ПРИЛОЖЕНИЕ

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ В.Ф УТКИНА»

Кафедра «Электронные приборы»

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДИСЦИПЛИНЫ**

**МИКРОПРОЦЕССОРЫ В ЭЛЕКТРОННЫХ УСТРОЙСТВАХ**

Фонд оценочных средств – это совокупность учебно-методических материалов (контрольных заданий, описаний форм и процедур), предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной образовательной программы.

 Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям основной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

 Основная задача – обеспечить оценку уровня сформированности общекультурных и профессиональных компетенций, приобретаемых обучающимся в соответствии с этими требованиями.

 Контроль знаний обучающихся проводится в форме текущего контроля и

промежуточной аттестации.

 При оценивании (определении) результатов освоения дисциплины применяется традиционная система (отлично, хорошо, удовлетворительно, неудовлетворительно).

**1. Паспорт фонда оценочных средств по дисциплине**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ п/п** | **Контролируемые разделы (темы) дисциплины****(результаты по разделам)** | **Код контроли-****руемой компетен-ции (или её части)** | **Этап формирования** **контролируемой компетенции (или её части)** | **Вид, метод, форма****оценочного****средства** |
|
| 1 | 2 | 3 | 4 | 5 |
| 1 | Основы вычислительной техники | ПК-1.1ПК-2.1ПК-2.2ПК-3.2 | Лекционные и самостоятельные занятия обучающихся в течение учебного семестра | Экзамен, сдача лабораторных работ |
| 2 | Позиционные системы счисления.  | ПК-1.1ПК-2.1ПК-2.2ПК-3.2 | Лекционные, лабораторные и самостоятельные занятия обучающихся в течение учебного семестра | Экзамен, сдача лабораторных работ |
| 3 |  Представление данных в процессоре.  | ПК-1.1ПК-2.1ПК-2.2ПК-3.2 | Лекционные, лабораторные и самостоятельные занятия обучающихся в течение учебного семестра | Экзамен, сдача лабораторных работ |
| 4 | Основные способы адресации.  | ПК-1.1ПК-2.1ПК-2.2ПК-3.2 | Лекционные, лабораторные и самостоятельные занятия обучающихся в течение учебного семестра | Экзамен, сдача лабораторных работ |
| 5 | Виды и форматы команд | ПК-1.1ПК-2.1ПК-2.2ПК-3.2 | Лекционные, лабораторные и самостоятельные занятия обучающихся в течение учебного семестра | Экзамен, сдача лабораторных работ |
| 6 | Архитектура учебного процессора.  | ПК-1.1ПК-2.1ПК-2.2ПК-3.2 | Лекционные и самостоятельные занятия обучающихся в течение учебного семестра | Экзамен, сдача лабораторных работ |
| 7 | Периферийные устройства микропроцессоров.  | ПК-1.1ПК-2.1ПК-2.2ПК-3.2 | Лекционные и самостоятельные занятия обучающихся в течение учебного семестра | Экзамен, сдача лабораторных работ |

**2.Формы текущего контроля**

Текущий контроль успеваемости проводится с целью определения степени усвоения учебного материала, своевременного выявления и устранениянедостатков в подготовке обучающихся и принятия необходимых мер по совершенствованию методики преподавания учебной дисциплины,организации работы обучающихся в ходе учебных занятий и оказания иминдивидуальной помощи.

Текущий контроль по дисциплине проводится в виде проверки заданий, выполняемых самостоятельно и на лабораторных занятиях, а также в виде опросов на практических занятиях.

**3. Формы промежуточного контроля**

Формой промежуточного контроля по дисциплине является экзамен.К экзамену допускаются обучающиеся, полностью выполнившие все виды учебной работы, предусмотренные учебным планом и настоящей программой. Форма проведения экзамена – устный ответ, по утвержденным экзаменационным билетам, сформулированным с учетом содержания учебной дисциплины.

