

Пермский государственный технический университет
Кафедра Информационных технологий
и автоматизированных систем

Щемелева Т.К.

**Методические указания
к выполнению курсовой работы по дисциплине
«Микропроцессорные системы»
Тема: Разработка МП – системы.**

Пермь-2006

681.3

М

Щемелева Т.К. **Методические указания к выполнению курсовой работы по дисциплине «Микропроцессорные системы» для студентов специальности 220100 – ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ, КОМПЛЕКСЫ, СИСТЕМЫ И СЕТИ.**

Составитель: Щемелева Т.К. Пермь, Перм. гос. техн. ун-т 2006г., 26

Методуказания рассмотрены и утверждены на заседании кафедры ИТАС ПГТУ « 2 »_февраля 2006г. Протокол №___

Содержание

1. <i>Методология процесса проектирования МП – системы</i>	4
1.1. <i>Концепции и этапы проектирования МП- системы</i>	4
1.2. <i>Факторы, влияющие на специфику проектирования</i>	5
1.3. <i>Этапы проектирования с помощью САПР</i>	6
1.4. <i>Сопряженное проектирование и сопряженная верификация</i>	8
1.5. <i>Проектирование типовой конфигурации МП систем</i>	9
1.5.1. <i>Типовые конфигурации</i>	9
1.5.2. <i>Основные этапы процедуры проектирования</i>	10
1.6. <i>Средства и методы проектирования и автономной отладки аппаратных средств МП – системы</i>	12
1.6.1. <i>Выбор семейства МП/МК и стандартной периферии</i>	12
1.6.2. <i>Аппаратные средства отладки</i>	13
1.7. <i>Средства и методы разработки программного обеспечения</i>	13
1.8. <i>Программные средства поддержки проектирования и отладки систем</i>	14
2. <i>Методические указания к разработке МП – системы</i>	15
2.1. <i>Исследовательский раздел</i>	15
2.1.1. <i>Исследование предметной области курсовой работы</i>	15
2.1.2. <i>Составление технического задания на разработку МП – системы</i>	15
2.2. <i>Конструкторский раздел</i>	15
2.2.1. <i>Системное проектирование</i>	15
2.2.2. <i>Структурно – алгоритмическое проектирование</i>	16
2.2.3. <i>Функционально – алгоритмическое проектирование</i>	17
2.2.4. <i>Анализ функционирования МП – системы с помощью временной диаграммы</i>	17
3. <i>Правила оформления курсовой работы (по ГОСТ 7.32-2000)</i>	19
4. <i>Примерные варианты заданий</i>	
<i>Список рекомендуемой литературы</i>	24

1. Методология процесса проектирования МП – системы

1.1. Общие характеристики процесса проектирования МП- системы

Основные концепции проектирования МП – системы изложены в [1]. Проектирование - разработка технической документации, позволяющей изготовить устройство с заданным функционированием, с заданными свойствами и в заданных условиях. В основе *стратегии проектирования* лежит *функциональная декомпозиция*. Для системы в целом и ее блоков используется концепция «*черного ящика*».

Декомпозиция функций блоков выполняется до тех пор, пока не получатся типовые функции, каждая из которых может быть реализована элементами выбранного уровня иерархии.

Процесс проектирования -многоуровневый, многошаговый и итерационный, с возвратами назад и пересмотром ранее принятых решений.

Последовательная декомпозиция проекта на отдельные фрагменты (с определением функций каждого фрагмента и его интерфейса) не зависит от иерархического уровня проектирования и характерна для разработки широкого класса цифровых устройств, начиная от устройства целиком и кончая проектированием отдельных БИС/СБИС. Такая методология проектирования отображает процесс проектирования «сверху-вниз»: от технического задания до электрических схем, файлов прошивки ПЗУ и конфигурации программируемых приборов, а также конструкции устройства в целом.

Следует отметить, что приведенное выше наглядное описание процесса проектирования относится к каждому уровню проектирования. При этом декомпозиция заканчивается при получении типовых функций, соответствующих выбранному уровню иерархии. Так, на верхнем уровне (при многоплатной реализации) декомпозиция заканчивается при представлении проекта в виде отдельных плат, на следующем уровне - в виде отдельной платы (типового элемента замены), еще ниже декомпозиция осуществляется до реализации функций при помощи той или иной микросхемы. А при ориентации на программируемые (разрабатываемые) пользователем микросхемы процедура декомпозиции осуществляется уже для этой микросхемы в соответствии с составом функциональных библиотек программируемых БИС/СБИС.

С учетом возможностей современных систем автоматизации проектирования (САПР) проектирование может считаться законченным после верификации проекта в целом, когда завершена отладка готового изделия.

Различие теоретической базы и понятийного аппарата, используемых на разных стадиях проектирования, приводит к тому, что традиционным является разбиение процесса проектирования цифровых устройств на следующие этапы:

- системного проектирования;
- структурно-алгоритмического проектирования;
- функционально-логического проектирования;
- конструкторско-технологического проектирования.

На этапе *системного проектирования* определяется архитектура будущей системы, состав компонентов и основные характеристики системы при таком её построении.

При *структурно-алгоритмическом проектировании* определяются алгоритмы функционирования аппаратных и программных компонентов системы.

На этапе *функционально-логического проектирования* разрабатываются функциональные и принципиальные электрические схемы, программы, подготавливаются тестовые и контрольные данные.

На *конструкторском* этапе производится привязка элементов проекта к конструктивным элементам. Широкое использование САПР на всех этапах проектирования приводит к тому, что современные подходы к разбиению процесса проектирования связывают с различием как технических средств (инструментария), привлекаемых для создания проекта, так и технических средств, используемых в качестве компонентов проекта и технологических особенностей реализации конечного продукта. Хотя общая методология процесса проектирования не зависит от

варианта разбиения процесса проектирования на отдельные уровни, содержание, а также методы и средства проектирования для различных уровней оказываются очень специфичными и существенно зависят как от типа применяемой элементной базы, так и от способа реализации (изготовления) конечного продукта.

1.2. Факторы, влияющие на специфику проектирования:

Среди факторов, влияющих на специфику проектирования, можно выделить:

- 1) тип обрабатываемой информации, методы и способы ее обработки (аналоговые, аналого-цифровые и/или цифро-аналоговые элементы)
- 2) техническая база и технический способ реализации проекта (стандартные и/или заказные ИС)

На рис.1 приведена классификация ИС

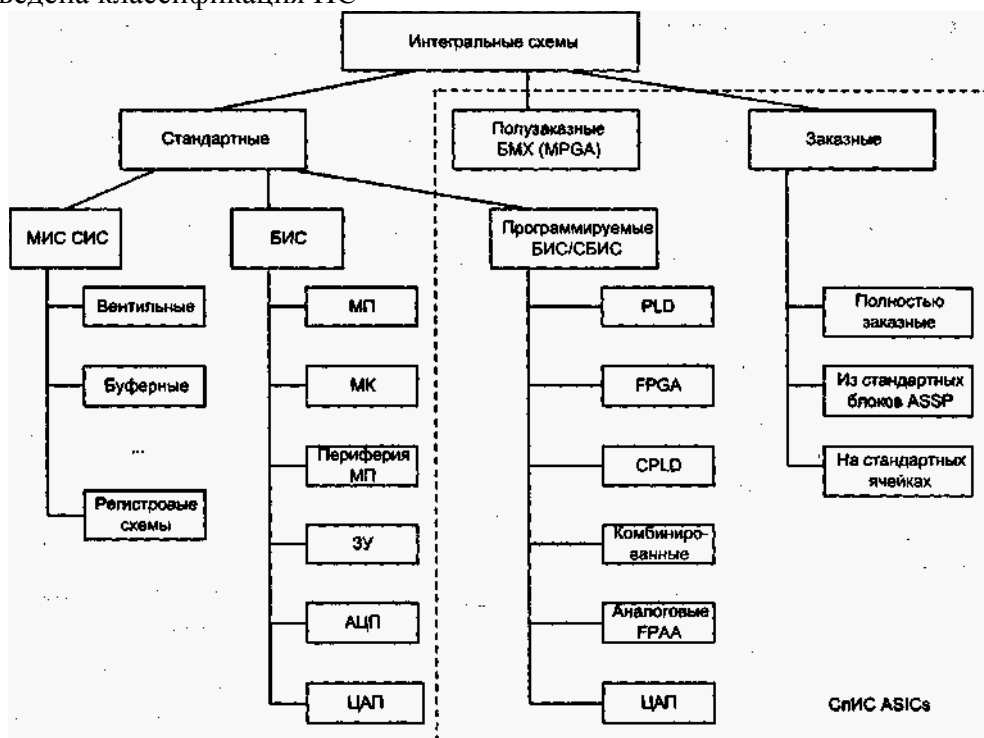


Рисунок 1.-Классификация ИС

К стандартным микросхемам отнесены схемы малой и средней степени интеграции -МИС и СИС. Эти микросхемы производятся массовыми тиражами и реализуют стандартные элементы и узлы, функционирование которых никак не определяется конкретными потребителями. К стандартным схемам высокого уровня интеграции (БИС и СБИС) относятся цифровые схемы: микропроцессоры, микроконтроллеры и запоминающие устройства (ЗУ), разнообразные периферийные схемы для МП и МК, включая и аналого-цифровые схемы: аналого-цифровые преобразователи (АЦП), цифроаналоговые преобразователи (ЦАП). Общее свойство этих схем то, что они остаются неизменными после изготовления независимо от устройств и систем, в которых они используются.

