

Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет
информатики и радиоэлектроники»

Кафедра электронно-вычислительных средств

***ПРОЕКТИРОВАНИЕ ПРОБЛЕМНО-ОРИЕНТИРОВАННЫХ
ВЫЧИСЛИТЕЛЬНЫХ СРЕДСТВ***

Учебное пособие

по курсовому проектированию
для студентов специальности I-40 02 02
«Электронные вычислительные средства»
дневной формы обучения

Минск 2005

УДК 004.9 (075.8)
ББК 32.973.26-02 я 73
П 91

Рецензент:

заведующий кафедрой радиоэлектронных средств БГУИР, профессор,
кандидат технических наук Н.С. Образцов

Авторы:

А.А. Петровский, В.Б. Ключ, А.Б. Давыдов, М.В. Качинский

Проектирование проблемно-ориентированных вычислительных
П 91 средств: Учеб. пособие по курсовому проектированию для студ. спец.
I-40 02 02 «Электронные вычислительные средства» дневной формы
обуч. / А.А. Петровский, В.Б. Ключ, А.Б. Давыдов, М.В. Качинский. –
Мн.: БГУИР, 2005. – 51 с.: ил.
ISBN 985-444-871-1

В учебном пособии рассмотрен системный подход к проектированию систем различного назначения на базе проблемно-ориентированных процессоров, приведено руководство по системному проектированию цифровых устройств на цифровых процессорах сигналов, содержатся требования и методические указания по выполнению курсового проекта по дисциплине «Проектирование проблемно-ориентированных вычислительных средств» для студентов специальности I-40 02 02 «Электронные вычислительные средства».

УДК 004.9 (075.8)
ББК 32.973.26-02 я 73

ISBN 985-444-871-1

© Коллектив авторов, 2005
© БГУИР, 2005

В В Е Д Е Н И Е

Дисциплина «Проектирование проблемно-ориентированных вычислительных средств» занимает ведущее место в подготовке специалистов в области проектирования специализированных электронных вычислительных средств (ЭВС). Цель дисциплины – формирование у студентов технического мышления, обучение будущих инженеров практическим навыкам проектирования систем различного назначения на базе проблемно-ориентированных микропроцессоров. Для закрепления этих навыков выполняется курсовой проект, который является необходимым условием для успешной работы над дипломным проектом.

Интенсивное развитие микроэлектроники привело к созданию различных специализированных микропроцессоров с малым уровнем потребления энергии, широкими функциональными возможностями и низкой стоимостью, которые можно встраивать в системы цифровой обработки сигналов (ЦОС). В результате может быть решена задача выпуска дешевой продукции с учетом индивидуальных интересов потребителей. На базе таких микропроцессорных средств можно создавать различные устройства и приборы – системы промышленной автоматики и управления, системы обработки аудио- и видеосигналов, системы связи, устройства автомобильной электроники и т. п. Для решения этой задачи необходимо готовить универсальных специалистов, умеющих применять универсальные и специализированные микропроцессорные средства и разрабатывать для них соответствующее программное обеспечение, следовательно, при работе над курсовым проектом основное внимание необходимо уделять системному подходу к проектированию, оптимизации принятых решений, обоснованию их соответствующими расчетами. Поэтому основным содержанием данного учебного пособия является руководство по системному проектированию проблемно-ориентированных микропроцессорных систем.

1. ЦЕЛИ И ЗАДАЧИ КУРСОВОГО ПРОЕКТИРОВАНИЯ

Выполнение курсового проекта по дисциплине «Проектирование проблемно-ориентированных вычислительных средств» имеет следующие цели и задачи:

- закрепить, углубить и систематизировать теоретические знания, полученные ранее при изучении схмотехнических дисциплин;
- приобрести практические навыки самостоятельного решения комплекса задач, связанных с проектированием проблемно-ориентированных вычислительных систем с использованием современной элементной базы, путем выполнения самостоятельной творческой разработки по заданному индивидуальному заданию;
- научить пользоваться специальной, справочной и другой нормативно-технической литературой, действующими стандартами;

– подготовить студента к дипломному проектированию и последующей самостоятельной работе по специальности.

