

«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра конструирования и технологии электронно-  
вычислительных средств

Методические указания по выполнению курсовой работы  
по дисциплине  
«Современные проблемы науки и производства»  
направления подготовки магистров 210200.68

Курск 2010

Составитель А.В. Кочура, О.Г. Бондарь

УДК 681.325.5:621.382.049.77(075.8)

Рецензент

Кандидат технических наук, доцент кафедры вычислительной техники *В.И. Иванов*

Методические указания к курсовой работе по дисциплине «Современные проблемы науки и производства» / ЮЗГУ; Сост. А.В. Кочура, О.Г. Бондарь. Курск, 2010, 24 с.

Излагаются методические указания по выполнению курсовой работы по курсу «Современные проблемы науки и производства». Приведены 50 вариантов заданий.

Предназначены для студентов, обучающихся по направлению подготовки магистров 210200.68 «Конструирование и технология электронных вычислительных средств»

Ил.1. Табл. 5. Библиогр.: 5 назв.

Текст печатается в авторской редакции

ИД № 06430 от 10.12.01.

Подписано в печать

Формат 60x84 1/16. Печать офсетная

Усл. печ. л.

Уч.-изд. л.

Тираж 50 экз.

Заказ

Бесплатно.

Курский государственный технический университет.

305040 Курск, ул. 50 лет Октября, 94

## Содержание

ВВЕДЕНИЕ .....	6
1..... Общая структура проектируемой системы .....	6
2. Алгоритм управления .....	7
3. Задание на курсовую работу .....	8
4. Этапы курсовой работы .....	8
4.1. Расшифровка и анализ задания .....	9
4.2. Разработка уточненной структурной схемы выбранной системы.....	10
Рис. 1. Обобщенная структурная схема УМПС.....	12
4.3. Разработка процессорного модуля.....	12
4.4. Разработка подсистем памяти .....	13
4.5. Разработка подсистем ввода/вывода, прерываний.....	14
4.6. Разработка программного обеспечения.....	16
4..... Оформление курсовой работы .....	17
Библиографический список.....	19
6. Варианты заданий .....	20

## СПИСОК ИСПОЛЬЗОВАННЫХ СОКРАЩЕНИЙ

- БИС – большая интегральная схема
- ВУ – внешнее устройство
- ВТ – вычислительная техника
- ИНД – индикация
- ЗУ – запоминающее устройство
- МП – микропроцессор
- МПС – микропроцессорная система
- ОЗУ – оперативное запоминающее устройство
- ОУ – объект управления
- ПЗУ – постоянное запоминающее устройство
- ПО – программное обеспечение
- УВВ – устройство ввода/вывода
- УМПС – управляющая микропроцессорная система
- ША, ШД, ШУ – шины адреса, данных и управления
- EEPROM – (Electrically Erasable Programmable Read-Only Memory) электрически стираемое ПЗУ
- EPROM – (Electrically Programmable Read- Only Memory) электрически программируемое ПЗУ
- RAM – (Random Access Memory) память произвольного доступа - ОЗУ
- RS232C – (Recommended Standard) интерфейс обмена данными через последовательный коммуникационный порт
- SPI – (Serial Peripheral Interface) - полнодуплексный скоростной синхронный трёхпроводной интерфейс



## ВВЕДЕНИЕ

Целью выполнения курсовой работы по дисциплине “Современные проблемы науки и производства” является:

- обобщение, закрепление и углубление знаний по данной дисциплине и дисциплинам, связанным с проектированием средств ВТ;
- отработка методик проектирования микропроцессорных систем и их программирования с применением САПР;
- формирование навыков разработки и оформления текстовой и графической техдокументации;

Содержанием курсовой работы является разработка микропроцессорной управляющей системы, реализующей заданные взаимодействия с ОУ и разработка программных средств системы, обеспечивающих выполнение алгоритма управления.

### 1. Общая структура проектируемой системы

Проектируемая управляющая микропроцессорная система (УМПС) включает в себя:

- 8-разрядный микропроцессор, микроконтроллер или микроЭВМ;
- ПЗУ и ОЗУ, при необходимости;
- устройство параллельного ввода дискретной информации с объекта управления (ОУ);
- устройство параллельного вывода дискретной информации на (ОУ);
- канал для последовательной связи с персональным компьютером (RS232C);
- пульт оператора.