К специализированным ИС (СпИС) относятся все, структура которых в отличие от структур стандартных ИС массового производства каким-либо способом приспособляется к конкретным требованиям того или иного проекта. В английской терминологии СпИС именуется ASICs (Application Specific Integrated Circuits). Среди СпИС различают классы полузаказных и заказных. Разновидностями заказных микросхем являются полностью заказные и спроектированные методом «на стандартных ячейках».

Полностью заказные схемы целиком проектируются по требованиям конкретного заказчика. Проектировщик имеет полную свободу действий, определяя схему по своему

усмотрению вплоть до уровня схемных компонентов (отдельных транзисторов и т. п.). Для изготовления схемы требуется разработка всего комплекта фотшаблонов, верификация и отладка всех схемных фрагментов. Такие схемы очень дороги и имеют длительные циклы проектирования.

МП-система создается в результате разработки комплекса программно-аппаратных средств. Разработка аппаратной части сводится к компоновке системы из типовых модулей: центрального процессорного элемента, различных видов памяти, адаптеров, контроллеров и внешних устройств. Способы подключения модулей к шинам микропроцессорной системы, описания основных модулей, сведения о типах и методике их выбора, программировании и применении приводятся в литературе, посвященной описанию конкретных семейств. Ключевой проблемой при проектировании МП-систем является разработка программного обеспечения (ПО). Некоторые сведения о средствах и методике отладки ПО приведены в данной главе.

1.3. Этапы проектирования с помощью САПР

Основные этапы проектирования приведены на рисунке 2:

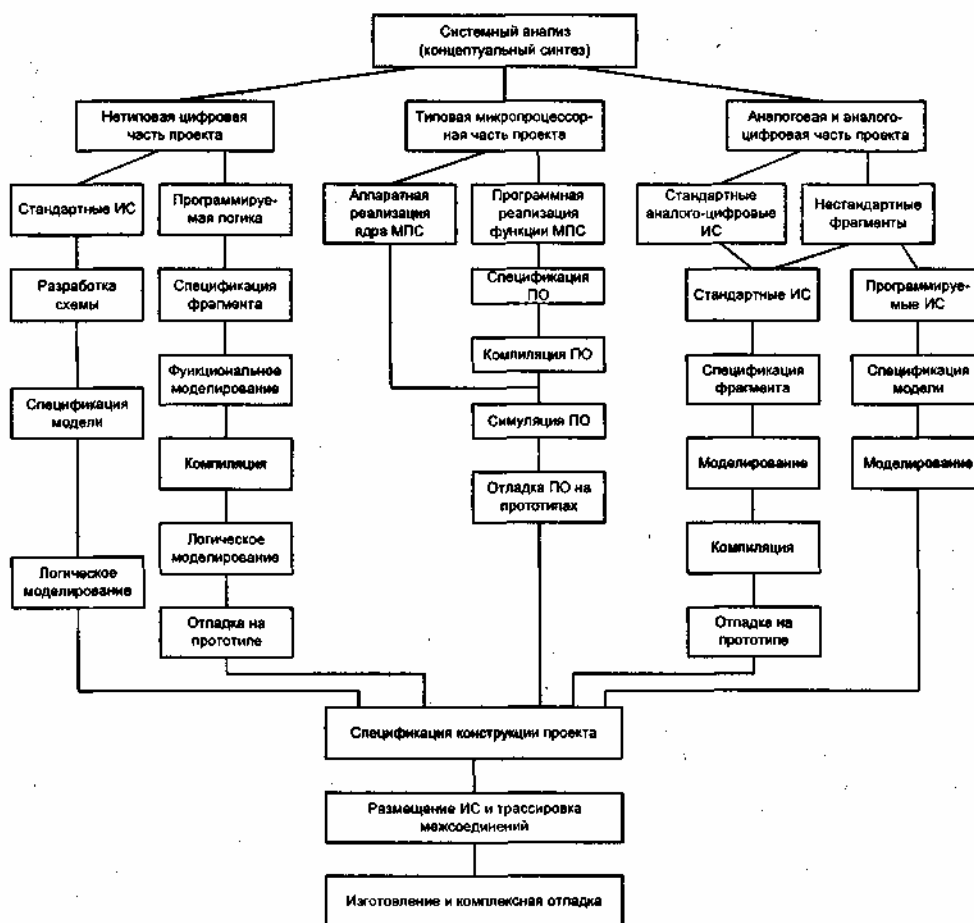


Рисунок 2.- Укрупненная структура алгоритма автоматизированного проектирования вычислительных устройств с использованием МП-средств

Проектирование на концептуальном уровне возлагается на проектировщика и слабо связано с автоматизацией. Исходные данные для проектирования на этом этапе содержат требования к основным технико-экономическим показателям: производительности, энергопотреблению, стоимости, надежности, конструктивным и другим параметрам. Кроме того, для управляющих систем должны быть определены реализуемые алгоритмы управления, для универсальных систем - классы выполняемых задач.

На этом уровне, исходя из требуемого функционирования устройства, проектировщик осуществляет разбиение проекта на части, определяет множества входных и выходных сигналов (как устройства в целом, так и его составных частей), их характер и взаимосвязь, а также решает отдельные вопросы реализации составных частей. Основным результатом этого этапа является разбиение алгоритмов работы системы на две составляющие для реализации программным и аппаратным обеспечением выбранного типа МП-ядра, а также выделение задач, требующих для своего выполнения разработки нетипового оборудования, как цифрового, так и аналогового.

Результаты концептуального этапа позволяют перейти к следующим этапам проектирования. Порядок работы по параллельным ветвям процедуры проектирования произволен и может во времени выполняться как параллельно или последовательно, так и в произвольных комбинациях. Более того, даже этап конструкторско-технологического проектирования может начинаться (благодаря перепрограммируемости результатов любой ветви проектирования) до получения окончательных результатов проектирования отдельных фрагментов устройства. Следует отметить схожесть процедур проектирования по всем параллельным ветвям. Как разработка программного обеспечения для МП- (МК) ядра, так и разработка дискретной и аналоговой частей проекта могут рассматриваться как последовательность трех этапов.

- 1) Ввод исходной для проектирования информации (спецификация работ этапа).
- 2) Компиляция проекта.
- 3) Верификация (тестирование) полученных результатов.

Конкретное содержание этапов для аппаратной и программной частей проекта (а тем более цифровой и аналоговой частей) естественно различное. Компиляция аппаратной части проекта приводит к синтезу устройства (или устройств) в базе выбранных элементов (со стандартной и/или программируемой структурой), а компиляция программной части проекта приводит к синтезу кодового представления программ. Полученные результаты требуют тщательной проверки, поэтому за этапом синтеза следует этап верификации, проводимого моделированием и/или реальными экспериментами. Моделирование, как правило, имеет несколько уровней с разной степенью отображения свойств реального объекта. Оно может быть функциональным, проверяющим правильность логической структуры устройства или программы, временным, учитывающим задержки сигналов в схемах устройства без учета окончательной топологии трассировки или время исполнения отдельных программных фрагментов и т. д. В результате верификации могут выявиться ошибки, требующие исправления, что придает процессу проектирования итеративный характер с возвратами к прежним этапам и введением в проект нужных коррекций.

Более того, по мере отработки решений по отдельным ветвям проектирования обрабатываются и вопросы связи между этими ветвями (например, между программной и аппаратной частями проекта), хотя комплексный анализ и отладка могут быть выполнены только после завершения отдельных ветвей процедуры проектирования. Естественно, такое последовательное проектирование является условным. В реальных условиях выполняется последовательно-параллельное проектирование с многократными итерационными возвратами к началу проектных процедур.

Завершение этапов проектирования по отдельным ветвям создает исходные данные для завершающего конструкторско-технологического этапа проектирования, результатом которого является создание реальной системы. Физическая реализация проекта в свою очередь создает основу для комплексной отладки решений, полученных на отдельных ветвях проектирования.

