

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Курганский государственный университет»

Кафедра «Автоматизация производственных процессов»

**МИКРОКОНТРОЛЛЕРЫ И МИКРОПРОЦЕССОРЫ
В СИСТЕМАХ УПРАВЛЕНИЯ**

Методические указания для практических занятий по дисциплинам
«Микроконтроллеры и микропроцессоры в системах управления»,
«Микропроцессорные устройства»
для студентов направлений подготовки

15.03.04 «Автоматизация технологических процессов и производств»
(профиль «Автоматизация технологических процессов и производств
в машиностроении),

27.03.04 «Управление в технических системах» (профиль «Системы и
технические средства автоматизации и управления»)

Курган 2018

Кафедра: «Автоматизация производственных процессов».

Дисциплина: «Микроконтроллеры и микропроцессоры в системах управления»

Составил: канд. техн. наук, доц. Е.К. Карпов.

Утверждены на заседании кафедры

21 декабря 2017 г.

Рекомендованы методическим советом университета

12 декабря 2016 г.

1 Принципы построения системы управления групповой замерной установкой. Подготовка, редактирование, трансляция и отладка программ на симуляторе и при загрузке в контроллер, Работа с имитатором объекта управления

Для автоматизации производственных процессов используют программируемые логические контроллеры (ПЛК). Одним из примеров таких контроллеров является семейство КЕДР-2. Контроллер КЕДР-2, предназначен для логической и арифметической обработки информации, поступающей с датчиков и управления внешними устройствами в соответствии с программой, записанной в его ППЗУ (программируемое постоянное запоминающее устройство), а также для обмена информацией.

Контроллер относится к перепрограммируемым микропроцессорным устройствам и может выполнять дополнительные функции при наличии соответствующего программного обеспечения.

Программы для контроллера КЕДР-2 пишутся на ассемблере. Система команд совместима с КР580ИК80.

Формат регистра флагов:

S	Z	0	AC	0	P	1	C
---	---	---	----	---	---	---	---

S – знак (0 – «+», 1 – «-»)

Z – признак нуля (1 – результат операции равен нулю)

AC – вспомогательный перенос из младшей тетрады в старшую

P – паритет (0 – чет, 1 – нечет)

C – перенос из разряда в разряд за границы разрядной сетки (из старшего разряда при сложении или из младшего - при сдвигах.)

Цель работы заключается в освоении лабораторного комплекса, последовательности разработки, трансляции программы, предварительной отладки ее на эмуляторе, преобразования типа файла, загрузки в контроллер и выполнения ее с сохранением результата. Проработка системы команд и техники программирования. Проверка выполнения отдельных команд с анализом изменения/сохранения признаков результата.

Реализация команды ADD (суммирования) в контроллере “Кедр”:

Код программы:

op1 equ 50h

op2 equ 40h

mem1 equ 3870h

mem2 equ 3871h

mem3 equ 3872h

org 3850h

;Задаем адрес для программы в ОЗУ

mvi a,op1

;Записываем в регистры a и b два

mvi b,op2

;Операнда для сложения

ora a

;Поразрядное логическое сложение

```

push psw          ;Сохраняем с стек PSW
xthl              ;Загрузка данных из SP
mov a,l
sta mem2          ;Сохраняем их в памяти по адресу
xthl
pop psw           ;Возвращаем из стека PSW
add b             ;Операция сложения
sta mem1          ;Запись результата в память
push psw
xthl              ;обмен данными между SP и H-L
mov a,l
sta mem3          ;Запись SP после операции сложения в память
pop psw
ret               ;Завершение работы программы

```

Для проверки признаков результата попробуйте при помощи программы сложить следующие наборы чисел:

- 1) A=50h, B=40h;
- 2) A=20h, B=30h;
- 3) A=90h, B=90h.

