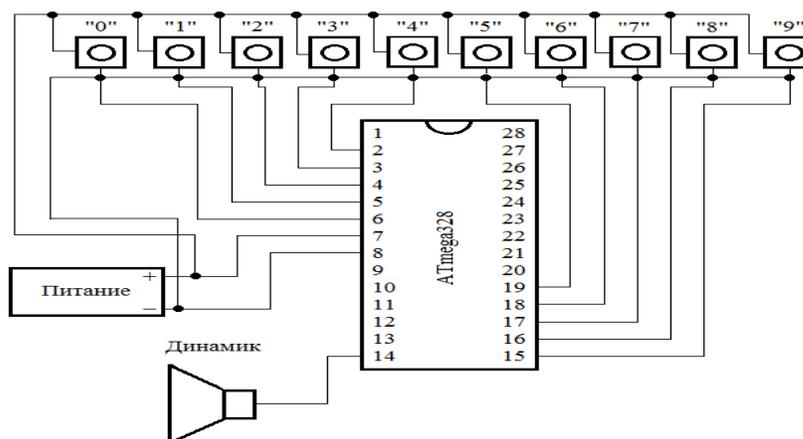


Рисунок 1. Структурная схема устройства

Таблица 1 – Назначение ПИНов микроконтроллера

Номер ПИНа	Назначение
2	Чтение состояния кнопки «4»
3	Чтение состояния кнопки «3»
4	Чтение состояния кнопки «2»
5	Чтение состояния кнопки «1»
6	Чтение состояния кнопки «0»

14	Вывод сигнала на динамик
15	Чтение состояния кнопки «9»
16	Чтение состояния кнопки «8»
17	Чтение состояния кнопки «7»
18	Чтение состояния кнопки «6»
19	Чтение состояния кнопки «5»

Устройство постоянно находится в состоянии чтения выводов, к которым подключены кнопки. После последовательного нажатия четырех кнопок (ввода определенной кодовой комбинации), микроконтроллер сравнивает какие кнопки нажаты и в каком порядке. Если введена верная комбинация (для данного примера верная комбинация «1369»), то в течение пяти секунд будет генерироваться звуковой сигнал. В случае ввода неверной комбинации, будет генерироваться три прерывистых сигнала, с периодом звучания и затишья в 1 секунду.

4. ВАРИАНТЫ

Кодовую комбинацию, звуковые сигналы (частота, время звучания, выдержки времени) при верной и неверной кодовой комбинации выбрать самостоятельно.

5. ОТЧЕТ

Отчет по практической работе должен содержать:

1. Наименование и цель работы.
2. Краткие теоретические данные по практической работе.
3. Код программы, написанной в соответствии с вариантом.
4. Вывод по проделанной работе.

Самостоятельная работа обучающегося:

Выполнение индивидуальных заданий по созданию программ микроконтроллера в соответствии с заданием на разработку электронного устройства

4.КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

4.1.Назначение

Контрольно-оценочное средство предназначено для промежуточной аттестации по дисциплине ОП.08. Микропроцессорные системы оценки знаний и умений аттестуемых, а также элементов ПК и ОК.

4.2.Форма и условия аттестации

Аттестация проводится в форме экзамена по завершению освоения всех тем учебной дисциплины, при положительных результатах текущего контроля.

К экзамену по дисциплине допускаются студенты, полностью выполнившие все лабораторные работы и практические задания по дисциплине.

Контрольно-оценочные средства для проведения промежуточной аттестации доводятся до сведения студентов не позднее, чем за месяц до окончания изучения дисциплины. На основе разработанного и объявленного обучающимся перечня теоретических вопросов и практических задач, рекомендуемых для подготовки к экзамену, составляются экзаменационные билеты, содержание которых до обучающихся не доводится. Комплект билетов по своему содержанию охватывает все основные вопросы пройденного материала по предмету. Число экзаменационных билетов разрабатывается больше числа студентов в экзаменуемой группе. Номер экзаменационного билета для обучающихся определяется с помощью генератора случайных чисел.