1.4. Сопряженное проектирование и сопряженная верификация

До настоящего времени в проектировании аппаратно-программных систем доминирует подход, основанный на разделении задачи на аппаратно-реализуемую и программно-реализуемую части на ранних этапах проектирования, и эти части проектируются относительно независимо вплоть до окончательного объединения системы. Тесное взаимодействие аппаратных и

программных средств как в системах типа «процессор общего назначения -программируемый аппаратный периферийный модуль», так и в системах SOPC потребовало разработки новых подходов к процессу проектирования, что нашло свое отражение в концепции «сопряженного проектирования аппаратно-программных систем» - (Hardware-Software Codesign).

Основа методологии сопряженного проектирования (сопроектирования) - параллельная взаимосвязанная проработка программных и аппаратных средств, что обеспечивает создание наиболее эффективных конфигураций при сокращении времени разработки.

Концепция сопроектирования предполагает решение следующих вопросов.

1) Анализ задачи и ее разделение на фрагменты, безусловно назначаемые к исполнению программно, безусловно исполняемые в аппаратуре, и фрагменты, которые могут быть назначены как в аппаратную, так и в программную части таким образом, чтобы максимизировать показатель качества системы в целом в зависимости от имеющихся ресурсов. Процедуру такого предварительного распределения весьма сложно формализовать. Рекомендуется назначать в программную часть сравнительно редко выполняемые фрагменты и фрагменты, требующие больших аппаратных ресурсов, например, содержащие операции арифметики с плавающей запятой. К безусловно аппаратным относят обычно операции непосредственного управления периферией.

2) Создание библиотеки возможных исполнителей алгоритмов, типичных для предполагаемой области применения. Каждый объект такой библиотеки представляет некоторую задачу и включает несколько вариантов программной реализации, например, в форме С-кодов, а также несколько вариантов реализующих структур, обычно представляемых как описания на языках схемотехнического проектирования, например VHDL. Эти варианты сопровождаются количественными характеристиками возможных исполнителей, таких как время исполнения, затраты памяти, используемые ресурсы микросхем программируемой логики.

3) Выбор оптимального сочетания исполнителей частей задачи исходя из определенной целевой функции, ограничений и характеристик задачи. Обычно за критерий оптимизации принимается время исполнения задачи. Имеющиеся ресурсы (память, свободные макроячейки FPGA и т. п.) выступают как ограничения. Задача поиска оптимума является дискретной оптимизационной задачей. Прямые, «точные» методы оптимизации, такие как метод ветвей и границ, требуют весьма большого времени решения. Известен ряд приближенных эвристических методов сокращения перебора, которые позволяют решать задачу выбора исполнителей с приемлемой точностью при сравнительно не-больших затратах.

4) Разработка соответствующего интерфейса между процессором общего назначения и специализированным модулем, равно как и между блоками, включаемыми в аппаратную часть системы. При этом следует обращать внимание на такие проблемы, как согласованность форматов данных, буферизация, взаимное оповещение и взаимное блокирование процессов.

1.5. Проектирование типовой конфигурации МП систем

1.5.1. Типовые конфигурации

Методика проектирования/отладки МП- и МК-систем имеют определенную специфику. В соответствии с названием МК-системы ориентированы на выполнение задач управления определенными устройствами или их комплексами. МП-системы можно условно разделить на два основных класса: универсальные, которые используются для решения широкого круга задач обработки информации, и управляющие, которые специализируются на решении задач управления процессами и объектами. Типичными примерами универсальных МП-систем являются персональные компьютеры и рабочие станции, которые применяются в самых различных сферах деятельности.

Управляющие МП-системы имеют много общего с МК. Они также содержат различные устройства, расширяющие возможности процессора для реализации сложных алгоритмов управления. При этом периферийные устройства, многие из которых располагаются на кристалле

МК, в МП-системах реализуются с помощью дополнительных микросхем, что повышает их стоимость и снижает надежность. Разработка интегрированных МП, имеющих в своем составе ряд периферийных устройств, и сложнофункциональных МК, содержащих высокопроизводительное 32-разрядное процессорное ядро, приводит к размыванию границы применения управляющих МП- и МК- систем, постепенному стиранию функциональных и структурных различий между ними.

Основной особенностью МК является наличие в их составе ПЗУ (ППЗУ, РППЗУ, ЭСПЗУ, флэш-памяти), в которое записывается резидентная рабочая программа системы. Разработка, отладка и запись в ПЗУ этой программы является важнейшей стадией проектирования МК-систем. Записанная в ПЗУ рабочая программа становится составной частью системы, последующее изменение или коррекция которой обычно нежелательны или невозможны. При использовании внутреннего ПЗУ возможности внешнего контроля работы МК в процессе отладки очень ограничены. Поэтому комплексная отладка программного и аппаратного обеспечения МК-систем является достаточно сложной процедурой, требующей использования специализированных методов и средств контроля. Данный этап проектирования является также наиболее ответственным, так как невыявленная ошибка может привести к весьма дорогостоящим последствиям. Особенностью МП-систем для ряда областей применения является необходимость строгого соблюдения определенных норм времени на выполнение программы или ее отдельных модулей.

В МП-системах выполняемые модули рабочей программы загружаются в ОЗУ. Благодаря этому имеется возможность оперативной коррекции рабочей программы в случае необходимости. В процессе отладки проектировщик имеет доступ к общей шине, что облегчает текущий контроль за работой системы. Однако наличие в большинстве современных МП внутренней кэш-памяти ограничивает возможности внешнего контроля за ходом выполнения программы. Особенно возрастают сложности отладки при использовании МП с суперскалярной структурой, в которых несколько команд выполняются одновременно и естественная очередность их выполнения может не соблюдаться. Хотя при проектировании МП-систем выполняются практически те же этапы, что и для МК-систем, однако используемая процедура разработки и средства отладки во многих случаях существенно различаются. Рассмотрим основные этапы проектирования/отладки этих систем и особенности их реализации.

1.5.2. Основные этапы процедуры проектирования

Общая процедура проектирования-отладки МП- и МК-систем включает этапы, показанные на рисунке 3. Исходные данные для проектирования содержат требования к основным технико-

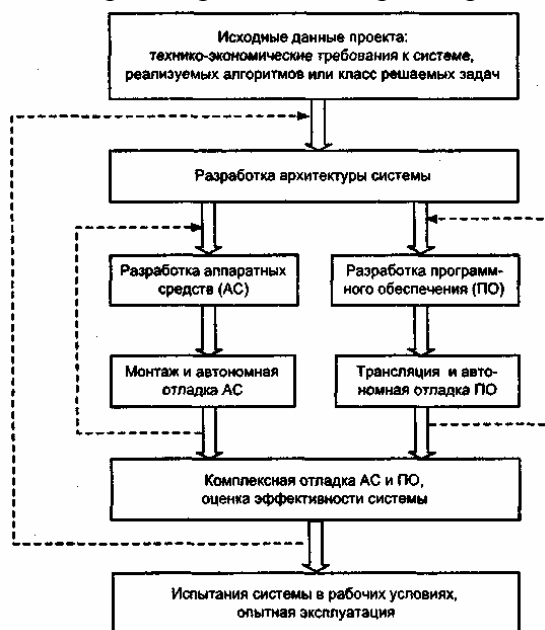


Рисунок 3.- Основные этапы проектирования/отладки МП- и МК-систем
экономическим показателям: производительности, энергопотреблению, стоимости, надежности, конструктивным и другим параметрам.

Выбор типа МП-средств определяется прежде всего, классом выполняемых задач. В табл.1 приведены некоторые классы задач и рекомендации типа МП-средств.

Таблица 1. Характеристики задач.

Характеристика задачи	Тип МП – средства	Основные требования
Управление специализированными и/или в реальном времени объектами и процессами, как встраиваемое или автономное оборудование.	Микроконтроллеры	Быстродействие; габариты, стоимость.
Управление системами и устройствами связи	Универсальные МП	Быстродействие; габариты, стоимость, объем внешней и внутренней памяти.
Информационно – измерительные системы: 2. Система регистрации аналоговых данных 3. Измерительные приборы		
Системы управления технологическими процессами(ТП): • Устройства сбора данных для контроля параметров ТП • и др.		Унификация, гибкость подсистемы ввода вывода информации, габариты, стоимость, потребление энергии.
Задачи фильтрации и поиск сигналов, вычисление спектров, преобразование сигналов из одного вида в другой, устранение отражений и выделение полезного сигнала на фоне помех	Сигнальные микропроцессоры и нейропроцессоры	малые габариты, легкая стыковка с аналого-цифровыми и цифро-аналоговыми преобразователями, большая разрядность обрабатываемых данных и небольшой набор математических операций
Обработка сигналов в реальном времени: • распознавание речи • обработка изображений	Транспьютеры	Максимальное быстродействие.
	Нейропроцессоры	

Если состав средств, имеющихся на плате развития, достаточен для реализации проектируемой системы, то ее разработка сводится к созданию ПО и выполнению комплексной отладки системы. Если имеющихся средств недостаточно, то они проектируются и размещаются на дополнительной плате, подключаемой к разъему на плате развития не посредственно или с помощью кабеля. Так реализуется прототип проектируемой системы, на котором можно выполнить комплексную отладку программных и аппаратных средств, а в ряде случаев и провести проверку их функционирования в рабочих условиях. После этого нетрудно разработать рабочий вариант системы, объединив на одной плате используемые модули прототипной системы. Прототипная система может использоваться в качестве рабочей (целевой), если ее параметры и конструктивное оформление удовлетворяют требованиям технического задания. В этом случае достигается сокращение сроков и стоимости проектирования системы.