2. ОСНОВНЫЕ ТРЕБОВАНИЯ К КУРСОВОМУ ПРОЕКТУ

2.1. Тематика курсового проектирования

Курсовой проект по дисциплине «Проектирование проблемно-ориентированных вычислительных средств» должен представлять собой самостоятельное решение комплексной инженерно-технической задачи, связанной с проектированием специализированных вычислительных систем для цифровой обработки сигналов и их программного обеспечения. Темы курсовых проектов должны быть посвящены проектированию микропроцессорных систем различного назначения, а также их системного и прикладного программного обеспечения. В качестве курсовых проектов могут выполняться научно-исследовательские работы, связанные с данной тематикой, в рамках госбюджетных и хоздоговорных НИР, заказов предприятий и организаций. Научно-исследовательские проекты должны содержать аналитический материал по решаемой проблеме, теоретический и экспериментальный разделы.

Объектом курсового проектирования является схемотехнически законченное микропроцессорное устройство. Тематика проектов – цифровая техника, устройства автоматики, управления и регулирования, автомобильная электроника, системы безопасности, измерительные устройства, системы связи, устройства цифровой обработки аудио- и видеосигналов и т.п.

Рекомендуются следующие типовые темы курсового проекта:

1. Разработка специализированного устройства для цифровой фильтрации или для спектрального анализа.
2. Разработка специализированного устройства цифрового модема.
3. Разработка специализированного устройства для формирования и анализа сигналов различного типа и назначения.
4. Разработка специализированного устройства ввода и обработки сигналов.
5. Разработка специализированного устройства для компрессии, декомпрессии, кодирования и декодирования данных.
6. Разработка специализированного устройства для управления различным оборудованием.

2.2. Задание на курсовое проектирование и исходные данные

Индивидуальное задание на курсовое проектирование выдается студенту в течение первой недели текущего семестра преподавателем – руководителем проекта. Задание оформляется на специальном бланке (прил.). Заданием на курсовое проектирование предусматривается разработка структурной организации специализированной вычислительной системы, разработка алгоритма ра-

боты, функциональной схемы вычислительной системы, ее программного обеспечения. В задание включаются:

- 1) тема проекта;
- 2) сроки сдачи студентом законченного проекта;
- 3) исходные данные к проекту;
- 4) содержание пояснительной записки (перечень вопросов, подлежащих разработке);
- 5) перечень графического материала;
- 6) календарный график работы над проектом.

В качестве исходных данных для выполнения курсового проекта задаются:

- назначение разрабатываемого устройства;
- объемы и организация внешней памяти;
- требования пользователя (выполняемые функции, устройства ввода/вывода, быстродействие, частотный диапазон, разрядность, погрешность, параметры входных и выходных сигналов и т.п.).

Перечень вопросов, подлежащих разработке в курсовом проекте, включает:

- постановку задачи;
- анализ задачи (функциональная спецификация системы: список функций, выполняемых системой, описание интерфейса между системой и пользователем);
- предварительное проектирование системы (разбиение системы на модули, выбор соотношения между аппаратными и программными средствами), построение структурной схемы аппаратной части системы, описание структурной схемы;
- проектирование аппаратных средств системы, разработку функциональной схемы системы, описание работы системы по функциональной схеме;
- проектирование программного обеспечения, разработку схемы алгоритма работы системы и программы на ассемблере, описание алгоритма работы системы и программы.

В состав графической части курсового проекта должны входить:

- 1) схема электрическая структурная устройства;
- 2) схема электрическая функциональная устройства;
- 3) схема алгоритма работы системы.

В зависимости от сложности объекта проектирования состав графической части проекта может изменяться.

2.3. Содержание и объем курсового проекта

Курсовой проект должен содержать: проектировочную часть, помещаемую в пояснительной записке, и графическую часть, оформленную в виде комплекта схем.

При изложении материала в пояснительной записке рекомендуется придерживаться следующего расположения разделов:

- титульный лист;
- задание на курсовое проектирование;
- содержание;
- введение;
- анализ задачи;
- предварительное проектирование системы;
- проектирование аппаратных средств системы;
- проектирование программного обеспечения;
- выводы и заключение;
- список литературы;
- приложения.

Общий объем пояснительной записки должен составлять 30-40 страниц. Графическая часть проекта включает комплект схем на разрабатываемое устройство объемом не менее трех листов соответствующего формата.

2.4. Организация курсового проектирования и защита проекта

На выполнение курсового проекта по дисциплине «Проектирование проблемно-ориентированных вычислительных средств» отводится примерно 13 учебных недель. Работа над курсовым проектом является самостоятельной работой студента, проводимой под контролем руководителя проектирования. В установленное расписанием время студент консультируется у своего руководителя. Время работы над курсовым проектом разбивается на 6 этапов (прил.). Объем выполненной студентом работы по каждому этапу оценивается руководителем проектирования в процентах от общего объема проектирования. **Явка на опрочентовки студентов строго обязательна.** Оформленный курсовой проект сдается студентом руководителю на проверку не позднее чем за неделю до назначенного срока защиты и после проверки может быть представлен к защите.