## 2. Алгоритм управления

УМПС реализует следующий алгоритм управления (рис. 1): С ОУ принимается поток значений входных слов  $X_i$  и размещается в кольцевом буфере на 17 значений в памяти УМПС. При приёме 16 значений производится вычисление среднего арифметического значения  $X = (X_1 + X_2 + \dots + X_{16})/16$ . В дальнейшем, при приёме каждого нового значения, оно записывается в кольцевой буфер вместо самого старого значения, а среднее значение пересчитывается по последним 16 значениям. В зависимости от значений  $X$ , вычисляется управляющее воздействие  $Y$ :

$$\begin{aligned}
 Y &= 00H \text{ при } X < G1; \\
 Y &= \Phi_1(X) \text{ при } G1 \leq X < G2; \\
 Y &= \Phi_2(X) \text{ при } G2 \leq X \leq G3; \\
 Y &= 0FFH \text{ при } X > G3.
 \end{aligned} \tag{1}$$

В выражении (1) константы  $G_1, G_2, G_3$  и вид функций  $\Phi_1, \Phi_2$  определяется заданием. Во всех вариантах все переменные и константы рассматриваются как 8-разрядные двоичные (2-разрядные шестнадцатеричные) целые без знака, причем:

$$0 < G_1 < G_2 < G_3 < 0FFH.$$

Для случаев  $X < G_1$  и  $X > G_3$  необходимо выдать на пульт оператора сигнал <Авария> – звуковой тон частотой 500 Гц или включить мигание специального светового индикатора (лампы накаливания, питающейся от осветительной сети переменного тока напряжением 220В частотой 50 Гц) с частотой 2 Гц.

По запросу с пульта оператора следует выдать на его индикацию значения  $X_{\min}, X_{\max}, X(\text{среднее}), Y$  для цикла управления, предшествующему текущему. Выдача этой информации должна осуществляться в *режиме прерывания программы*. Константы, обеспечивающие индивидуальную настройку, должны храниться в энергонезависимой памяти (EEPROM или RAM с буферным источником питания).

### 3. Задание на курсовую работу

Задание на курсовую работу включает в себя:

- семейство 8-разрядных микропроцессоров, микроконтроллеров или микроЭВМ, на которых следует реализовать УМПС;
- типы БИС, на которых реализуется ПЗУ и ОЗУ. Объем ОЗУ и ПЗУ студент выбирает самостоятельно исходя из объема программ и данных, используемых в УМПС и с учетом удобства реализации на заданном типе БИС. При наличии в микроконтроллере ПЗУ и/или ОЗУ достаточного объёма, необходимость в установке дополнительных ЗУ данного типа отпадает;
- тип внешних EEPROM, RAM для хранения констант настройки;
- набор констант G1, G2, G3, определяющих способ вычисления выходного воздействия Y;
- значения булевых функций  $\Phi_1(X)$ , определяющих зависимость разрядов выходного слова  $Y(7:0)$  от разрядов среднего значения  $Y(7:0)$ ;
- значение функции  $\Phi_2(X)$ , определяющей зависимость выходного слова Y от X и константы K, вводимой с пульта оператора;
- способ подачи аварийного сигнала.

Основные элементы задания сведены в таблицы 1.1 ... 1.4. Номер строки соответствует номеру группы и порядковому номеру студента по журналу группы. В строках табл. 1.1 ... 1.4 есть ссылки на строки таблиц 2.3.4, в которых определены наборы констант G(1..3) и функции  $\Phi_1$ ,  $\Phi_2$ .

### 4. Этапы курсовой работы

Содержание этапов и результаты выполнения курсовой работы

В курсовой работе предлагается следующий состав и порядок выполнения этапов:

1. Расшифровка и анализ задания.
2. Разработка интерфейса и уточненной структурной схемы.

3. Разработка процессорного модуля.
4. Разработка подсистемы памяти (ОЗУ + ПЗУ + энергонезависимая память).
5. Разработка подсистемы ввода/вывода, прерываний.
6. Разработка программного обеспечения.
7. Оценка системы.
8. Оформление работы.

Выполнение этапов 3 и 4 осуществляется одновременно, т.к. в большинстве случаев основные потребности в памяти покрываются внутренними ресурсами микроЭВМ или микроконтроллера.

#### **4.1. Расшифровка и анализ задания**

По номеру группы и порядковому номеру в списке группы следует выбрать строку табл. 1.1 ... 1.4, а также указанные в этой строке строки табл. 2,3,4, выписать их содержимое. Изучить общую структуру системы (рис. 1) и проанализировать возможность ее реализации на заданных БИС. Если в процессе предварительного анализа будут замечены противоречия в задании, например – несоответствие параметров БИС памяти и базовой БИС, то по согласованию с руководителем задание следует частично изменить.

На этом этапе осуществляется сбор исходных данных для курсовой работы.

Прежде всего, необходимо собрать сводную информацию по микроконтроллерам заданного семейства. Эта информация должна включать в себя: наименование модели, объём всех внутренних видов памяти, объём и возможность подключения внешней памяти, количество линий цифрового ввода/вывода, перечень встроенных аппаратных средств (таймеры, АЦП, ЦАП), типы поддерживаемых интерфейсов, производительность, стоимость. Данные необходимо свести в таблицу, пример которой приведен в учебно-методическом комплексе «Схемотехника» на сайте кафедры <http://www.kurskelectronic.ru>.