Следует отметить перспективность использования при разработке АС мезонинной технологии, которая унифицирует размеры и интерфейс базовой платы-носителя и размещаемых над ней небольших плат - мезонинов (типичный размер 45x99 мм). Одна плата-носитель несет от 2 до 12 мезонинов. Каждый мезонин соединяется с носителем двумя разъемами.

На этапе автономной отладки АС основными орудиями разработчика являются традиционные измерительные приборы - осциллографы, мультиметры, пробники и другие, а также логические анализаторы, которые обладают широкими возможностями контроля состояния различных узлов системы в заданные моменты времени. Весьма эффективным является использование на этом этапе средств тестирования по стандарту JTAG, которые имеются в составе многих современных моделей МП и МК. С помощью размещенного на кристалле тест-порта TAP и специальных выводов TDI, TDO, TCK, TMS, TRST# обеспечивается возможность подачи необходимых входных воздействий и считывания выходной реакции, запуск-останов процессора, изменение режима его работы. Вводом специальной команды можно установить выходы микропроцессора или микроконтроллера в отключенное состояние, чтобы отдельно протестировать другие устройства системы,

1.6. Средства и методы проектирования и автономной отладки аппаратных средств МП – системы.

1.6.1. Выбор семейства МП/МК и стандартной периферии

Разработка архитектуры системы подразумевает определение оптимального состава АС и ПО для решения поставленных задач. При этом разработчик решает, какие функции системы будут реализованы АС, а какие ПО. Определяется номенклатура АС - выбираются тип МП или МК, объем и тип памяти, номенклатура периферийных устройств, протоколы обмена информацией и состав требуемых сигналов управления системой. Определяется также состав ПО - наличие операционной системы, ее тип и характеристики, номенклатура необходимых программных модулей, характер их взаимодействия, используемый язык программирования. Результатом выполнения этого этапа являются частные технические задания на проектирование АС и ПО.

Широкая номенклатура МП и МК, выпускаемых различными фирмами, позволяет удовлетворить запросы подавляющего большинства потребителей. Однако выбор типа МП/МК является только первым шагом на пути создания системы, соответствующей требованиям заказчика. Реализация такой системы является сложным и трудоемким процессом, выполнение которого на современном уровне невозможно без использования комплекса специализированных программных и аппаратных средств, помогающих разработчику на различных этапах проектирования, программирования и отладки. Поэтому при оценке и выборе типа МП/МК для конкретного применения необходимо учитывать не только его технико-экономические характеристики, но и уровень развития программно-аппаратных средств, предлагаемых для использования в процессе проектирования-отладки систем на его основе.

Этап разработки АС может быть выполнен традиционными методами, с помощью которых проектируется и моделируется электрическая схема, разрабатывается печатная плата или комплект плат, после чего выполняются монтаж и отладка системы. Однако во многих случаях можно обеспечить сокращение сроков и повышение качества разработки АС путем использования «полуфабрикатов» или готовых изделий, выпускаемых рядом производителей.

1.6.2. Аппаратные средства отладки

На этапе автономной отладки АС основными орудиями разработчика являются традиционные измерительные приборы - осциллографы, мультиметры, пробники и другие, а также логические анализаторы, которые обладают широкими возможностями контроля состояния различных узлов системы в заданные моменты времени. Весьма эффективным является использование на этом этапе средств тестирования по стандарту JTAG, которые имеются в составе многих современных моделей МП/МК. С помощью размещенного на кристалле тест-порта TAP и специальных выводов TDI, TDO, TCK, TMS, TRST# обеспечивается возможность подачи необходимых входных воздействий и считывания выходной реакции, запуск/останов процессора, изменение режима его работы. Вводом специальной команды можно установить выводы МП/МК в отключенное состояние, чтобы отдельно протестировать другие устройства системы.

1.7. Средства и методы разработки программного обеспечения

При программировании управляющих систем чаще всего используются машинно-ориентированный язык Ассемблера или языки C / C++. Язык Ассемблера применяется в случаях, когда имеются жесткие ограничения на объем требуемой памяти или на время выполнения программных модулей. Такие случаи являются достаточно типичными при решении задач управления, поэтому Ассемблеры являются одним из основных средств создания ПО для МК-систем. В тех случаях, когда указанные ограничения не очень жесткие, для создания ПО используются языки высокого уровня (обычно C / C++). Автономная отладка ПО выполняется с помощью симулятора - программной модели используемого микропроцессора или микроконтроллера. На этом этапе разработчики используют широкий набор средств программирования - компиляторы, ассемблеры, дисассемблеры, отладчики, редакторы связей и другие, без которых практически невозможно создание работоспособного ПО в течение ограниченных сроков выполнения проекта.

Комплексная отладка АС и ПО является наиболее сложным и ответственным этапом создания системы. На этом этапе разработчик использует весь набор программных и аппаратных средств, применяющихся для автономной отладки АС и ПО, а также ряд специальных средств комплексной отладки. К числу таких средств относятся схемные эмуляторы -специализированные устройства, включаемые вместо МП/МК прототипной системы и обеспечивающие возможность контроля ее работы с помощью ПЭВМ, связанного со схемным эмулятором. являются наиболее эффективным средством комплексной отладки систем.

Одним из наиболее эффективных средств комплексной отладки МК-систем являются эмуляторы ПЗУ. Это устройство включается вместо ПЗУ прототипной системы и работает под управлением подключенного к нему ПЭВМ. Так обеспечивается текущий контроль за выполнением программы и ее оперативная коррекция, что значительно упрощает процесс отладки.

Для МК-систем заключительной процедурой комплексной отладки является запись в ПЗУ объектных модулей отлаженной программы и завершающее испытание ее работоспособности. Запись программы в ПЗУ осуществляется с помощью специальных программаторов.

После выполнения указанных этапов отлаженная прототипная система может быть испытана в рабочих условиях с подключением полного набора реальных периферийных устройств и объектов управления. В процессе опытной эксплуатации выявляются ошибки, не обнаруженные на этапе отладки, определяется реакция системы на возможные непредвиденные ситуации.

1.8. Программные средства поддержки проектирования и отладки систем

В процессе разработки и отладки программного обеспечения микропроцессорных систем используются следующие программные средства: ассемблеры, компиляторы; симуляторы (программно-логические модели); отладчики, редакторы связей (компоновщики, загрузчики).

В современных комплексах проектирования/отладки систем эти средства обычно работают совместно, в составе интегрированной среды (оболочки) программирования.

Особенно сложные задачи приходится решать при программировании управляющих систем, работающих в реальном масштабе времени. В этом случае разработчик должен использовать какую-либо из имеющихся операционных систем реального времени (ОСРВ) или создавать собственные программы-мониторы реального времени с помощью указанных выше средств программирования. В составе многих ОСРВ имеются средства поддержки программирования, которые могут использоваться при проектировании/отладке системы.

Таким образом, при создании ПО МП- и МК-систем разработчик имеет следующие возможности: использовать набор отдельных средств поддержки программирования (ассемблер или компилятор, симулятор, отладчик), которые предлагаются рядом фирм-производителей; выполнять программирование и отладку с помощью интегрированной среды разработки (development environment); разрабатывать программное обеспечение с помощью средств поддержки, имеющихся в составе ОСРВ, которая используется в проектируемой системе.

В настоящее время программирование и отладка чаще всего выполняются с помощью интегрированной среды развития или средств ОСРВ. Программирование производится обычно с помощью кросс-средств, установленных на инструментальном компьютере с мощной операционной системой. В качестве инструментальных компьютеров используются ПЭВМ. Операционными системами этих компьютеров служат различные версии Windows и UNIX (Solaris, AIX, ULTRIX и другие).

Язык Ассемблера очень часто применяется при программировании МП- и МК-систем, так, его использование обеспечивает существенное уменьшение объема памяти программ и времени выполнения программных модулей (до 20 - 50 %). Упрощенные (демонстрационные) версии Ассемблеров для всех семейств МП и МК Motorola предоставляются бесплатно рядом фирм и распространяются по сети Интернет. Эти версии обычно имеют ограничения на объем транслируемых программ (до нескольких сотен или тысяч строк), а также не обеспечивают ряд сервисных возможностей. Ассемблеры с широким набором функциональных возможностей, включая макросы (макроассемблеры), поставляются рядом разработчиков, в том числе фирмой «Motorola».