Защита курсового проекта проводится перед комиссией и включает в себя доклад в течение 4-6 минут и ответы на вопросы членов комиссии. В докладе студент должен изложить задачи проектирования и их реализацию, дать обоснование принятых технических решений, кратко охарактеризовать каждый лист графического материала. Принятые в курсовом проекте технические решения должны сопровождаться выводами. В заключении студент должен отразить степень соответствия разработанного проекта требованиям задания на проектирование. Студенту на защите могут задаваться любые вопросы по теме курсового проекта. По выступлению студента и его ответам на вопросы комиссия судит о его способностях правильно и доходчиво излагать результаты своей работы, поэтому выступление рекомендуется подготовить заранее. Общая продолжительность защиты с учетом ответов на вопросы составляет 10-15 минут.

3. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ КУРСОВОГО ПРОЕКТА

3.1. Общие указания

Методические указания имеют цель облегчить студентам выполнение проекта, намечают основные направления работы над курсовым проектом. При этом не исключается самостоятельный выбор студентом путей решения тех или иных задач курсового проекта, применения других обоснованных методов проектирования.

Рекомендуется начинать проектирование с составления плана работы над проектом, в котором необходимо предусмотреть изучение литературы, повторение некоторых дисциплин, разработку аппаратной и программной частей устройства, оформление пояснительной записки, выполнение схем и т.д. Студент самостоятельно составляет развернутый календарный план выполнения курсового проекта с учетом контрольных точек, оговоренных заранее руководителем проекта. В этом случае обеспечивается самоконтроль, равномерное распределение нагрузки по этапам проектирования, что оказывает положительное влияние на качество разработки и облегчает работу.

Разработка устройства должна оцениваться с точки зрения эффективности затрат. Проектирование аппаратной части и программного обеспечения должно проводиться на системной основе с целью минимизации как стоимости проектирования, так и времени, затрачиваемого на разработку. Для лучшего понимания этих положений необходимо рассмотреть цикл проектирования микропроцессорной системы [1].

Первый шаг цикла проектирования системы включает определение набора требований пользователя и построение функциональной спецификации, вытекающей из требований пользователя. Требования пользователя определяют, что пользователь хочет от системы и что она должна делать. Функциональная спецификация определяет функции, которые система должна выполнять для пользователя после завершения проектирования. Они включают описания форматов данных как на входе, так и на выходе системы, а также внешние условия, управляющие действиями системы. Функциональная спецификация и требования пользователя являются критериями оценки функциональных характеристик системы после завершения проектирования. Требования пользователя определяют, что пользователь хочет от системы, а функциональная спецификация фиксирует, что система должна делать и как она взаимодействует с окружением. Как только функциональная спецификация определена, она используется вместе с требованиями пользователя в качестве основы для проектирования системы. По этой причине важно, чтобы как требования пользователя, так и функциональная спецификация были не только полными и точными, но также четкими и легко усваиваемыми.

Следующим шагом является проектирование системы на основе функциональной спецификации. Для системы, содержащей микропроцессор или цифро-

вой процессор сигналов (ЦПС), требуется проектирование как аппаратных, так и программных средств. Необходимо определить организацию аппаратной и программной частей системы, из каких компонентов должна состоять система и как эти части должны быть взаимосвязаны. Проектирование аппаратной части может быть выполнено с использованием стандартной методологии проектирования аппаратуры. Проектирование программной части лучше всего может быть выполнено с использованием языка программирования, ориентированного на выбранный ЦПС. Программное обеспечение тестируется и одновременно с аппаратурой объединяется в единое целое, после чего оцениваются эксплуатационные характеристики системы. Цикл проектирования такой системы показан на рис. 3.1. Две части системы часто разрабатываются параллельно, что на рисунке выглядит в виде отдельных ветвей.