Экзамен проводится в специально подготовленных помещениях. На выполнение задания по билету студенту отводится не более 1 академического часа. В случае неточных и неполных ответов обучающего на вопросы экзаменационного билета преподаватель вправе задать дополнительные вопросы из перечня включенных в оценочное средство в форме блиц-опроса (без предварительной подготовки). Во время сдачи промежуточной аттестации в устной форме в аудитории может находиться одновременно не более 4-6 обучающихся.

Перечень вопросов для экзаменационных билетов

1. Комбинационная логика, основные элементы (И, ИЛИ, НЕ), их схемы на КМОП транзисторах.
2. RS-триггер. Простейший триггер на биполярных транзисторах. RS-триггер на логических элементах И-НЕ, ИЛИ-НЕ. Синхронный RS триггер.
3. Статический D триггер, схема, принцип работы. Таблица истинности D триггера. Принципиальная схема статического D триггера на элементах И-НЕ, ИЛИ-НЕ. Динамические D триггеры. Временные диаграммы работы D триггера.
4. T-триггер (счетный триггер). Схема T триггера (на основе D триггера). 4-битный

- суммирующий счётчик на основе T-триггеров, его временная диаграмма. 4-битный вычитающий счётчик на основе T-триггеров, его временная диаграмма.
5. Регистры на основе D-триггеров. Параллельные регистры, последовательные (сдвиговые) регистры. Временная диаграмма работы сдвигового регистра.
 6. Общая структура микроконтроллерного устройства, АЛУ, программный счётчик.
 7. Виды памяти микроконтроллеров (на примере семейства AVR). Стек и его инициализация.
 8. Прерывания, вектор прерываний.
 9. Основы языка Assembler для микроконтроллеров семейства AVR. Синтаксис, классификация команд.
 10. Устройство портов ввода-вывода. Упрощённая схема порта ввода-вывода микроконтроллера. Управляющая логика, определяющая конфигурацию порта ввода-вывода. Режимы работы: выход, вход с подтяжкой к питанию (PullUp). Состояние высокого импеданса.
 11. Таймеры-счётчики. Источники тактовой частоты. Регистры, определяющие работу таймера. Предделитель таймера. Основные режимы работы таймера: стандартный счётный режим (Normal), режим сброса при совпадении. Источники прерываний таймера.
 12. Широтно-импульсная модуляция. Понятие широтно-импульсной модуляции. Различные режимы широтно-импульсной модуляции. Режим быстрый ШИМ (Fast PWM). ШИМ без фазового сдвига (Phase Correct PWM).
 13. Цифро-аналоговые преобразователи. Структурная схема ЦАП. Параллельная схема суммирования токов. Последовательная схема суммирования токов.
 14. Аналого-цифровые преобразователи. АЦП параллельного преобразования (параллельные АЦП). АЦП последовательного приближения. Интегрирующие АЦП. Сигма-дельта АЦП. Схемы и принцип работы, основные преимущества и недостатки.
 15. Структурная схема встроенного аналого-цифрового преобразователя, его основные параметры. Настройка АЦП, управляющие регистры. Прерывания по завершению цикла преобразования.
 16. Передача данных с использованием асинхронного последовательного интерфейса UART. Общая структура асинхронного приёмопередатчика. Протокол передачи данных RS-232. Настройка и программирование интерфейса UART. Скорость передачи данных.
 17. Передача данных с использованием асинхронного последовательного интерфейса SPI. Схема организации SPI интерфейса, сдвиговые регистры. Пакетная передача данных. Диаграмма передачи данных по SPI, режимы передачи данных.

Управляющие регистры, скорость передачи данных. Работа в режиме передачи (Master) и приёма данных (Slave)

18.Интерфейс I2C. Протокол передачи данных, режимы работы. Основные преимущества и недостатки интерфейса.