В качестве языков высокого уровня чаще всего используются С, С++. Некоторыми фирмами поставляются также компиляторы для языков FORTRAN, Modula-2, Ada, Pascal. Все эти компиляторы обеспечивают также программирование на языке Ассемблера. Большинство из них содержат компоновщики для связи программных модулей, библиотеки функций. Многие компиляторы по указанию разработчика могут оптимизировать процесс трансляции исходного текста с целью получения объектного кода с минимальным объемом или минимальным временем выполнения программы. Такие компиляторы называются оптимизирующими.

1.9. Средства и методы комплексной отладки МП-систем

К ним относятся программаторы[1§8.6.1.], логические анализаторы[1§8.6.2.] и встроенные в МП средства отладки.

2. Методические указания к разработке МП – системы

2.1. Исследовательский раздел.

2.1.1. Исследование предметной области курсовой работы

- 2.1.1.1. Проанализировать задание на курсовую работу, выявить специфику алгоритма решения задачи и определить её принадлежности к типовому классу алгоритмов или задач, решаемых МП – системами.
- 2.1.1.2. Ознакомиться по рекомендуемым источникам и методическим указаниям с методологией (направлениям, принципам и методам) и методикой(основными этапами) разработки МП- систем.
- 2.1.1.3. Выполнить поиск дополнительных источников, соответствующих заданию на курсовую работу, проанализировать и выбрать из них основные.
- 2.1.1.4. Изучить по основным источникам теоретические основы и принципы функционирования аналогичных разрабатываемой МП – системе и критерии их эффективности.
- 2.1.1.5. Выбрать и оценить параметры и характеристики, аппаратные и программные средства, архитектуру и прототипы проектируемой МП – системы.
В ходе изложения результатов в исследованиях показать знание основных понятий и терминов.

2.1.2. Составление технического задания на разработку МП – системы.

- 2.1.2.1. Доопределить необходимые исходные данные для разработки МП – системы: параметры, характеристики, режимы.
- 2.1.2.2. Выделить комплекс проблем составляющих сущность функционирования и проектирования разрабатываемой МП- системы и выделить из них те, что будут решены в курсовой работе.
- 2.1.2.3. По критериям проанализировать возможности разных классов МП – средств, обосновать выбор класса. Обосновать критерии выбора возможности применения для реализации задачи типовых классов МП – средств (МК и МП). Привести результаты сравнения (выбора) в виде таблиц.
- 2.1.2.4. Сформулировать техническое задание на курсовую работу, уточняя, детализируя и конкретизируя задание, полученное у преподавателя, на основании выполненного выше анализа.

2.2. Конструкторский раздел.

2.2.1. Системное проектирование.

- 2.2.1.1. Обосновать требования к основным технико – экономических показателей эффективности (критериям эффективности) МП – системы: производительности, надёжности или др.
- 2.2.1.2. Выделить функции и процессы, составляющие решение задачи с помощью МП – системы. Описать способы (алгоритмы) их реализации в выбранном классе МП – средств.
- 2.2.1.3. На основе выявленного класса алгоритмов (задач) выбранного класса МП – средств, а также выделенных для реализации функций и процессов, обосновать выбор архитектуры, состава и конфигурации МП – системы, по следующим основным параметрам и характеристикам, с учётом выбранных в п.3.1.1. критериев:
 - а) система команд

- Cisc, Risc и др.
- векторная, скалярная, суперскалярная
- б) разрядности данных
- в) система прерываний
- г) внутренней шинной организации МП – средства
 - Принстонская
 - Гарвардская
- д) универсальные средства ввода – вывода
- е) возможности подсистемы реального времени
- ж) способы управления памятью в МП
- з) возможности ресурсов МП – средств
- и) интерфейс с ПЭВМ, для мульти МП – систем

2.2.1.4. Сделать предварительный выбор семейства МП

Результатом системного проектирования должно быть обоснование и выбор критериев эффективности, функции и процессов, а также требования к архитектуре и структуре МП ядра системы.

2.2.2. Структурно – алгоритмическое проектирование

2.2.2.1. На основании выделенных функций МП–системы и типовой структуры с использованием концепции «черного ящика» и метода *функциональной декомпозиции* представить МП – систему во взаимосвязи с внешней средой (управляемыми и управляющими объектами), например, как показано на рисунке 1 а),б),в):

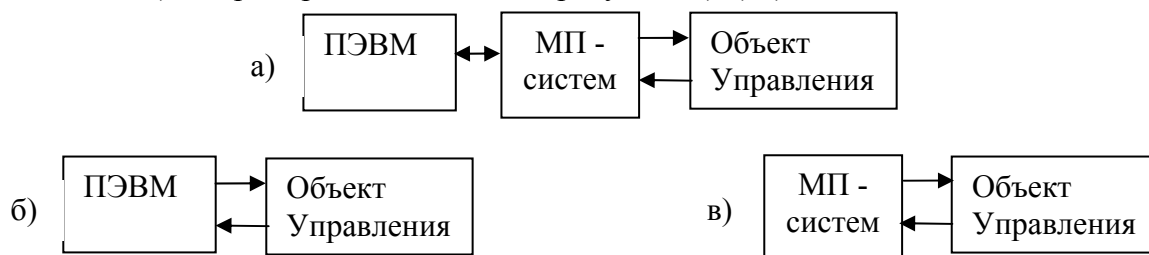


Рисунок 1.-Взаимосвязь МП-системы с внешней средой

Уточнить количество, разрядность типовых шин, линии, сигналов в том минимальном объёме, который следует из описания типовой задачи в техническом задании на МП – систему.

2.2.2.2. Составить в виде блочной схемы алгоритмы решения задачи МП – системы, в терминах задачи (действий, выявленных на этапе исследования теоретических основ, принципов, законов, режимов и алгоритмов решения задачи).

2.2.2.3. Проанализировать полученный в п.2.2.1.2.. алгоритм: выявить наличие специфичных блоков вычислений и др., оценить частотность и продолжительность исполнения отдельных специфичных операций, и обосновать распределение алгоритмов или их фрагментов между программным и аппаратным обеспечением выбранного МП – ядра (МП - средств). Причем ряд фрагментов можно безусловно назначить для аппаратной и безусловно для программной реализации, а ряд фрагментов могут быть назначены как в программной, так и в аппаратной части. В программную часть назначаются сравнительно редко выполняемые фрагменты алгоритмов и фрагменты, требующие больших аппаратных ресурсов, не содержащихся в выбранном типовом классе МП – средств. К исполнению в аппаратной части относят операции управления переферией.

Обосновать оптимальное распределение функций между аппаратным и программным обеспечением исходя из критерия ограничений. Если за критерий принять время исполнения задачи, то ресурсы (память, модули ввода-вывода, специализированные модули и т.д.) выступают как ограничения. Показать эвристические методы оценки выбора возможных вариантов распределения.

- 2.2.2.4. Сделать окончательный выбор модели МП/МК из выбранного ранее семейства(см.2.2.1.4.). Результаты выбора сравнения моделей свести в таблицу.
- 2.2.2.5. Составить схему алгоритм функционирования аппаратной части, используя как термины задачи, так и названия основных типовых модулей выбранного базового МП – средства.
- 2.2.2.6. Составить структуру программного обеспечения и блочную схему алгоритма выполнения основного модуля программы в терминах задачи и типовых команд и процедур, выбранного класса МП – средства.

2.2.3. Функционально- логическое проектирование

2.2.3.1. Разработка аппаратной части МП – системы.

- 2.2.3.1.1. Составить блочную схему функционирования аппаратной части МП – системы, реализующей алгоритм, полученный в п.2.2.2.4. включив в неё, кроме основного МП средства, дополнительные функциональные блоки, буферов, генераторов, преобразователей и т.д., протоколов обмена информацией внутри МП – системы и между МП – системой и внешней средой. Описать сигналы, функции и законы работы каждого блока в формализованном виде: таблицами, временными диаграммами, схемами алгоритмов.
- 2.2.3.1.2. Выбрать тип (семейство, модель) МП – средства, определённого на этапе системного проектирования. Привести данные выбора из ряда моделей в табличной форме и дать обоснование выбора, не противоречащее решениям, принятым ранее.
- 2.2.3.1.3. Привести функциональную схему кристалла БИС МП – средства, описание использующих сигналов, анализ используемых терминов, алгоритмов работы и логическую организацию используемых ресурсов МП/МК.
- 2.2.3.1.4. Составить функциональную схему аппаратной части МП – системы. Составить полную спецификацию внутренних и внешних сигналов МП – системы в виде таблиц со следующим описанием сигнала: имя, тип и назначение сигнала.
- 2.2.3.1.5. Представить описание работы аппаратной части в виде упрощенной временной диаграммы её функционирования, включить в неё все блоки МП – системы и отдельные модули МП – средств. Диаграмма должна иллюстрировать основной типовой режим работы МП – системы.