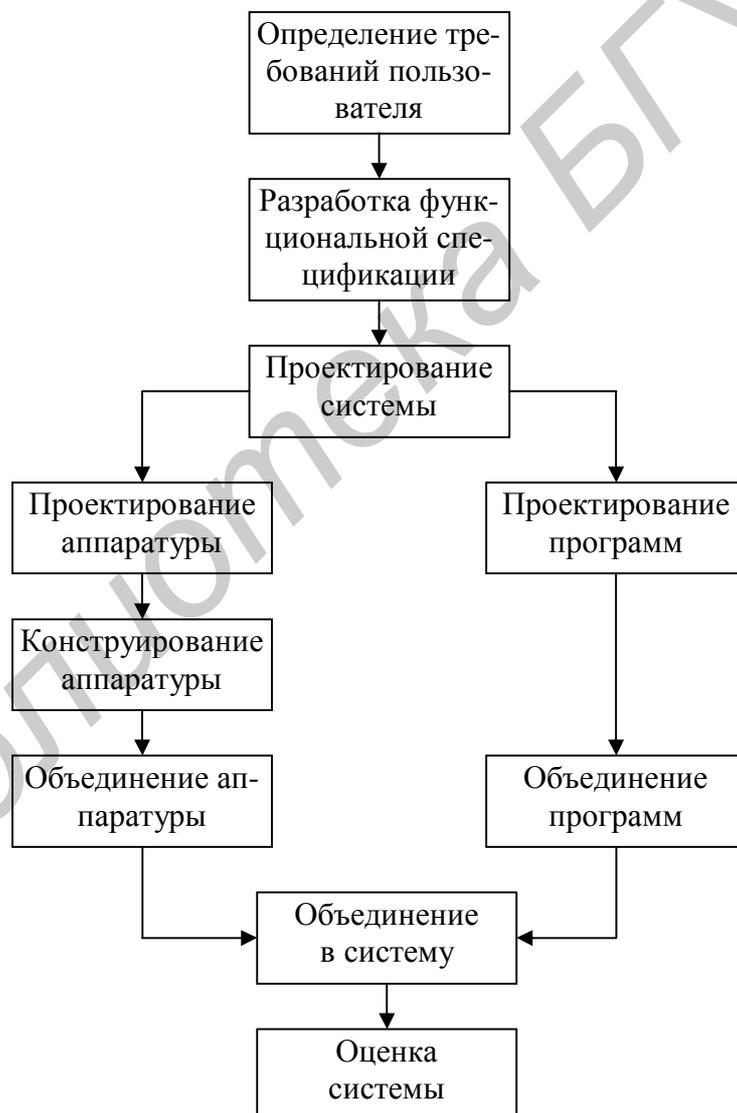


Рис. 3.1. Цикл проектирования системы, содержащей микропроцессор

3.2. Требования к построению и содержанию пояснительной записки

Ниже приводятся требования к последовательности разделов и содержанию пояснительной записки, вытекающие из логической последовательности процесса системного проектирования ЭВС. При рассмотрении требований будем использовать пример проектирования системы диагностики двигателя компрессора холодильника, которая должна выполнять следующие действия:

- 1) включать и выключать двигатель;
- 2) задавать нагрузку на двигатель;
- 3) обеспечивать измерение скорости вращения двигателя;
- 4) обеспечивать измерение напряжения и потребляемого тока;
- 5) вычислять полную, активную и реактивную мощность.

3.2.1. Введение

В этом разделе пояснительной записки должна быть выполнена постановка задачи проектирования. В начале раздела необходимо кратко характеризовать современное состояние задачи, решению которой посвящен курсовой проект. Остальная часть раздела должна быть посвящена выработке требований пользователя.

Требования пользователя определяют, чего хочет или в чем нуждается пользователь или потребитель. Требования пользователя могут быть получены во время встреч с пользователем или покупателем с целью выявления его нужд и определения того, что пользователь хочет от системы. Другой подход используется при планировании ассортимента изделий, когда требования пользователей могут быть определены путем изучения рынка сбыта на основе спроса покупателей.

На первом шаге необходимо сформулировать вопросы, ответы на которые позволят получить информацию, касающуюся того, что система должна делать, т.е. определить требования пользователя. Эти вопросы должны быть связаны только с тем, что должна делать система диагностики двигателей. В частности, для системы диагностики двигателей необходимо получить ответ на следующие вопросы:

- 1) как осуществляется управление двигателем?
- 2) с какой точностью и в каких пределах измеряется скорость двигателя?
- 3) с какой точностью и в каких пределах измеряются ток и напряжение?
- 4) в каком виде необходимо отображать результаты измерений и вычислений?