**4. Критерии оценки компетенций обучающихся и шкалы**

**оценивания**

Оценка формирования компетенций у обучающихся на различных этапах проводится преподавателем во время лекций, консультаций, лабораторныхзанятийпо шкале оценок «зачтено» – «не зачтено».

Освоение материала дисциплины и достаточно высокая степень формирования контролируемых компетенций обучающегося служат основанием для допуска, обучающегося к этапу промежуточной аттестации - экзамену.

Целью проведения промежуточной аттестации (экзамена) является проверка компетенций, приобретенных студентом при изучении дисциплины.

Экзамен организуется и осуществляется, как правило, в форме собеседования. Средством, определяющим содержание собеседования студента с экзаменатором, является утвержденный экзаменационный билет, содержание которого определяется ОПОП и рабочей программой предмета. Экзаменационный билет включает в себя, как правило, два вопроса относящихся к теоретическим разделам дисциплины.Оценке на заключительной стадии экзамена подвергаются устные ответы экзаменующегося на вопросы экзаменационного билета, а также дополнительные вопросы экзаменатора.

Применяются следующие критерии оценивания компетенций (результатов):

-уровень усвоения материала, предусмотренного программой;

 -умение анализировать материал, устанавливать причинно-следственные связи;

- полнота, аргументированность, убежденность ответов на вопросы;

 -качество ответа (общая композиция, логичность, убежденность, общая эрудиция);

 -использование дополнительной литературы при подготовке к этапу промежуточной аттестации.

Применяется четырехбальная шкала оценок: "отлично", "хорошо", "удовлетворительно", "неудовлетворительно", что соответствует шкале "компетенции студента полностью соответствуют требованиям ФГОС ВО", "компетенции студента соответствуют требованиям ФГОС ВО", "компетенции студента в основном соответствуют требованиям ФГОС ВО ", " компетенции студента не соответствуют требованиям ФГОСВО".

К оценке уровня знаний и практических умений и навыков рекомендуется предъявлять следующие общие требования.

**«Отлично»:**

глубокие и твердые знания программного материала программы дисциплины, понимание сущности и взаимосвязи рассматриваемых явлений (процессов);

полные, четкие, логически последовательные, правильные ответы на поставленные вопросы; умение выделять главное и делать выводы.

**«Хорошо»:**

достаточно полные и твёрдые знания программного материала дисциплины, правильное понимание сущности и взаимосвязи рассматриваемых явлений (процессов);

последовательные, правильные, конкретные, без существенных неточностей ответы на поставленные вопросы, свободное устранение замечаний о недостаточно полном освещении отдельных положений при постановке дополнительных вопросов.

**«Удовлетворительно»:**

знание основного программного материала дисциплины, понимание сущности и взаимосвязи основных рассматриваемых явлений (процессов):

понимание сущности обсуждаемых вопросов, правильные, без грубых ошибок ответы на поставленные вопросы, несущественные ошибки в ответах на дополнительные вопросы.

**«Неудовлетворительно»:**

отсутствие знаний значительной части программного материала дисциплины; неправильный ответ хотя бы на один из вопросов, существенные и грубые ошибки в ответах на дополнительные вопросы, недопонимание сущности излагаемых вопросов, неумение применять теоретические знания при решении практических задач, отсутствие навыков в обосновании выдвигаемых предложений и принимаемых решений.

При трех вопросах в билете общая оценка выставляется следующим образом: «отлично», если все оценки «отлично» или одна из них «хорошо»; «хорошо», если не более одной оценки «удовлетворительно»; «удовлетворительно», если две и более оценок «удовлетворительно»; «неудовлетворительно», если одна оценка «неудовлетворительно», а остальные не выше чем «удовлетворительно» или две оценки «неудовлетворительно».

5. Вопросы к экзаменупо дисциплине микропроцессоры **в электронных устройствах**

1. Позиционные системы счисления

2. Перевод целых десятичных чисел в двоичную форму.

3. Перевод дробной части в двоичную систему.

4. Шестнадцатиричная система счисления, перевод из шестнадцатиричной системы в десятичную.