2.2.3.2. Разработка программной части МП – системы

- 2.2.3.2.1. Уточнить состав и взаимодействие программного обеспечения (ПО) МП – системы, определенный в п.2.2.2.5.
- 2.2.3.2.2. Выполнить анализ системы команд в формализованном виде: таблиц, графов, деревьев и т.п. по следующим компонентам:
- типы и форматы используемых в программе данных
 - форматы используемых команд
 - группы операций, в т.ч. используемых
 - способы адресации
- 2.2.3.2.3. Разработать алгоритмы программ, в терминах операций, типовых команд МП – средств.
- 2.2.3.2.4. Составить программу на языке ассемблер специфического фрагмента алгоритма основного программного модуля.
- 2.2.3.2.5. Для выделенного в п.2.2.3.2.4. фрагмента просчитать числовой контрольный пример, иллюстрирующий исполнение программы.
- 2.2.3.2.6. При необходимости, на любом этапе разработки МП – системы можно вернуться к предыдущим этапам, и в них пересмотреть свои решения.

2.2.4. Анализ функционирования МП – системы с помощью временной диаграммы.

Диаграмма должна отображать все применения на входных, выходных и внутренних шинах и шинах МП – системы в пределах выбранного временного интервала. Временной интервал должен охватывать основной фрагмент программного модуля, а также предшествующие и последующие преобразования и передачи информации на внутренних и внешних шинах и шинах МП – систем. Ось абсцисс (времени) должна быть получена с точностью до периода сигнала тактовой частоты. Выполнение 1-2х команд показать с точностью до периода сигнала такта. Остальные - свернуть с точностью до применения сигналов на входах – выходах МП – средств. Не забывать в эти же интервалы времени отображать сигналы на других блоках МП – системы. Ось времени можно разрывать, не показывая однотипные или простые команды, не ведущие к изменению сигналов на выводах МК/МП, и переходить к более важным командам передачи и преобразования.

На временной диаграмме с помощью кривых линий со стрелками (можно выполнить рукописным способом на готовой диаграмме). Отобразить все важные причинно-следственные динамические параметры сигналов.

2.2.5. Привести способы инженерных оценок показателей, методику верификации МП – системы.

3. Правила оформления

3.1 Общие требования

3.1.1 Страницы текста курсовой работы (к.р.) и включенные в к.р. иллюстрации и таблицы должны соответствовать формату А4 по ГОСТ 9327.

3.1.2 Курсовая работа должна быть выполнена с использованием ПЭВМ и принтера на одной стороне листа белой бумаги формата А4 через полтора интервала. Цвет шрифта должен быть черным, высота букв, цифр и других знаков — не менее 1,8 мм (кегель - не менее 12).

Текст к.р. следует печатать, соблюдая следующие размеры полей: правое — 10 мм, верхнее — 20 мм, левое и нижнее — 20 мм.

Разрешается использовать компьютерные возможности акцентирования внимания на определенных терминах, формулах, теоремах, применяя шрифты разной гарнитуры.

3.1.3 Вне зависимости от способа выполнения к.р. качество напечатанного текста и оформления иллюстраций, таблиц, распечаток с ПЭВМ должно удовлетворять требованию их четкого воспроизведения.

3.1.4 При выполнении к.р. необходимо соблюдать равномерную плотность, контрастность и четкость изображения по всей к.р.. В к.р. должны быть четкие, не расплывшиеся линии, буквы, цифры и знаки.

3.1.5 Опечатки, описки и графические неточности, обнаруженные в процессе подготовки к.р., допускается исправлять подчисткой или закрашиванием белой краской и нанесением на том же месте исправленного текста (графики) машинописным способом или черными чернилами, пастой или тушью — рукописным способом.

Повреждения листов текстовых документов, помарки и следы не полностью удаленного прежнего текста (графики) не допускаются.

3.1.6 Сокращение русских слов и словосочетаний в к.р.— по ГОСТ 7.12.

3.2 Построение курсовой работы

3.2.1 Наименования структурных элементов к.р., «Реферат», «Содержание», «Обозначения и сокращения», «Введение», «Заключение», «Список использованных источников» служат заголовками структурных элементов к.р..

3.2.2 Основную часть к.р. следует делить на разделы, подразделы и пункты. Пункты, при необходимости, могут делиться на подпункты. При делении текста к.р. на пункты и подпункты необходимо, чтобы каждый пункт содержал законченную информацию.

3.2.3 Разделы, подразделы, пункты и подпункты следует нумеровать арабскими цифрами и записывать с абзацного отступа.

Разделы должны иметь порядковую нумерацию в пределах всего текста, за исключением приложений.

Пример — 1, 2, 3 и т. д.

Номер подраздела или пункта включает номер раздела и порядковый номер подраздела или пункта, разделенные точкой.

Пример — 1.1, 1.2, 1.3 и т. д.

Номер подпункта включает номер раздела, подраздела, пункта и порядковый номер подпункта, разделенные точкой.

Пример - 1.1.1.1, 1.1.1.2, 1.1.1.3 и т. д.

После номера раздела, подраздела, пункта и подпункта в тексте точку не ставят.

3.2.4 Заголовки разделов, подразделов и пунктов следует печатать с абзацного отступа с прописной буквы без точки в конце, не подчеркивая.

Если заголовок состоит из двух предложений, их разделяют точкой.

3.3 Нумерация страниц курсовой работы

3.3.1 Страницы к.р. следует нумеровать арабскими цифрами, соблюдая сквозную нумерацию по всему тексту к.р.. Номер страницы проставляют в центре нижней части листа без точки.

3.3.2 Титульный лист включают в общую нумерацию страниц к.р.. Номер страницы на титульном листе не проставляют.

3.3.3 Иллюстрации и таблицы, расположенные на отдельных листах, включают в общую нумерацию страниц к.р..

Иллюстрации и таблицы на листе формата А3 учитывают как одну страницу.

3.4 Нумерация разделов, подразделов, пунктов, подпунктов курсовой работы

3.4.1 Разделы к.р. должны иметь порядковые номера в пределах всего документа, обозначенные арабскими цифрами без точки и записанные с абзачного отступа. Подразделы должны иметь нумерацию в пределах каждого раздела. Номер подраздела состоит из номеров раздела и подраздела, разделенных точкой. В конце номера подраздела точка не ставится. Разделы, как и подразделы, могут состоять из одного или нескольких пунктов.

3.4.2 Если документ не имеет подразделов, то нумерация пунктов в нем должна быть в пределах каждого раздела, и номер пункта должен состоять из номеров раздела и пункта, разделенных точкой. В конце номера пункта точка не ставится.

Пример

1 XXXXXXXXXXXXXXXXXXXX

1.1

1.2

1.3

Нумерация пунктов первого раздела документа

2 XXXXXXXXXXXXXXXXXXXX

2.1

2.2

2.3

Нумерация пунктов второго раздела документа

Если документ имеет подразделы, то нумерация пунктов должна быть в пределах подраздела и номер пункта должен состоять из номеров раздела, подраздела и пункта, разделенных точками, например:

3 Методы испытаний

3.1 XXXXXXXXXXXXX

3.1.1

3.1.2

3.1.3

Нумерация пунктов первого подраздела третьего раздела документа

3.2 XXXXXXXXXXXXX

3.2.1

3.2.2

3.2.3

Нумерация пунктов второго подраздела третьего раздела документа

3.4.3 Если раздел состоит из одного подраздела, то подраздел не нумеруется. Если подраздел состоит из одного пункта, то пункт не нумеруется. Наличие одного подраздела в разделе эквивалентно их фактическому отсутствию.

3.4.4 Если текст к.р. подразделяется только на пункты, то они нумеруются порядковыми номерами в пределах всего к.р..

3.4.5 Пункты, при необходимости, могут быть разбиты на подпункты, которые должны иметь порядковую нумерацию в пределах каждого пункта, например 4.2.1.1, 4.2.1.2, 4.2.1.3 и т. д.

3.4.6 Внутри пунктов или подпунктов могут быть приведены перечисления.

Перед каждым перечислением следует ставить дефис или, при необходимости ссылки в тексте документа на одно из перечислений, строчную букву (за исключением ё, з, о, г, ь, й, ы, ъ), после которой ставится скобка.

Для дальнейшей детализации перечислений необходимо использовать арабские цифры, после которых ставится скобка, а запись производится с абзачного отступа, как показано в примере.