Запишите полученные вами результаты и значения флагового регистра до и после операции суммирования. Сравните их со следующими выводами:

1) перед выполнением операции сложения 50h и 40h признаки результата имели значение 000 100. После выполнения операции сложения мы получили значение A=90h и признаки результата 100 100. Изменение старшего разряда с 0 на 1 означает, что полученное число является отрицательным;

2) перед выполнением операции сложения 20h и 30h признаки результата имели значение 000 000. После выполнения операции сложения мы получили значение A=50h и признаки результата 000 000. Неизменность признаков результата показывает, что было получено чётное положительное число;

3) перед выполнением операции сложения 90h и 90h признаки результата имели значение 100 100. После выполнения операции сложения мы получили значение A=20h и признаки результата 000 101. Изменение младшего разряда с 0 на 1 означает, перенос из разряда в разряд за границы разрядной сетки.

Попробуйте ответить на вопросы, чем обусловлено наличие признака отрицательности результата во флаговом регистре? Чему оказывается равен результат сложения в случае превышения 100h и как определить, что это произошло?

2 Развитие 51 семейства микроЭВМ. Параметры МК80С51F125.

Система Контар, контроллеры МС8.3, МС6. Язык FBD

Для ознакомления с особенностями построения программ на языке Функциональных блочных диаграмм в инструментальной среде “Конграф”, создайте новый проект и постройте следующие простые схемы, демонстрирующие особенности использования программных блоков:

1) защита от дребезга контакта (способ 1) (рисунок 1) – это пример комплексного алгоритмического блока, реализующего защиту от дребезга контакта. Сигнал с контакта подается на преобразователь типов: булева переменная в аналоговую. Полученная аналоговая величина фильтруется и подаётся на компаратор верхнего уровня, на выходе которого формируется логический сигнал единицы в том случае, если выходной сигнал фильтра превысил пороговое значение, являющееся параметром компаратора верхнего уровня.

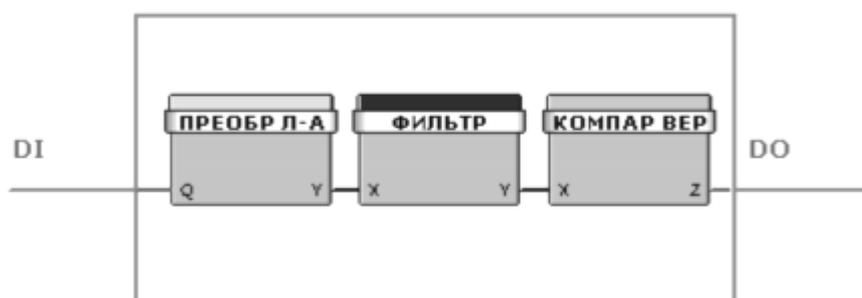


Рисунок 1 – Защита от дребезга контакта

2) защита от дребезга контакта (способ 2) (рисунок 2) – это пример использования функционального блока для реализации защиты от дребезга контакта. Сигнал с контакта подается на фильтр тревог, который выполняет функцию фильтрации цифрового сигнала. Он выдает на выходе логическую единицу или ноль в случае, если на входе блока какое-то время держится логическая единица или ноль соответственно. Время удержания на входе Q неизменяемой величины зависит от параметров инкремента/декремента и предела, устанавливаемых в свойствах блока при счете на увеличение или уменьшение.

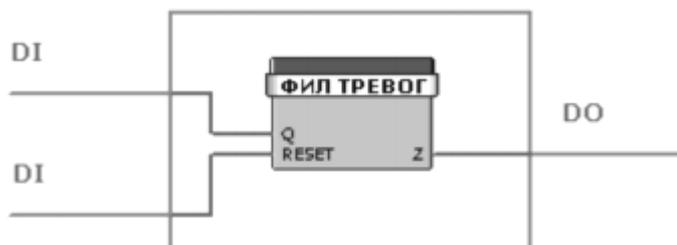


Рисунок 2 – Защита от дребезга при помощи фильтра тревог

3) комплексный блок определения выхода измеряемой величины из заданного диапазона (рисунок 3) – определяет, находится ли сигнал от датчика температуры в заданном диапазоне. Если измеренная температура меньше нижнего температурного предела или выше его, то соответствующие компараторы (компаратор нижнего уровня и компаратор верхнего уровня)

формируют логическую единицу на своих выходах. Выходы компараторов подаются на входы блока логического “ИЛИ”, на выходе которого формируется флаг того, что измеренная температура находится вне заданного диапазона.