Из ответов на эти вопросы можно сделать вывод относительно того, что будут из себя представлять требования пользователя. Для системы диагностики двигателей набор соответствующих требований пользователя приведен ниже:

- 1) система предназначена для диагностики двигателей компрессоров;
- 2) система с помощью цифрового кода должна включать и выключать двигатель и задавать нагрузку на двигатель;

- 3) система должна измерять число оборотов двигателя в диапазоне от 0 до 3000 оборотов в минуту с точностью до 1 оборота;
- 4) система должна измерять ток и напряжение с точностью не ниже 0,2%;
- 5) измеряемое на входе системы напряжение, пропорциональное току и напряжению на двигателе, изменяется в диапазоне $\pm 10\text{В}$;
- 6) система должна рассчитывать полную, активную и реактивную мощность, потребляемую двигателем;
- 7) система должна отображать изменение числа оборотов, тока, напряжения и мощности в графическом виде;
- 8) система должна сопрягаться с персональным компьютером.

На основе этих требований можно определить функциональную спецификацию системы.

3.2.2. Анализ задачи

В этом разделе пояснительной записки должны быть тщательно проанализированы требования пользователя с целью определения функциональной спецификации проектируемой системы.

Функциональная спецификация должна определять, какие функции должны выполняться для удовлетворения требований пользователя и обеспечения интерфейса между системой и окружением. Функциональная спецификация включает два основных компонента:

- 1) список функций, выполняемых системой;
- 2) описание интерфейса между системой и пользователем.

Так как система проектируется на основе информации, содержащейся как в требованиях пользователя, так и в функциональной спецификации, важно, чтобы функции, которые должны отображать требуемое поведение системы, были описаны достаточно подробно. По отношению к требованиям пользователя системы диагностики двигателей функциональная спецификация должна давать ответы на следующие вопросы:

- 1) какие средства необходимо предусмотреть для включения двигателя?
- 2) какие средства необходимо предусмотреть для выключения двигателя?
- 3) какие средства необходимо предусмотреть для управления нагрузкой на двигатель?
- 4) какие средства необходимо предусмотреть для измерения скорости двигателя?
- 5) какие средства необходимо предусмотреть для измерения тока?
- 6) какие средства необходимо предусмотреть для измерения напряжения?
- 7) какие средства необходимо предусмотреть для управления системой?
- 8) какие средства необходимо предусмотреть для отображения информации?

Ответив на эти вопросы, можно составить функциональную спецификацию для системы диагностики двигателя. В рассматриваемой системе:

1) для включения двигателя необходимо использовать один цифровой выход;

2) для выключения двигателя можно использовать тот же цифровой выход (1 – включение, 0 – выключение) или отдельный выход;

3) для управления пятиступенчатой нагрузкой необходимо 3 или 5 цифровых выходов и таким образом система должна иметь от четырех до семи цифровых выходов;

4) для измерения скорости двигателя необходим датчик и аналого-цифровой преобразователь нужной разрядности и быстродействия;

5) для измерения тока необходим преобразователь тока в напряжение и аналого-цифровой преобразователь нужной разрядности и быстродействия;

6) для измерения напряжения необходим делитель напряжения и аналого-цифровой преобразователь нужной разрядности и быстродействия;

7) для управления системой необходим интерфейс с компьютером;

8) для отображения информации может использоваться экран монитора персонального компьютера.

Эти ответы содержат информацию, необходимую для определения функциональной спецификации. Если распределить эту информацию по категориям ВХОДЫ, ВЫХОДЫ и ФУНКЦИИ, функциональную спецификацию системы диагностики двигателя можно представить в следующем виде.

Функциональная спецификация системы диагностики двигателя

1. ВХОДЫ.

1.1. Вход датчика скорости.

1.2. Входы для измерения тока и напряжения.

1.3. Регистр команд со стороны компьютера.

2. ВЫХОДЫ.

2.1. Сигнал включения/выключения двигателя.

2.2. Сигналы управления нагрузкой.

2.3. Регистр данных для выдачи в компьютер.

3. ФУНКЦИИ.

3.1. Включение и выключение двигателя компрессора.

3.2. Установка заданной нагрузки на двигатель.

3.3. Измерение числа оборотов двигателя.

3.4. Измерение тока и напряжения.

3.5. Вычисление мощности.

3.6. Передача измеренных значений в компьютер.

Важным вопросом при разработке функциональной спецификации является взаимодействие между пользователем и системой. В системе диагностики двигателя взаимодействие между системой и пользователем осуществляется с помощью регистров команд, данных и состояния (связь с персональным компьютером), цифрового выходного порта и аналого-цифровых преобразователей (рис. 3.2).