Пример

- а) _____
- б) _____
- 1) _____
- 2) _____
- в) _____

3.4.7 Если к.р. состоит из двух и более частей, каждая часть должна иметь свой порядковый номер. Номер каждой части следует проставлять арабскими цифрами на титульном листе под указанием вида к.р., например, «Часть 2».

3.4.8 Каждый структурный элемент к.р. следует начинать с нового листа (страницы).

3.4.9 Нумерация страниц к.р. и приложений, входящих в состав к.р., должна быть сквозная.

3.5 Иллюстрации

3.5.1 Иллюстрации (чертежи, графики, схемы, компьютерные распечатки, диаграммы, фотоснимки) следует располагать в к.р. непосредственно после текста, в котором они упоминаются впервые, или на следующей странице.

Иллюстрации могут быть в компьютерном исполнении, в том числе и цветные.

На все иллюстрации должны быть даны ссылки в тексте КР.

3.5.2 Чертежи, графики, диаграммы, схемы, иллюстрации, помещаемые в к.р., должны соответствовать требованиям государственных стандартов Единой системы конструкторской документации (ЕСКД).

Допускается выполнение чертежей, графиков, диаграмм, схем посредством использования компьютерной печати.

3.5.3 Фотоснимки размером меньше формата А4 должны быть наклеены на стандартные листы белой бумаги.

3.5.4 Иллюстрации, за исключением иллюстрации приложений, следует нумеровать арабскими цифрами сквозной нумерацией.

Если рисунок один, то он обозначается «Рисунок 1». Слово «рисунок» и его наименование располагают посередине строки.

3.5.5 Допускается нумеровать иллюстрации в пределах раздела. В этом случае номер иллюстрации состоит из номера раздела и порядкового номера иллюстрации, разделенных точкой. Например, Рисунок 1.1.

3.5.6 Иллюстрации, при необходимости, могут иметь наименование и пояснительные данные (подрисуночный текст). Слово «Рисунок» и наименование помещают после пояснительных данных и располагают следующим образом: **Рисунок 1 — Детали прибора.**

3.5.7 Иллюстрации каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения. Например, Рисунок А.3.

3.5.8 При ссылках на иллюстрации следует писать «... в соответствии с рисунком 2» при сквозной нумерации и «... в соответствии с рисунком 1.2» при нумерации в пределах раздела.

3.6 Таблицы

3.6.1 Таблицы применяют для лучшей наглядности и удобства сравнения показателей. Название таблицы, при его наличии, должно отражать ее содержание, быть точным, кратким. Название таблицы следует помещать над таблицей слева, без абзачного отступа в одну строку с ее номером через тире.

При переносе части таблицы название помещают только над первой частью таблицы, нижнюю горизонтальную черту, ограничивающую таблицу, не проводят.

3.6.2 Таблицу следует располагать в к.р. непосредственно после текста, в котором она упоминается впервые, или на следующей странице.

3.6.3 На все таблицы должны быть ссылки в к.р.. При ссылке следует писать слово «таблица» с указанием ее номера.

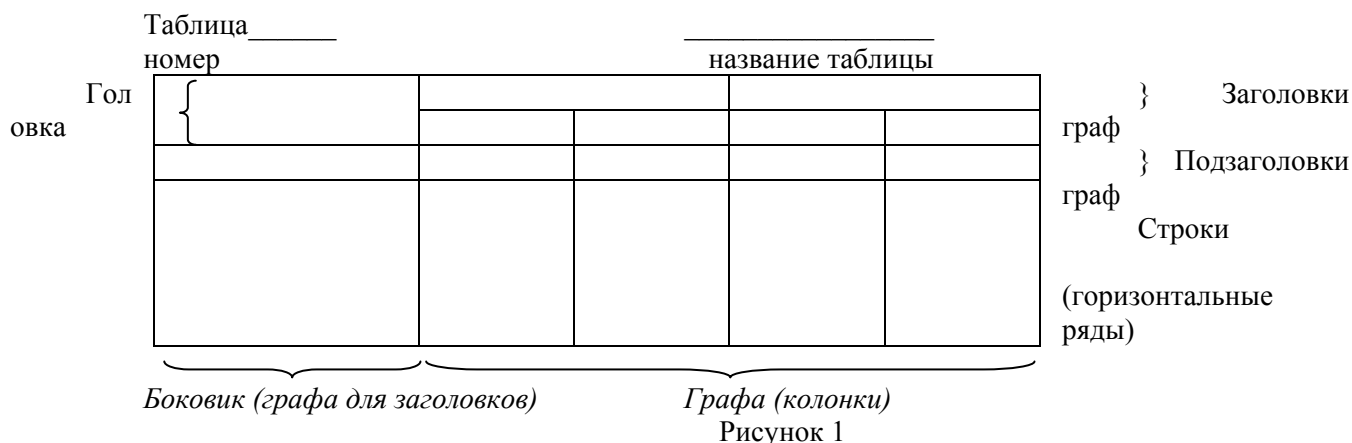
3.6.4 Таблицу с большим количеством строк допускается переносить на другой лист (страницу). При переносе части таблицы на другой лист (страницу) слово «Таблица» и номер ее указывают один раз справа над первой частью таблицы, над другими частями пишут слово «Продолжение» и указывают номер

таблицы, например: «Продолжение таблицы 1». При переносе таблицы на другой лист (страницу) заголовок помещают только над ее первой частью.

Таблицу с большим количеством граф допускается делить на части и помещать одну часть под другой в пределах одной страницы. Если строки и графы таблицы выходят за формат страницы, то в первом случае в каждой части таблицы повторяется головка, во втором случае — боковик.

Если повторяющийся в разных строках графы таблицы текст состоит из одного слова, то его после первого написания допускается заменять кавычками; если из двух и более слов, то при первом повторении его заменяют словами «То же», а далее — кавычками. Ставить кавычки вместо повторяющихся цифр, марок, знаков, математических и химических символов не допускается. Если цифровые или иные данные в какой-либо строке таблицы не приводят, то в ней ставят прочерк.

3.6.5 Цифровой материал, как правило, оформляют в виде таблиц. Пример оформления таблицы приведен на рисунке 1.



3.6.6 Таблицы, за исключением таблиц приложений, следует нумеровать арабскими цифрами сквозной нумерацией.

Допускается нумеровать таблицы в пределах раздела. В этом случае номер таблицы состоит из номера раздела и порядкового номера таблицы, разделенных точкой.

Таблицы каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения.

Если в документе одна таблица, то она должна быть обозначена «Таблица 1» или «Таблица В.1», если она приведена в приложении В.

3.6.7 Заголовки граф и строк таблицы следует писать с прописной буквы в единственном числе, а подзаголовки граф — со строчной буквы, если они составляют одно предложение с заголовком, или с прописной буквы, если они имеют самостоятельное значение. В конце заголовков и подзаголовков таблиц точки не ставят.

3.6.8 Таблицы слева, справа и снизу, как правило, ограничивают линиями. Допускается применять размер шрифта в таблице меньший, чем в тексте.

Разделять заголовки и подзаголовки боковика и граф диагональными линиями не допускается.

Горизонтальные и вертикальные линии, разграничивающие строки таблицы, допускается не проводить, если их отсутствие не затрудняет пользование таблицей.

Заголовки граф, как правило, записывают параллельно строкам таблицы. При необходимости допускается перпендикулярное расположение заголовков граф.

Головка таблицы должна быть отделена линией от остальной части таблицы.

3.6.9 Оформление таблиц в к.р. должно соответствовать ГОСТ 1.5 и ГОСТ 2.105.

3.7 Формулы и уравнения

3.7.2 Пояснение значений символов и числовых коэффициентов следует приводить непосредственно под формулой в той же последовательности, в которой они даны в формуле.

3.7.3 Формулы в к.р. следует нумеровать порядковой нумерацией в пределах всего к.р. арабскими цифрами в круглых скобках в крайнем правом положении на строке.

Пример

$A=a:b,$ (1)

$B=c:e.$ (2)

Одну формулу обозначают — (1).

3.7.4 Формулы, помещаемые в приложениях, должны нумероваться отдельной нумерацией арабскими цифрами в пределах каждого приложения с добавлением перед каждой цифрой обозначения приложения, например формула (B.1).

3.7.5 Ссылки в тексте на порядковые номера формул дают в скобках. Пример –... в формуле (1).

3.7.6 Допускается нумерация формул в пределах раздела. В этом случае номер формулы состоит из номера раздела и порядкового номера формулы, разделенных точкой, например (3.1).

3.7.7 Порядок изложения в к.р. математических уравнений такой же, как и формул.

3.7.8 В к.р. допускается выполнение формул и уравнений рукописным способом черными чернилами.