Кроме того, необходимо найти справочную информацию на БИС, используемые в работе. Эта информация должна содержать: условное графическое обозна-

чение БИС, функциональное назначение выводов, типы корпусов, электрические характеристики (напряжение питания, потребляемые токи, уровни входных и выходных сигналов, входные и выходные токи, допустимые ёмкостные нагрузки, ёмкости входов и выходов), динамические характеристики и временные диаграммы, логическое описание интерфейса (в том числе протоколы, форматы). Подбор источников информации необходимо начинать с сервера кафедры К и Т ЭВС.

**Результатом этапа являются:**

1. подписанное студентом техническое задание,
2. список литературы, ссылок на документы в файловой форме (с демонстрацией самих документов),
3. таблица параметров семейства микроконтроллеров,
4. заранее заготовленный лист вопросов на консультацию.

#### **4.2. Разработка уточненной структурной схемы выбранной системы**

На рис. 1 приведен вариант обобщенной структурной схемы УМПС, построенной по магистрально-модульному принципу. На этапе ее уточнения следует:

- изобразить магистраль, перечислив все входящие в нее линии;
- изобразить все модули, входящие в состав МПС, в том числе:
- полный набор УВВ, включая пульт оператора, контроллер прерываний, селекторы устройств и т.п.;
- показать состав линий магистрали, поступающих на каждый модуль и связь между модулями;
- оценить объём памяти программ и памяти данных;
- произвести предварительное распределение адресного пространства системы, назначив адреса ОЗУ, ПЗУ, энергонезависимого ЗУ и регистрам всех ВУ. Распределение адресов может уточняться при разработке ПО.

Разрабатываемая система может иметь и другую организацию, например, радиальную. В этом случае каждое устройство подключается к микроэвм через свои индивидуальные интерфейсы (параллельные или последовательные).

Наиболее трудоёмкой частью этапа является оценка необходимого объёма памяти программ и данных. Эта оценка проводится на основе анализа прототипных систем. При отсутствии близкого прототипа подбирают несколько систем имеющих сходные с проектируемой системой подсистемы. Большое количество таких систем представлено на сайтах производителей микропроцессоров и микроконтроллеров. При оценке общий объём определяют суммированием объёмов соответствующих подпрограмм прототипов и вводя коэффициент запаса в пределах 1,5-2.

При распределении адресного пространства не следует стремиться к чрезмерной детализации. Прежде всего, необходимо выяснить общую структуру адресного пространства используемого семейства микроконтроллеров. Обычно выделяется область программы, ОЗУ данных, область регистров общего назначения, область управляющих регистров и область ввода-вывода. Отдельно выделяется область энергонезависимой памяти. Некоторые из этих областей могут совмещаться.

Важно определить особенности адресного пространства семейства микроконтроллеров. Возможность подключения внешней памяти определяет наличие соответствующего интерфейса и реализации магистрально-модульного принципа организации системы. Отсутствие расширяемости может потребовать эмуляции параллельного интерфейса ввода-вывода на портах микроконтроллера. В памяти программ следует обозначить зоны, резервируемые для подсистемы прерываний. Программа пользователя не должна размещаться в этих зонах.

**Результатами этапа должны быть:**

1. уточненная структурная схема УМПС,
2. карта распределения адресного пространства,
3. дополнения к списку литературы со ссылками на прототипные проекты.