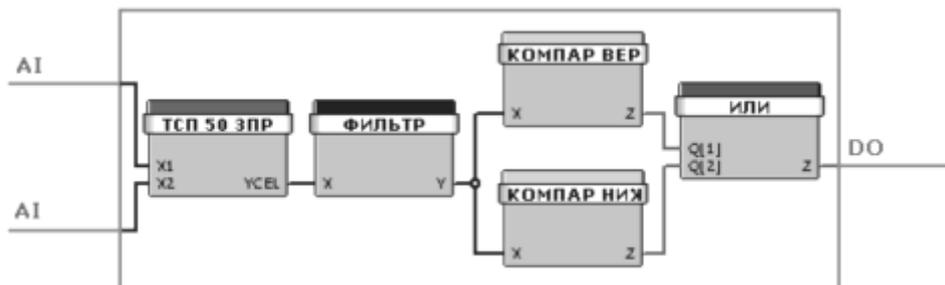


Рисунок 3 – Комплексный блок определения выхода измеряемой величины из заданного диапазона

4) ПИД-регулятор (рисунок 4) – аналоговый регулятор с обратной связью, состоящий из блока задания аналоговой величины, формирующего задание на одном из входов блока определения сигнала ошибки. Аналоговый сигнал обратной связи от объекта управления подается на вход AI и далее через термистор и фильтр, сглаживающий выбросы сигнала обратной связи, поступает на второй вход блока разности. Сигнал рассогласования (разность между заданием и сигналом обратной связи) подается на вход аналогового ПИД-регулятора, выход которого подается на объект управления.

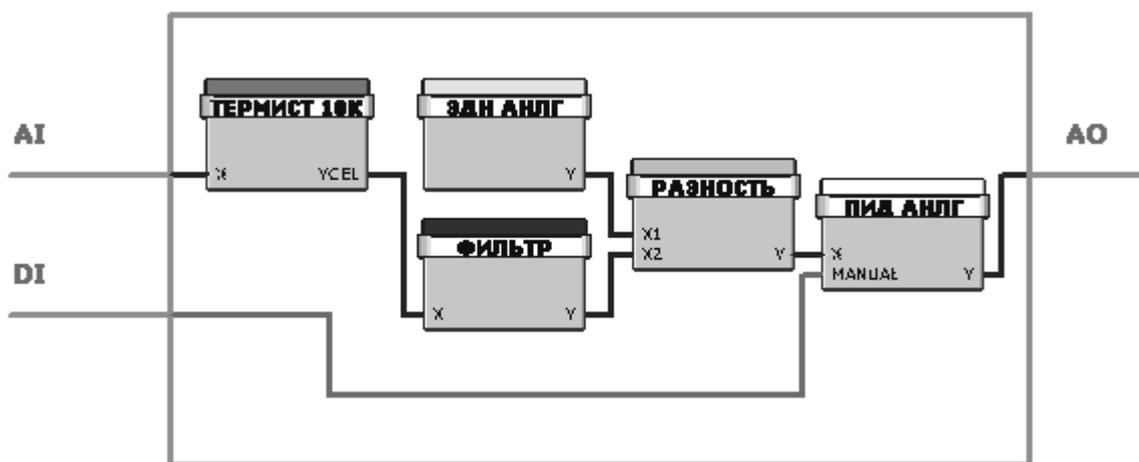


Рисунок 4 – ПИД-регулятор

Перестройте схему с рисунка 3 в соответствии с имеющимся на учебном стенде оборудованием, загрузите программу в контроллер и проверьте её работоспособность, выводя сигнал о выходе из допустимого диапазона температур на один из красных светодиодов.