5. Бит, байт слово, двойное слово.

6. Сложение целых десятичных чисел.

7. Дополнительный и обратный коды, вычитание целых двоичных чисел.

8. Представление чисел с плавающей запятой.

9. Представление символов.

10. Байтовая адресация памяти, прямой и обратный порядок байтов в словах.

11. Непосредственная, регистровая и абсолютная типы адресации, пример команд.

12. Индексная и индексная базовая адресация, пример применения.

13. Относительная адресация через счетчик команд, пример применения.

14. Автоинкрементная и автодекрементная адресация, пример применения.

15. Стек, организация, стековая адресация, пример использования.

16. Команды перемещения данных между регистрами и ячейками памяти.

17. Регистр-аккумулятор, назначение, особенности использования.

18. Регистровые пары, назначение, загрузка регистровых пар.

19. Регистр флагов, флаги S,Z,C,P – назначение, использование.

20. Линейный код и организация ветвлений.

21. Команды передачи управления, организация циклов.

22. Команды условных и безусловных переходов.

23. Подпрограммы - назначение, вызов, возврат, передача параметров.

24. Команды обработки данных, логические и арифметические операции.

25. Прерывание - понятие, назначение, команды обработки прерываний, векторные прерывания.

26. Процессор – базовые концепции. Шинная архитектура.

27. Структура модели процессора.

28. Последовательность микрокоманд при пересылке данных между регистрами.

29. Последовательность микрокоманд при выполнении арифметических и логических операций.

30. Структура простейшего АЛУ.

31. Последовательность микрокоманд при выборке-записи слова в память.

32. Аппаратное управление элементами процессора, понятие микрокоманды.

33. Программное управление элементами процессора, микропрограмма.

34. Микропроцессор КР580ВМ80А (INTEL 8085) - структура, состав системы, память, адресное пространство, периферия.

35. Микропроцессор КР580ВМ80А (INTEL 8085) – сигналы управления, синхронизации, понятие машинных тактов и машинного цикла, назначение регистра состояния.

36. Основные принципы RISC архитектуры. Сопоставление RISC и CISC архитектур.

6. Типовые задания к лабораторным занятиям по дисциплине "**Микропроцессоры в электронных устройствах**"

**Задание №1**

1. Прочитать содержимое ячеек памяти с адреса 02CB H по 02D4 H.

2. Программа сравнения чисел, содержащаяся в ячейках 84F0 H и 84F1 H. Программа загружается с адреса 8400.

Исходные данные для трёх вариантов:

1. (84F0) = 218 D, (84F1) = 55 D

2. (84F0) = 55 D, (84F1) = 218 D

3. (84F0) = 13 D, (84F1) = 13 D

**Задание №2**

1. Сложить содержимое трёх ячеек (02C0, 02C1 и 02C2). Результат занести в ячейку 821C.

2. Сравнить содержимое ячеек 03C0 и 03C1. Если в ячейке 03C1 число меньше, то включить светодиодный индикатор «Флаг Z», иначе закончить программу.

**Задание №3**

 Составить программу с использованием системы индикации микро ЭВМ «Электроника-580» для следующего алгоритма:

В ячейках памяти хранятся следующие шестнадцатеричные числа:

(8400) = 1A H, (8401) = FB H

1. сложить числа, хранящиеся в ячейках памяти 8400 и 8401;
2. проанализировать содержимое младшего разряда результата;
3. если оно равно \*\*\* если оно не равно ---

нулю то на первый \* - нулю, то на пятый - \*

индикатор вывести \* - индикатор вывести - \*

символ \* - символ - \*

 --- ---

 \* - - \*

 \* - - \*

 \* - - \*

 \*\*\* ---

(знаком «-» отмечены погашенные сегменты индикаторов)

Числа в ячейки памяти загрузить с помощью программы «монитор»

**Задание №4**

Написать подпрограмму возведения числа в заданную степень (xy). Число, степень и результат содержатся в теле основной программы.