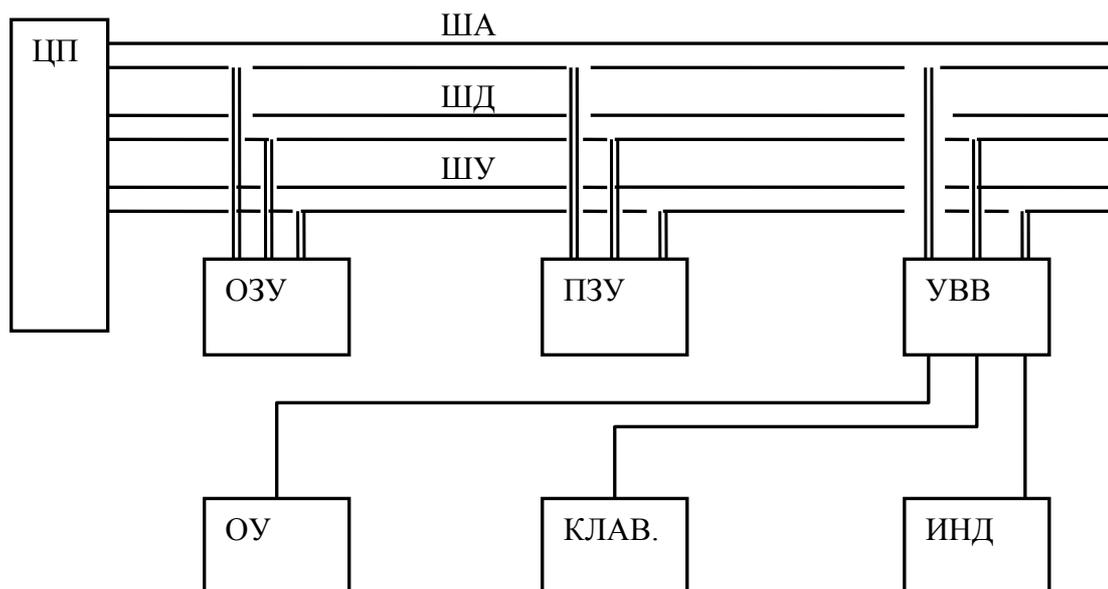


Рис. 1. Обобщенная структурная схема УМПС

### 4.3. Разработка процессорного модуля

Процессорный модуль должен включать в себя базовую БИС и вспомогательные схемы, обеспечивающие её работоспособность. К ним относятся: генератор тактовых импульсов, времязадающие цепочки, супервизор питания или устройство рестарта, буферные регистры и т. п.

**Процедура выбора предполагает использование полного перечня микроэвм (контроллеров или микропроцессоров) заданного семейства с их характеристиками.**

При выборе базовой БИС **заданного семейства** следует руководствоваться наличием или отсутствием встроенных ОЗУ и ПЗУ и их объёмом, возможностью расширения, количеством портов, аппаратной поддержки дополнительных интерфейсов (RS232, SPI, I2C...) и стоимостью решения. Следует учитывать, что стоимость микроконтроллеров со встроенной памятью может существенно превышать суммарную стоимость необходимых компонентов при использовании отдельных БИС памяти.

Выбор можно осуществить последовательным исключением из полного перечня микроэвм не удовлетворяющих требованиям системы по каждому из параметров. Из оставшегося списка выбирается вариант наименьшей стоимости.

При проектировании процессорного модуля следует определить функции каждого дополнительного устройства и требования к техническим параметрам.

Для генератора тактовых импульсов, например, такими параметром являются частота и её стабильность. Частоту выбирают исходя из требуемого быстродействия и энергопотребления (не всегда следует стремиться к максимальному значению). Ограничением может служить и быстродействие используемых периферийных интегральных схем или требования к конкретным значениям. Для определения необходимой стабильности частоты, следует проанализировать все подсистемы и выявить ту, которая будет лимитировать величину нестабильности.

**Результатом выполнения курсовой работы на этом этапе должны быть:**

1. перечень вспомогательных устройств процессорного модуля с перечнем выполняемых ими функций,
2. результаты анализа подсистемы лимитирующей тактовую частоту процессора (если имеется ограничение)
3. результаты анализа требований подсистем к стабильности частоты,
4. функциональная схема процессорного модуля с указанием всех входных и выходных линий.

#### **4.4. Разработка подсистем памяти**

Подсистема памяти разрабатываемой УМПС включает в себя модуль ОЗУ, модуль ПЗУ и модуль энергонезависимой памяти, доступ к которым осуществляется по системной магистрали или специальным интерфейсным линиям. Обычно потребности в памяти программ и оперативной памяти в небольших системах построенных на базе микроконтроллеров покрываются за счёт внутренних ресурсов микроконтроллера. Установка внешней энергонезависимой памяти является обя-

зательным требованием, независимо от наличия таковой в составе микроЭВМ или микроконтроллера.

Исходные данные для разработки подсистемы памяти определены в табл. 1.1 ... 1.4, где указаны типы применяемых микросхем. Требуется разработать функциональные схемы ОЗУ и ПЗУ с учетом особенностей функционирования заданных микросхем и схему селектора адресов. Селектор адресов вырабатывает сигналы  $CS_i$  для модулей памяти с учётом распределения адресного пространства, принятого в п. 4.2. Схема селекции может включать в себя селекторы внешних устройств (ВУ), если память и ВУ функционируют в едином адресном пространстве.

Для модуля энергонезависимой памяти следует определить тип и количество необходимых линий интерфейса, а также осуществить необходимое согласование интерфейсных линий памяти с основной БИС. Необходимо определить функции микроконтроллера при обмене и выделить из них реализуемые аппаратно и программно

**Результатом построения подсистемы памяти должны быть:**

1. функциональная схема подсистемы памяти и схема селектора адресов,
2. перечень аппаратных и программных интерфейсных функций.