3 Анализ одной-двух самостоятельных/контрольных работ, доклад группы студентов

Блок-схема – распространенный тип схем, описывающих алгоритмы или процессы, в которых отдельные шаги изображаются в виде блоков различной формы, соединенных между собой линиями, указывающими направление последовательности. Правила выполнения регламентируются ГОСТ 19.701-90.

Основные элементы блок-схем представлены на рисунке 5.



Рисунок 5 – Элементы блок-схем

Процесс – выполнение операции или группы операций, в результате чего изменяется значение, форма представления или расположения данных. Внутри символа или же в виде комментария на естественном языке или в виде формулы записываются действия, которые производятся при выполнении операции или группы операций.

Решение – выбор направления выполнения алгоритма или программы в зависимости от некоторых переменных условий.

Ввод/вывод – ввод или вывод данных вне зависимости от физического носителя.

Пуск – Останов – начало, конец, прерывание процесса обработки данных или выполнения программы.

Соединитель – указание связи между прерванными линиями потока, связывающими символы. Если блок-схема состоит из нескольких частей, расположенных на одной странице, то линия потока одной части заканчивается символом Соединителя, а линия потока на продолжении блок-схемы начинается с этого же символа. Внутри символов Соединителя ставятся одинаковые порядковые номера, соответствующие разорванной линии потока.

Выполните следующие задания и попробуйте ответить на вопросы тестирования.

Разделитесь на подгруппы аналогично подгруппам на лабораторных работах и вычертите блок-схемы трёх программ для стендов, на которых вы сейчас занимаетесь. Задания возьмите из методических указаний к текущим лабораторным.

Тестирование

1. Алгоритмы называются эквивалентными если:

- а) они написаны на одном и том же языке программирования;
- б) они имеют одинаковое количество строк;
- в) они выполняются одинаковое количество времени;

г) результаты, получаемые с помощью этих алгоритмов для одних и тех же исходных данных, одинаковы.

2. Что выполняет команда SWAP A?

- а) обмен байтов;
- б) обработку запрещенной команды;
- в) обмен тетрад аккумулятора.

3. Программа написанная на языке ASSEMBLER это:

- а) псевдокод;
- б) программный код;
- в) графический способ описания алгоритма.

4. Возможна ли косвенная адресация битов в микропроцессоре МК51?

- а) Да, через регистр R0;
- б) нет;
- в) да, с ограничением зоны прямоадресуемых битов.

5. Операции, имеющие два операнда называются:

- а) бинарные;
- б) унарные;
- в) арифметические;
- г) логические.

6. Каково основное назначение регистра SP и в каких ситуациях необходимо учитывать его содержимое?

- а) Для выбора текущей команды процессора;
- б) Для сохранения адреса возврата из подпрограммы;
- в) Для выбора одного из процессоров в мультипроцессорной системе.

7. Каково назначение указателя DPTR микропроцессора I8051, и в каких случаях он используется?

а) Расширитель шины (РШ) используется для работы с внешним ЗУ – памяти программ или памяти данных. Как правило, внешнее ЗУ используется тогда, когда для размещения программы или данных при решении какой-то задачи внутренних ресурсов ОМЭВМ оказывается недостаточно; режим работы с внешним ЗУ не является типичным для ОМЭВМ.

б) Для косвенной адресации в операциях регистр-регистр

в) Для указателя рабочего банка регистров; режим работы с внешним ЗУ не является типичным для ОМЭВМ.

8. Каково назначение регистра В микропроцессора I8051, и в каких случаях он используется?

- а) Для резервирования аккумулятора

б) Для хранения старшего байта при выполнении команд умножения/деления

в) Для использования к качестве старшего байта регистровой пары ВС

9. Какими командами можно определить знак произведения двух операндов?