3.8 Обязательны ссылки на использованные источники, в том числе на приведенные таблицы и иллюстрации, которые следует приводить в квадратных скобках.

3.9 Перечень обозначений и сокращений, условных обозначений, символов, единиц физических величин и терминов.

Перечень должен располагаться столбцом. Слева в алфавитном порядке приводят сокращения, условные обозначения, символы, единицы физических величин и термины, справа — их детальную расшифровку.

3.10 Список использованных источников

Сведения об источниках следует располагать в порядке появления ссылок на источники в тексте к.р. и нумеровать арабскими цифрами без точки и печатать с абзацного отступа.

3.11 Приложения

3.11.1 Приложение оформляют как продолжение данного документа на последующих его листах или выпускают в виде самостоятельного документа.

3.11.2 В тексте документа на все приложения должны быть даны ссылки. Приложения располагают в порядке ссылок на них в тексте документа, за исключением справочного приложения «Библиография», которое располагают последним.

3.11.3 Каждое приложение следует начинать с новой страницы с указанием наверху посередине страницы слова «Приложение», его обозначения и степени.

Приложение должно иметь заголовок, который записывают симметрично относительно текста с прописной буквы отдельной строкой.

3.11.4 Приложения обозначают заглавными буквами русского алфавита, начиная с А, за исключением букв Ё, З, Й, О, Ч, Ъ, Ы, Ъ. После слова «Приложение» следует буква, обозначающая его последовательность.

Допускается обозначение приложений буквами латинского алфавита, за исключением букв I и O.

В случае полного использования букв русского и латинского алфавитов допускается обозначать приложения арабскими цифрами.

Если в документе одно приложение, оно обозначается «Приложение А».

3.11.5 Текст каждого приложения, при необходимости, может быть разделен на разделы, подразделы, пункты, подпункты, которые нумеруют в пределах каждого приложения. Перед номером ставится обозначение этого приложения.

3.11.6 Приложения должны иметь общую с остальной частью документа сквозную нумерацию страниц.

4. Примерные варианты заданий

Вариант 1. Разработать цифровой фильтр с КИХ ($y_n = \sum_{j=0}^n w_j \cdot X_{n-j}$) для отсчетов входного и выходного сигналов на базе процессора ЦОС.

Вариант 2. Разработать цифровой фильтр с КИХ ($y_n = \sum_{j=0}^n w_j \cdot X_{n-j}$) для отсчетов входного и выходного сигналов на базе микроконтроллера.

Вариант 3. Разработать устройство для спектрального анализа периодического сигнала на базе процессора ЦОС методом прямого дискретного преобразования Фурье с 8 сигнала.

Вариант 4. Разработать устройство для выполнения спектрального анализа периодического сигнала на базе процессора ЦОС методом обратного преобразования Фурье с 8 отсчетами сигнала.

Вариант 5. Разработать аппаратную часть и структуру ПО МПС на базе четырех-трех МП класса Pentium для обеспечения «горячего резервирования».

Вариант 6. Разработать систему самотестирования по стандарту JTAG (IEEE 1149.1) МП класса P6 (контроллер TAP и его подключение к порту TAP МП).

Вариант 7. Разработать систему самотестирования по стандарту JTAG (IEEE 1149.1) МК фирмы Motorola (TAP-контроллер и схему его подключения к МК).

Вариант 8. Разработать эмулятор КЭШ-памяти и ее контроллера для уровня L2 МП класса Pentium (P6) на базе МК.

Вариант 9. Разработать контроллер КЭШ-памяти для уровней L1 и L2 процессоров класса Pentium (P6).

Вариант 10. Разработать аппаратную часть и структуру ПО МПС на базе трех-четырёх МП класса Pentium для повышения достоверности результатов обработки информацию.

Вариант 11. Разработать блок управления и защитой сегментированной памяти в защищенном режиме МП класса Pentium (P6) на базе МК.

Вариант 12. Разработать блок управления и защиты блока страничной памяти в защищенном режиме МП класса Pentium (P6) на базе МК.

Вариант 13. Разработать контроллер управления мультизадачностью для процессоров класса Pentium.

Вариант 14. Разработать программную модель блока контроля и защиты процессора класса Pentium при использовании передач управления через шлюз вызова.

Вариант 15. Разработать контроллер прерываний и особых случаев для процессора класса Pentium.

Вариант 16. Разработать мост PCI/ISA для передачи информации на ISA-устройство на базе МП-системы.

Вариант 17. Разработать мост PCI/ISA для передачи информации с ISA-устройства на базе МП-системы.

Вариант 18. Разработать средства интерфейса для подключения нейроматрицы NM 6403 к последовательному порту ПЭВМ IBM PC.

Вариант 19. Разработать средства интерфейса для подключения нейроматрицы NM 6403 к параллельному порту ПЭВМ IBM PC.

Вариант 20. Разработать контроллер RAID-массива.

Вариант 21. Разработать МП-систему «горячего резервирования» на базе трех транспьютеров фирмы INMOS.

Вариант 22. Разработать МП-систему, обеспечивающую повышение достоверности результатов обработки информации, на базе трех- четырех транспьютеров фирмы INMOS.

Вариант 23. Разработать контроллер шины SCSI для реализации протокола SPI на базе МП-системы.

Вариант 24. Разработать контроллер (мост) PCI /SCSI для передач с шины SCSI.

Вариант 25. Разработать контроллер (мост) PCI /SCSI для передач на шину SCSI.

Список рекомендуемых источников

1. Микропроцессорные системы: Уч. пособие для вузов/ Под общ.ред Д.В.Пузанкова – СПб.: Политехника, 2002.-935 с.
2. Гук М. Процессоры Intel: от 86 до Pentium II: архитектура. Интерфейс. Программирование. – СПб.: Питер, 1997. – 221 с.
3. Гук М. Процессоры Pentium II, Pentium Pro и просто Pentium: архитектура. Интерфейс. Программирование. – СПб.: Питер, 1999. – 283 с.
4. Корнеев В.В. Современные микропроцессоры. –М.: “Нолидж”, 2000.-315с.
5. Калабеков Б.А. Микропроцессоры и их применение в системах передачи и обработки сигналов. М., «Радио и связь», 2000.
6. Микропроцессоры и микропроцессорные комплекты интегральных микросхем ред. под Шахнова В.А., М., “Радио и связь”, 1988 .
7. Бродин В.Б., Шагурин И.И. Микропроцессор i80486, 1993
8. Райхлин В.А. и др. Основы организации микропроцессорных систем, 1998.
9. Современные микроконтроллеры: архитектура, средства программирования, примеры применения, 1998
10. Шагурин И.И. Микропроцессоры и микроконтроллеры фирмы MOTOROLA, 1998
11. Гук М., Юров В. Процессоры Pentium III, Athlon и другие. – СПб.: Питер, 2000. – 478 с.
12. Райхлин В.А., Борисов А.Н. Основы организации микропроцессорных систем. Учеб. пособие для вузов. – Казань: Изд-во КГТУ, 1998. – 299 с.
13. Рудаков П.И., Финогенов К.И. Программируем на языке Ассемблера IBM PC. – Обнинск: Принтер, 1997. – 583 с.
14. Транспьютеры. Архитектура и программное обеспечение. – М.: Радио и связь, 1993. – 304с
15. Бахтеяров С.Д. и др. Транспьютерная технология. – М.: Радио и связь, 1993.-304с
16. Внутри транспьютера. – М.: “Мейкер”, 1993.-206с.
17. Транспьютеры: архитектура и программирование. Харп 1993
18. Бродин В.Б., Шагурин И.И. Микроконтроллеры: Архитектура, программирование, интерфейс. –М.: ЭКОН, 1999. – 398 с.
19. Современные микроконтроллеры: Архитектура, средства программирования, примеры применения, ресурсы сети Интернет/ Под ред. Коршуна И.В. – М.: АКИМ, 1998. – 250 с.
20. Калабеков Б.А. Цифровые устройства и микропроцессорные системы. –М.: Горячая линия – Телеком, 2000. – 336 с
21. Корнеев В.В. Параллельные вычислительные системы. –М.: Нолидж, 1998. – 311 с.
22. Мюллер Модернизация и ремонт ПК. М.: Вильямс, 2000-1132с
23. Балашов Е.П., Григорьев В.Л., Петров Г.А., Микро- и мини- ЭВМ, Л.: Энергоатомиздат. 1984. – 376с.

24. Балашов Е.П. Микропроцессоры и микропроцессорные системы, М.: Радио и связь, 1981. – 328с.
25. Преснухин Л.Н., Шахнов В.А. Конструирование ЭВМ и систем. – М: Высшая школа., 1986. 512с.
26. Пухальский Г.И., Проектирование Микропроцессорных систем. СПб.: Политехника, 2001. – 544с.
27. www.module.ru
28. www.poit.org.ru