#### **4.5. Разработка подсистем ввода/вывода, прерываний**

В разрабатываемой МПС должна быть предусмотрена следующая номенклатура ВУ:

- устройство ввода информации  $X_i(7:0)$  с ОУ;
- устройство вывода информации  $Y_i(7:0)$  на ОУ;
- устройство ввода константы  $K(7:0)$  с пульта оператора;
- устройство вывода значения  $X_{min}(7:0)$  на пульт оператора;
- устройство вывода значения  $X_{max}(7:0)$  на пульт оператора;
- устройство вывода значения  $X_{cp}(7:0)$  на пульт оператора;
- устройство вывода значения  $Y(7:0)$  на пульт оператора;

- устройство вывода сигнала <АВАРИЯ>;
- устройство вывода по последовательному каналу RS232;
- устройство ввода по последовательному каналу RS232.

Каждое из перечисленных выше устройств должно получить свой адрес в едином адресном пространстве или в пространстве ввода/вывода.

Обмен с ОУ должен осуществляться в асинхронном режиме, для чего рекомендуется использовать порты микроконтроллера или внешний контроллер параллельного обмена K580BB55 или аналогичный ему по функциям.

Обмен с пультом оператора может осуществляться в синхронном режиме, поэтому для обмена можно использовать как контроллеры типа K580BB55, так буферные регистры или порты микроконтроллера.

Обязательным требованием к аппаратуре поддержки интерфейса RS232 является гальваническая развязка. Её целесообразно реализовать с применением специализированных устройств гальванической развязки iCoupler или оптронов.

#### **Подсистема прерываний должна обеспечить реакцию на четыре запроса:**

INT1 – запрос с пульта оператора на выдачу значений  $X_{min}$ ,  $X_{max}$ ,  $X$ ,  $Y$ ;

INT2 – запрос от ПК на обмен по последовательному каналу;

INT3 – запрос от ОУ на ввод  $X_i$ ;

INT4 – запрос от ОУ на вывод  $Y$ .

Запрос INT1 обладает высшим приоритетом и должен удовлетворяться всегда. Запросы INT2 также должен удовлетворяться всегда. Запросы INT3 и INT4 могут обслуживаться только на определенных участках программы управления.

При разработке подсистемы прерываний следует перечислить все её функции, выбрать и описать способ реализации каждой из функций. Обратит внимание на способы формирования сигналов запроса прерываний устройствами. Продумать способы исключения повторных вызовов при удержании ключей клавиатуры в нажатом состоянии.

При разработке подсистемы прерываний следует максимально использовать возможности имеющейся радиальной системы прерывания микроконтроллера.

При ограниченных возможностях микроконтроллера можно идентифицировать источники прерывания путем программного опроса.

Схему подсистемы прерываний можно объединить со схемами подсистемы ввода/вывода.

**Результатом проектирования подсистемы ввода/вывода являются:** уточнённая карта распределения адресного пространства ввода/вывода или фрагмента единого адресного пространства, отведенного под ввод/вывод,

1. функциональная схема УВВ ОУ,
2. функциональная схема канала последовательного ввода вывода,
3. функциональная схема пульта оператора,
4. функциональная схема устройства оповещения об аварии,
5. функциональная схема подсистемы прерываний,
6. перечень аппаратно и программно реализуемых функций каждой подсистемы.

#### **4.6. Разработка программного обеспечения**

В рамках разработки ПО в курсовой работе требуется:

- разработать общую структуру ПО МПС, включив в него программу инициализации микроконтроллера, основную управляющую программу, подпрограммы обслуживания прерываний, подпрограмму обмена по интерфейсу RS232; определить функции, входные и выходные параметры подпрограмм;

- разработать детализированные алгоритмы основных подпрограмм (управления, обмена информацией по интерфейсу RS232, обслуживания прерываний);

- разработать фрагмент одной из программ (по согласованию с руководителем) на языке ассемблера.

Рекомендуется нижеследующий порядок разработки программного обеспечения:

- собрать и представить в виде таблицы полный перечень функций каждой подсистемы с разбиением на реализуемые аппаратно и программно,
- представить укрупнённый алгоритм в виде нескольких несвязанных алгоритмов, выполняющих функции в фоновом режиме и обслуживающих события (прерывания),
- определить способы взаимодействия отдельных алгоритмов (методы синхронизации процессов),
- предусмотреть средства исключения взаимной и самоблокировки процессов,
- детализировать алгоритм до функций, выполняемых каждой подсистемой (функции подсистемы могут попадать в разные части общего алгоритма - фоновую или событийную),
- согласовать с руководителем модуль, выбранный для программирования,
- выполнить программирование на языке ассемблера с использованием аппаратно-программных комплексов разработки,
- произвести отладку модуля в режиме симуляции.