а) Команда MUL АВ перемножает целые числа без знаков из аккумулятора и регистра В. Произведение двух восьмиразрядных чисел будет, естественно, иметь длину 16 разрядов и занимать 2 байта. Старший байт произведения помещается в регистр В, а младший – в аккумулятор. Если результат произведения превышает 255, то устанавливается флаг переполнения ОV в регистре признаков, в противном случае он сбрасывается. В любом случае сбрасывается флаг переноса т.е. знак произведения нужно запоминать;

б) По знаковому разряду регистра В;

в) По признаку знака в PSW.

10. Каким образом оперативно обработать изменение дискретного входного сигнала не теряя контроля других процессов?

а) Использовать процедуру сканирования сигнала;

б) Подключить сигнал к входу запроса на прерывание;

в) Вставить в программы других процессов опрос данного входного сигнала с соответствующей обработкой.

11. Какой должна быть последовательность действий в подпрограмме обработки прерываний по запросу последовательного порта?

а) Принять очередной байт в буфер приема, проверить не переполнен ли буфер и ждать следующий символ. Прерывания от таймера/счетчика Т/С1 в этом режиме должны быть заблокированы.

б) При приеме или передаче – сохранить контекст, убедиться в отсутствии ошибок, обработать посылку, сбросить флаги прерываний, восстановить контекст и выйти из подпрограммы командой RETI;

в) При передаче или приеме – обработать очередной байт в регистре данных, проверить на кончился ли массив передачи и ждать окончания передачи байта.

12. Если флаг прерывания был установлен, но не получил обслуживания и к моменту окончания блокировки уже был сброшен, то:

а) запрос прерывания теряется и нигде не запоминается;

б) по команде оператора возможно совершить восстановление ситуации с обработкой пропущенного запроса;

в) обработать запрос можно предварительно подготовленной программой для трехкратного опроса сигнала.

13. Какой командой выполняется инверсия битов в МК51?

а) Команда XRL может быть эффективно использована для инверсии отдельных бит;

б) Специальной командой JBC bit, rel;

в) SUBB A, ad.

14. Запись main() означает, что это:

а) моя программа;

б) главная функция;

в) функция вычисления максимального значения.

15. Каким образом можно изменить обработку запросов на прерывания?

а) Обработка прерываний задается жестко при инициализации системы;

б) Прерывания могут быть вызваны или отменены программой, так как все флаги прерываний программно-доступны и могут быть установлены/сброшены программой с тем же результатом, как если бы они были установлены/сброшены аппаратурными средствами;

в) По командам оператора можно изменить приоритет запросов на прерывания.

16. Возможна ли косвенная адресация битов в микропроцессоре МК51?

а) да, через регистр R0;

б) нет;

в) да, с ограничением зоны прямоадресуемых битов

17. Блок-схема это способ описания алгоритма:

а) псевдокодом;

б) программным кодом;

в) графический;

г) табличный.

18. На каком этапе проектирования проверяется правильность работы программного проекта.

а) разработка внутренних структур данных;

б) постановка задачи;

в) проектирование структуры и взаимодействие модулей;

г) структурное проектирование;

д) тестирование и отладка.

19. Можно ли обработку внешних прерываний INT0 и INT1 в МК51 запрограммировать как краевой?

а) Нет, обработка внешних прерываний INT0 и INT1 могут быть запрограммированы только как уровневые (запрос на прерывание формируется, если уровень логического сигнала на входе равен 0),

б) Да, обработка внешних прерываний INT0 и INT1 могут быть запрограммированы как уровневые (запрос на прерывание формируется, если

уровень логического сигнала на входе равен 0) или как краевые (сигнал на входе переходит из состояния логической единицы в логический 0).

в) Прерывания INT0 (INT1) всегда интерпретируются процессором как краевые, при этом поступающий запрос на прерывание запоминается и, если он поступает в момент времени, когда данное прерывание запрещено или не может быть обслужено, может быть обслужен позже, когда прерывание будет разрешено им возможно (и даже будет обслужен, если не принять мер по сбросу флага IE0 (IE1)).

20. Какую особенность построения запоминающих устройств имеет микропроцессор I8051, (МК51)?