7. Пример лабораторной работы по дисциплине "М**икропроцессоры в электронных устройствах**"

Работа 4

**ОРГАНИЗАЦИЯ ПОДПРОГРАММ И ИСПОЛЬЗОВАНИЕ СТЕКА**

Цель работы: изучение основ разработки подпрограмм.

1.*Общие сведения*

При разработке программ часто имеет место ситуация, когда неоднократ­но необходимо выполнять некоторую последовательность команд. Для сокра­щения размера программы и уменьшения объема памяти, выделяемого под про­грамму, такие участки программы оформляются в виде подпрограмм. Подпро­граммы не только позволяют экономить ресурсы ЭВМ, но и являются мощным средством разработки программного обеспечения. Поскольку подпрограмма является самостоятельнымпрограммным модулем, она может быть написана и отлажена независимо от основной программы, что ускоряет отладку всей 'про­граммы. Отлаженные подпрограммы можно хранить в библиотеках и использо­вать их по мере необходимости.

Подпрограмма - это последовательность команд, выполнение которых может быть вызвано из любого места программы необходимоеколичество раз. Процесс передачи управления к подпрограмме называется вызовом. Данные и адреса, требуемые для работы подпрограммы, называются входными парамет­рами, а информация, передаваемая в основную программу, по окончании ее выполнения - выходными параметрами. Обращение к подпрограмме выполня­ется командой CALLADR, где ADR - адрес первой выполняемой команды под программы, а возврат -командой RET.

2. *Стек*

Стеком называют часть памяти, доступ к которой организован по прин­ципу: "последним записан - первым считан". Наиболее важное использованиестека связано с реализацией вызова подпрограмм. Команда CALLADR не только осуществляет переход по указанному адресу, но и включает в стек текущее содержимое программного счетчика PC, т.е. адресвозврата из подпрограммы. При входе в. подпрограмму в стеке запоминается содержимое регистров, которые использует подпрограмма. Перед выходом из подпрограммы эти данные восстанавливаются и удаляются из стека. Подпро­грамма заканчивается однобайтовой командой возврата RET, которая извлекает из стека адрес возврата и передает его ь программный счетчик.

При обращении к стеку используется косвенная адресация через специ­альный регистр - указатель стека (УС или SP - stackpoint). При записи данных в стек производится автоматический декремент указателя стека, а при считыва­нии - инкремент. Помещение данных в стек называется включением (PUSH), а обратное действие - извлечением (POP). Адрес последнего включенного в стек элемента называется вершиной стека.

В стек записывается содержимое 16-разрядных регистров. В ячейку по адресу (SP)-l заносится старший байт регистра, а по адресу (SP)-2 - младший байт регистра. После выполнения команды содержимое указателя стека уменьшается на 2. При извлечении из вершины стека в младший байт регистра записывается содержимое ячейки с адресом (SP), а в старший байт регистра - содержимое ячейки с адресом (SP)+1. При этом содержимое указателя стека увеличивается на 2 (см. рисунок 1).

При работе со стеком в начале программы необходимо инициализировать указатель стека - командой LXISP, ADR загрузить я регистр SP адрес, который на 1 больше начального адреса области стека (при включении ЭВМ монитор загружает в SP адрес 83Е0). В начале каждой подпрограммы командами PUSHв стеке сохраняется содержимое тех регистров, которые используются в под­программе. Перед выходом из подпрограммы командами POP восстанавливается содержимоерегистров. Восстановление регистроввыполняется в обратной последовательности по отношению к их записи в стек. Все операции со стеком должны быть сбалансированы - каждая подпрограмма должна содержать одинаковое количество команд PUSH и POP и заканчиваться командой RET.