#### **Результаты этапа:**

сводная таблица функций,  
алгоритм функционирования,  
листинг программного модуля.

### **4. Оформление курсовой работы**

Пояснительная записка должна содержать:

- титульный лист;
- оглавление с указанием страниц;
- задание на курсовую работу;
- структурную схему УМПС, краткое описание состава и назначения основных элементов системы;

- карту распределения адресного пространства УМПС: программа, данные, стек, константы и т.п.;
- процедуру и результаты оценки объёма программного кода и ОЗУ;
- описание процессорного модуля, перечень функций и обоснование основных характеристик окружения;
- описание подсистемы памяти, в т.ч. особенностей реализации интерфейса с энергонезависимой памятью;
- описание подсистемы ввода/вывода и прерываний;
- описание пульта оператора;
- описание общей структуры ПО МПС;
- описание алгоритмов основных программ;
- листинг программы (фрагмент);
- оценку быстродействия, энергопотребления и аппаратных затрат;
- перечень литературы.

Примерный объем пояснительной записки – 20 – 25 стр.

Графическая часть курсовой работы должна содержать:

- структурную схему УМПС с указанием всех информационных связей и управляющих сигналов;
- функциональную схему микропроцессорного модуля и подсистемы памяти;
- функциональную схему подсистемы ввода/вывода и пульта оператора;
- граф-схемы алгоритмов.

## 5. Указания

1. Структурно-функциональную схему системы рекомендуется выполнять в Visio 2003.
2. Схемы функциональные электрические выполнять в среде OrCAD 9.2.
3. Оглавление пояснительной записки формировать в режиме структуры редактора Microsoft Word.
4. Список литературы создавать в режиме концевой сноски.

### Библиографический список

1. Евстифеев, А.В. Микроконтроллеры AVR-семейств Tiny и Mega фирмы “AT-MEL” [Текст] /А.В. Евстифеев. - М.: Издательский дом “Додэка-XXI”, 2004. - 560 с.
2. Голубцов, М.С. Микроконтроллеры AVR [Текст]: от простого к сложному/ М.С. Голубцов. - М.: СОЛОН-Пресс, 2003. -288 с. Андриенко Ю., Комаров А., Труш А. Микроконтроллер AT90S2313 фирмы Atmel. – Новокузнецк, 2002. - 39 с. (доступна в электронной форме на сервере кафедры).
3. Тавернье, К. PIC-микроконтроллеры. Практика применения [Текст]: пер. с фр./ К. Тавернье. - М.: ДМК Пресс, 2003. 272 с.Предко М. Справочник по PIC-микроконтроллерам. Пер. с англ.-М.: Издательский дом «Додэка-XXI», 2002.- 512 с.
4. Рафикузаман, М. Микропроцессоры и машинное проектирование микропроцессорных систем [Текст]: В 2-х кн. Кн.1 / М. Рафикузаман. - М.: Мир, 1988. - 312 с.
5. Семейство микроконтроллеров Z8. Руководство пользователя [Текст]: пер. с англ. Zilog Inc., 2002. (электронная версия на сервере кафедры).

## 6. Варианты заданий

Таблица 1.1

№ вар.	БИС МП	БИС ОЗУ	БИС EPROM	БИС ЕЕ- PROM	Конст. Таб.2	Ф1 Таб.3	Ф2 Таб.4	Авария
1.1	AVR	KP537PY10	Am27C512	CAT25C02	4	11	3	звук
1.2	Z86	537PY6	28C17A	AK6002A	16	20	12	свет
1.3	PIC16XXX	K132PY11	KP568PE3	AK6420A	12	13	11	звук
1.4	AVR	K132PY11	27C16	93C45A	1	19	8	свет
1.5	Z86	537PY23	Am27C010	AT24C01	3	10	7	звук
1.6	PIC16XXX	K581PY5	KP556PT16	AT34C02	4	19	3	свет
1.7	AVR	537PY6	28C16A	CLA25C040	7	14	5	звук
1.8	Z86	K132PY2	NM27C128	CAT24C021	16	13	9	свет
1.9	PIC16XXX	KM132PY5	Am27C64	NM25C040	20	11	12	звук
1.10	AVR	KP537PY8	NM27C256	AT24C02	8	9	15	свет
1.11	Z86	537PY9	KP556PT18	AT93C46	18	6	14	звук
1.12	PIC16XXX	KP565PY2	28C04A	CAT24WC03	13	14	4	свет
1.13	AVR	KP565PY2	KP556PT6	AT25040	19	8	1	звук
1.14	Z86	KP537PY8	NM27C512	AT59C11	10	18	17	свет
1.15	PIC16XXX	537PY23	K573PФ2	SLA24C04	19	12	20	звук
1.16	AVR	KP537PY10	28C64A	25AA040	14	1	13	свет
1.17	Z86	537PY23	K573PФ5	S-29L294A	13	3	6	звук
1.18	PIC16XXX	K581PY5	S-2812A	AT25010	11	4	18	свет
1.19	AVR	584PY5	NMC27C64	AT24C08	9	7	21	звук
1.20	Z86	K541PY3	S-2817A	AT25020	8	16	10	свет
1.21	PIC16XXX	K134PY6	NM27C256	AT24C04	19	14	19	звук
1.22	AVR	K155PY7	28C64A	S24C01A	14	8	13	свет
1.23	Z86	–	S-2864B	NM24C05	13	18	16	звук
1.24	PIC16XXX	KP537PY8	KP556PT15	25AA320	11	13	14	свет
1.25	AVR	537PY30	S-2860B	AT59C22	9	12	12	свет