а) Память программ и память данных доступны с использованием трех шин: адреса, данных и управления;

б) Память программ и память данных, размещенные на кристалле МК51, физически и логически разделены, имеют различные механизмы адресации, работают под управлением различных сигналов и выполняют разные функции;

в) Запоминающие устройства МК51 организованы в виде набора регистров специальных функций и могут использовать битовую адресацию.

Задания для выполнения группами студентов к следующему практическому занятию: Подготовьте к следующему практическому занятию блок-схему по заданию одной из лабораторных работ, уточнив тему у преподавателя, и сделайте доклад с презентацией о применении микроконтроллера для решения этой задачи. Доклад должен включать краткое описание контроллера, автоматизированной системы, области её применения, необходимые фрагменты программного кода или скриншоты, комментарии к ним а также саму блок-схему и выводы по работе. Студентам заочной формы обучения можно использовать для доклада материалы своей контрольной работы.

4 Контроль быстротекущих процессов, понятие принципа обработки прерываний таймера

Цель работы заключается в освоении лабораторного стенда, последовательности разработки, трансляции программы, предварительной отладки ее на эмуляторе, преобразования типа файла, загрузки в контроллер и выполнения ее с сохранением результата. Проработка системы команд и техники программирования “Кедр”.

Напишите программу вращения двигателя по следующему алгоритму: 5 секунд – по часовой стрелке, останов на 3 секунды, 5 секунд вращение против часовой стрелки, 3 секунды останов.

Исходный код программы:

;Программа выполняется по прерыванию таймера – базовый период 20 мс

;Таймер - это устройство микропроцессора служащее для отсчета

;импульсов, от тактового генератора микропроцессора.

;Программа начинается с директивы

org 3850h ;Это означает что программа сохранится в ОЗУ

;по адресу 3850 (в шестнадцатеричной ;системе счисления)

;далее определяется константа для таймера:

delay equ 100

;ниже находится основной код программы:

setint:

push h

;эта команда для сохранения регистра h (два байта) в специальном месте

;ОЗУ – так называемом стеке, для того чтобы не нарушить прерванную

;программу и, после завершения своей программы, восстановить

;содержимое регистра.

LXI h,newint

;Эта команда загружает регистр h словом(16 бит), newint – это метка

;т.е. двухбайтный адрес (в данном случае 3859)

shld 3f72h

;Запись ОЗУ по адресу 3f72 двумя байтами из регистра h, по этому адресу

;содержится адрес обработки прерываний, заменяемый на 3859h

pop h

;возврат сохраненного регистра h в стеке в регистр h.

ret

;программа закончилась но по прерыванию таймера процессор перейдет

;на адрес newint(3859) содержимое которого:

newint:

push a

push d

push h

lhld paus ; в рег. паре H-L

xchg ; в рег. паре D-E

lhld count ; загружаем количество вызовов программы

```

    inx    h           ; увеличиваем его
    shld   count       ; сохраняем на прежнем месте
    mov    a,h
    sub    d
    jnz    quit
;команда проверит флаг и если d больше или меньше 0, то
;произойдет переход на метку quit, т.е. к стандартному обработчику 150h,
;но после 20мс опять будет выполняться newint
    mov    a,l
    sub    e
    jnz    quit
;команда проверит флаг и если e не равно 0, то команда прыгнет на метку
;quit – т.е. к стандартному обработчику 150h но после 20мс
;опять будет выполняться newint.
    lxi    h,0
    shld   count
    lda    state
    inr    a           ;увеличивает содержимое регистра a на 1
    sta    state
;вводим новую переменную которую используем для счета состояний
;1)вращение две секунды по часовой
;2)две секунды стоп
;3)вращение две секунды против часовой
;4)две секунды стоп
    cpi    4
;сравним счетчик состояний и, если сделали полный цикл состояний
;(т.е. a<4), то восстанавливаем прежний обработчик прерывания:
    jm    direction
    sub    a           ; сделали обнуление байта состояний
    sta    state
jmp    movestop;
;Эта команда загружает в рабочий регистр байт по адресу count –
;это переменная, которая определена в конце программы и используется
;для накопления.
;Определение точки возврата:
direction:
    cpi    0
    jz    movestop
    cpi    1
    jz    moveleft
    cpi    2
    jz    movestop
    cpi    3
    jz    moveright
; загружаем два байта движения против часовой стрелки