Рис.1. Обращение к стеку

3. *Пример программы*

Подпрограмма вычисляет среднее значение двух чисел. Исходные числа размещаются в двух ячейках сразу за командой вызова подпрограммы, а ре­зультат помещается в следующей ячейке памяти.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Адрес | Содерж.ячейки | Метки | Мнемоника | Операнд | Коментарий |
| 8200 | CD |  | CALL | AVR | вызов подпрограммы вычисления |
| 8201 | 50 |  |  |  | среднего двух чисел |
| 8202 | 82 |  |  |  |  |
| 8203 | 07 |  | DB | 07 | первое число |
| 8204 | 0F |  | DE | 0F | второе число |
| 8205 |  |  |  |  | ячейка памяти для хранения среднего |
| 8206 | E7 |  | RST |  | с этой ячеки продолжается программа |
|  |  |  |  |  |  |
| 8250 | E3 | AVR: | XTHL |  | сохранение HL всеке, выборка адреса |
|  |  |  |  |  | первого числа в региструвую пару HL |
| 8251 | F5 |  | PUSH | PSW | сохранение слова состояния процессора |
| 8252 | 7E |  | MOV | A,M | заносим в аккумулятор первое число |
| 8253 | 23 |  | INX | H | получаем адрес второго числа |
| 8254 | 86 |  | ADD | M | суммируем два числа |
| 8255 | A7 |  | ANA | A | очистка флага переноса |
| 8256 | 1F |  | RAR |  | делении суммы на 2 сдвигом вправо |
| 8257 | 23 |  | INX | H | получаем адрес результата |
| 8258 | 77 |  | NOV | M,A | запись результата в память |
| 8259 | 76 |  | INX | H | адрес возврата из подпрограммы |
| 825А | F5 |  | POP | PSW | восстановлении слова состояния |
| 825В | E3 |  | XTHL |  | восстановление адреса возврвта |
| 825С | C9 |  | RET |  | возврат из подпрограммы |

4.*Задания для подготовки к работе*

Ознакомиться с принципами организации подпрограмм и использования стека в микроЭВМ. Подготовить отчет, в котором должны быть:

1. Описаниекоманд CALL, RET, RST6, PUSH PSW, POP H, XTHL.

2. Бланк для программ по образцу таблиц описания.

5.*Задания к лабораторной работе*

5.1. *Исследовать подпрограмму расчета среднего двух чисел*

Порядок выполнения задания:

1. Ввести в микроЭВМ программу.

2. Проверить правильность работы программы в режиме ОТЛАДКА.

3. Дополнить программу командами вычисления среднего значения чисел 80 и 80. Повторить выполнение программы и объяснить полученный результат.

4. Записать d ячейку 8255 код 00. Повторить выполнение программы и объяснить полученный результат.

5.2. *Разработать и отладить программу*.

Порядок выполнения задания:

1. Получить у преподавателя задание на разработку программы.

2. Составить алгоритм\* программу на языке ассемблера и перевести в машинные коды, заполнив все колонки бланка программы.

3. Включить микроЭВМ и ввести в память программу.

4. Отладить программу и продемонстрировать ее работу преподавателю.

6. *Содержание отчета*

1. Выполненные задания при подготовке к работе (п.4).

2. Результаты выполнения заданий п.5.1.

3. Задание на разработку программы (п.5.2). Алгоритм в виде блок-схемы, полностью оформленный бланк программы на языке ассемблера. Результаты работы программы с их объяснением.

*7.Вопросы для подготовки к работе*

1. С какой целью используются подпрограммы?

2. Как можно организовать передачу параметров в подпрограмму?

3. Дайте определение стека, указателя стека, вершины и дна стека.

4.В какой последовательности записывается и считывается из стека содержимое PSW при выполнении команд PUSH PSW и POP PSW?

5. Опишите способы адресации, применяемые при обращении к стеку.

6. Какими командами можно задать или изменить область памяти под

стек?

7. В чем разница между командами CALL и RST?

Полный перечень заданий и вопросов к лабораторным работам, выполняемым для приобретения и развития знаний и практических умений, предусмотренных компетенциями, приведен в соответствующих методических указаниях.

1. Микропроцессоры в электронных устройствах: методические указания к лабораторным работам / Рязан. гос. радиотехн. ун-т; сост.: А.В. Зуев. Рязань, 2011. 43 с.