Таблица 1.2

№ вар.	БИС МП	БИС ОЗУ	БИС EPROM	БИС ЕЕ- PROM	Конст. Таб.2	Ф1 Таб.3	Ф2 Таб.4	Авария
2.1	Z86	K155PY5	S-2860B	NM93C46	3	12	15	звук
2.2	PIC16XXX	584PY5	28C64A	AK93C45A	4	9	6	свет
2.3	AVR	KP537PY10	M27C256	S-29255A	7	3	12	звук
2.4	Z86	K541PY3	S-2812A	AK6002A	16	4	9	свет
2.5	PIC16XXX	K134PY6	M27C512	AT24C01	20	7	3	звук
2.6	AVR	K155PY7	S-2864B	CAT64LC010	8	16	4	свет
2.7	Z86	KP537PY10	KP556PT16	CLA 25C160	18	20	7	звук
2.8	PIC16XXX	KP537PY8	K573PФ2	AK6420A	13	8	16	свет
2.9	AVR	K132PY2	KP556PT15	NM25C041	19	18	20	звук
2.10	Z86	KM132PY5	28C04A	AT24C02	10	13	8	свет
2.11	PIC16XXX	KP537PY8	Am27C64	TC89101P	19	19	18	звук
2.12	AVR	KP537PY10	M27C128	AT59C13	14	3	13	свет
2.13	Z86	KP565PY2	28C17A	93AA66	13	4	19	звук
2.14	PIC16XXX	KP565PY2	M27C128	AK6440A	11	7	10	свет
2.15	AVR	KP537PY8	K573PФ5	S24C01A	9	16	19	звук
2.16	Z86	KP537PY8	Am27C512	AT59C11	6	20	14	свет
2.17	PIC16XXX	KP537PY10	28C16A	25AA640	1	8	13	звук
2.18	AVR	K155PY5	S-2812A	AT24C04	14	18	11	свет
2.19	Z86	K155PY5	M27C128	AT25080	5	13	9	звук
2.20	PIC16XXX	584PY5	S-2860B	AT24C08	17	19	6	свет
2.21	AVR	KP537PY30	S-2812A	AT34C02	12	10	1	звук
2.22	Z86	K541PY3	M27C128	CAT24C163	15	19	14	свет
2.23	PIC16XXX	K134PY6	Am27C512	AT59C22	4	14	2	звук
2.24	AVR	M1821PY55	27C16	AK6004A	7	13	5	свет
2.25	Z86	KP537PY10	28C04A	CLA 25C080	1	11	17	свет

Таблица 2. Константы

№строки	G1 (HEX)	G2 (HEX)	G3 (HEX)
1	1	9	2F
2	16	6A	BC
3	22	80	AE
4	23	5F	99
5	0A	88	AF
6	32	69	99
7	14	7C	CA
8	30	70	C0
9	8	56	98
10	14	77	BB
11	8	93	F4
12	11	88	DD
13	4A	7D	F0
14	17	90	EA
15	1	39	9A
16	16	86	B4
17	23	97	D6
18	5	5B	E2
19	16	79	93
20	29	6B	FC