```

```

moveleft:
    lxi d,5dch
    lda left
    jmp write
;Переход без условия на метку write – где происходит вывод в порт.
;загружаем байт нулевой скорости
movestop:
    lxi d,150
    lda stop
    jmp write
; загружаем два байта движения по часовой стрелке
moveright:
    lxi d,5dch
    lda right
    jmp write
write:
    out 78h
;запись регистра в порт 78h,
;по получении контроллером байта, двигатель начнет вращаться.
quit:
    lxi h,0
    dad d
    shld state+1
    pop h
    pop d
    pop a
    jmp 150h
;Перейти в стандартный обработчик прерывания находящийся в ПЗУ
;далее определяются использованные в программе переменные:
left:  db 0f0h ; движение влево
right: db 0ffh ; движение вправо
stop:  db 0deh ;остановка
count:dw 0 ; счетчик обращений
state: db 0 ; состояние
paus:  dw 5dch
end
;конец

```

Нарисуйте блок-схему работы программы и ответьте на вопросы:

Какое максимальное время вращения привода в одну сторону можно задать в данной программе, изменяя только значение константы таймера? Подумайте, как изменить данную программу, чтобы увеличить максимально допустимое значение константы таймера.

СОДЕРЖАНИЕ

1 ПРИНЦИПЫ ПОСТРОЕНИЯ СИСТЕМЫ УПРАВЛЕНИЯ ГРУППОВОЙ ЗАМЕРНОЙ УСТАНОВКОЙ. ПОДГОТОВКА, РЕДАКТИРОВАНИЕ, ТРАНСЛЯЦИЯ И ОТЛАДКА ПРОГРАММ НА СИМУЛЯТОРЕ И ПРИ ЗАГРУЗКЕ В КОНТРОЛЛЕР, РАБОТА С ИМИТАТОРОМ ОБЪЕКТА УПРАВЛЕНИЯ.....	3
2 РАЗВИТИЕ 51 СЕМЕЙСТВА МИКРОЭВМ. ПАРАМЕТРЫ МК80С51F125. СИСТЕМА КОНТАР, КОНТРОЛЛЕРЫ МС8.3, МС6. ЯЗЫК FVD.....	5
3 АНАЛИЗ ОДНОЙ-ДВУХ САМОСТОЯТЕЛЬНЫХ/КОНТРОЛЬНЫХ РАБОТ, ДОКЛАД ГРУППЫ СТУДЕНТОВ.....	7
4 КОНТРОЛЬ БЫСТРОТЕКУЩИХ ПРОЦЕССОВ, ПОНЯТИЕ ПРИНЦИПА ОБРАБОТКИ ПРЕРЫВАНИЙ ТАЙМЕРА.....	12

Карпов Егор Константинович

**МИКРОКОНТРОЛЛЕРЫ И МИКРОПРОЦЕССОРЫ
В СИСТЕМАХ УПРАВЛЕНИЯ**

Методические указания для практических занятий по дисциплинам
«Микроконтроллеры и микропроцессоры в системах управления»,

«Микропроцессорные устройства»

для студентов направлений подготовки

15.03.04 «Автоматизация технологических процессов и производств»
(профиль «Автоматизация технологических процессов и производств
в машиностроении»),

27.03.04 «Управление в технических системах» (профиль «Системы и
технические средства автоматизации и управления»)

В авторской редакции

Подписано в печать 05.04.18	Формат 60x84 1/16	Бумага 65 г/м ²
Печать цифровая	Усл. печ. л. 1,00	Уч.-изд. л. 1,00
Заказ №69	Тираж 25	Не для продажи

БИЦ Курганского государственного университета.
640020, г. Курган, ул. Советская, 63/4.
Курганский государственный университет.