Таблица 3. Функция Ф1

№ строки	Значение бита функции							
	y7	y6	y5	y4	y3	y2	y1	y0
1	$\overline{X7}$	$\overline{X6}$	X5	$\overline{X4}$	$\overline{X3}$	X2	$\overline{X1}$	$\overline{X1}$
2	$\overline{X7}$	$\overline{X6}$	X5	X4	$\overline{X3}$	$\overline{X2}$	X1	X2
3	X0	X1	X2	X3	X4	X5	X6	$\overline{X3}$
4	$\overline{X7}$	$\overline{X7}$	X7	X7	X0	X0	$\overline{X0}$	X4
5	X7	$\overline{X6}$	$\overline{X5}$	X4	0	0	0	X5
6	$\overline{X5}$	$\overline{X4}$	X3	X2	$\overline{X1}$	X0	$\overline{X7}$	$\overline{X6}$
7	$\overline{X0}$	X0	$\overline{X0}$	X0	$\overline{X3}$	X3	$\overline{X3}$	X7
8	1	1	1	1	$\overline{X4}$	X3	$\overline{X2}$	X2
9	X7	0	$\overline{X5}$	1	$\overline{X3}$	0	$\overline{X1}$	X1
10	0	0	$\overline{X5}$	X4	1	1	$\overline{X1}$	X2
11	X2	X5	X7	X3	X1	X0	X6	X4
12	X2	$\overline{X2}$	X2	$\overline{X2}$	X6	$\overline{X6}$	$\overline{X6}$	X6
13	$\overline{X3}$	X2	$\overline{X1}$	X0	$\overline{X7}$	$\overline{X6}$	$\overline{X5}$	X4
14	X7	$\overline{X7}$	$\overline{X7}$	X7	X3	X2	X1	X0
15	0	0	0	0	0	$\overline{X7}$	X1	X0
16	1	0	X5	0	1	X2	X1	0
17	$\overline{X0}$	$\overline{X2}$	$\overline{X4}$	$\overline{X6}$	X7	X5	X3	X1
18	X4	$\overline{X4}$	X4	$\overline{X2}$	X2	$\overline{X2}$	0	1
19	X2	X3	X4	X5	X6	X7	X0	X1
20	0	0	$\overline{X7}$	X6	$\overline{X5}$	0	1	1

Таблица 4. Функция Ф2

№ строки	Функция Ф2(X, K)
1	$\max(2*X, K)$
2	$\min(X, K)$
3	$X + K$
4	$K - X$
5	$X - K$
6	$\text{Min}(X - K, X/2)$
7	$\max(X, K)$
8	$K + \min(X, K)$
9	$X - \max(X, K)$
10	$K + \max(X, K)$
11	$K - \min(X, K)$
12	$\max(X, K) - 1$
13	$\max(X, K) + 2*K$
14	$4*X + K$
15	$\min(X, K) + 3$
16	$5FH - \min(X, K)$
17	$3 + K - X/2$
18	$2*\min(X, K)$
19	$FFH - \min(X, K)/2$
20	$X + \min(X, K)/4$

## Критерии оценки курсовой работы

При оценивании проекта принимаются во внимание следующие факторы.

**Соответствие техническому заданию** (соответствие графических документов перечню в задании; наличие в пояснительной записке описания всех подсистем, подлежащих разработке в соответствии с заданием; соответствие ограничениям и указаниям по выбору элементной базы; соответствие заданию основных выполняемых функций). Работа несоответствующая этому критерию возвращается на доработку

**Работоспособность системы.** Под ней следует понимать возможность выполнения базовых функций (в системе отсутствуют условия для катастрофических отказов по причине грубых схемотехнических ошибок, или условия для нефункциональности основных подсистем).

**Функциональная полнота.** Полученная система должна содержать аппаратные и программные (алгоритмические) средства, в совокупности обеспечивающие решение всех задач, возлагаемых на систему.

**Степень обоснованности аппаратных решений.** Проверяется наличие *обоснованных* требований к характеристикам и параметрам каждой подсистемы, альтернативных вариантов построения, методики отбора вариантов.

**Качество проработки алгоритма функционирования.** Оценивается полнота (избыточность) перечня задач решаемых системой управления; наличие вариантов их выполнения и взаимодействия с выделением критических характеристик и параметров для каждого варианта; наличие сравнительного анализа и выбора методов выполнения и взаимодействия задач. Оценивается полнота перечня функций и соответствие их аппаратному оформлению для каждой из подсистем. Оценивается корректность граф-схемы алгоритма и соответствие ЕСПД.

**Качество системы.** Оценивается эффективность принятых решений. При этом проверяется обоснованность разделения функций на аппаратные и программные с учетом объемов производства (для единичных систем оптимальны аппаратные решения, позволяющие сократить время разработки и создания, а для крупносерийной продукции программная реализация функций позволит существенно снизить стоимость) минимальность аппаратных затрат, эффективность алгоритмов по критериям объём, быстродействие.

**Качество оформления курсовой работы.** При этом учитывается:

- применение средств автоматизации работ;
- соответствие правилам оформления технической документации по ГОСТ;
- чёткость структуры документа, наличие реферата, оглавления, нумерации страниц, введения и заключения;
- краткость, полнота, последовательность и ясность изложения материала, отсутствие в записке неиспользуемой и неактуальной информации;
- наличие грамматических ошибок (записка должна быть прочитана и отредактирована автором).