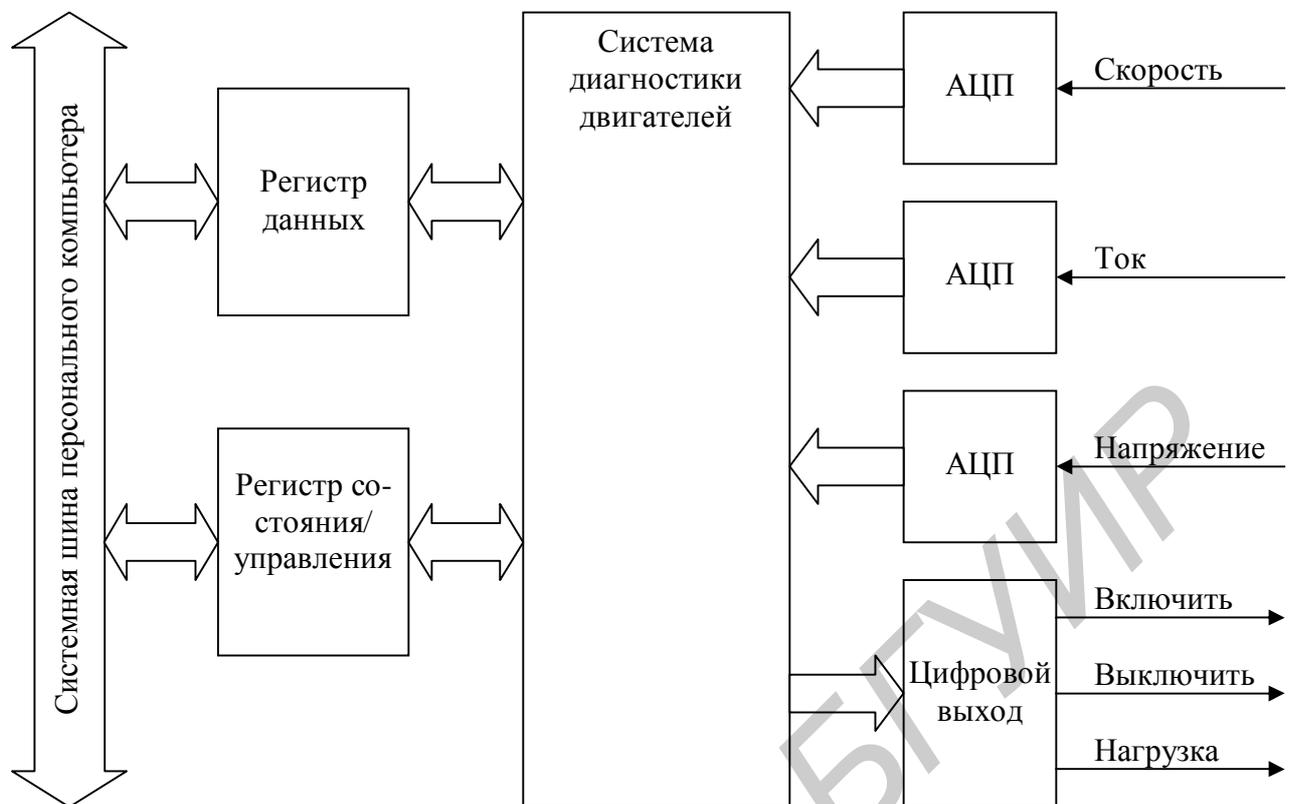


Рис. 3.2. Интерфейс между системой и пользователем

В общем случае могут существовать различные способы взаимодействия между пользователем и системой.

Тактильные (контактные) входы. Пользователь может нажимать кнопки, поворачивать ручки, манипулировать рычагами и т.д.

Визуальные выходы. Системы используют для вывода информации лампы, светодиоды, цифровые индикаторы, ЖКИ-экраны, печатающие устройства и т.д.

Звуковые выходы. Сигналы звуковой тревоги, зуммер могут использоваться для оповещения оператора о каких-то специальных ситуациях.

Аналоговые входы. В системе диагностики двигателя аналоговый ввод информации осуществляется датчиками скорости, тока, напряжения.

Цифровые входы. В системах могут использоваться цифровые входы от различных датчиков с двумя состояниями.

Аналоговые выходы. В качестве аналоговых выходов в микропроцессорных системах используются цифроаналоговые преобразователи.

Цифровые выходы. В системе диагностики двигателя цифровые выходы используются для управления исполнительными механизмами (включение/выключение двигателя, задание нагрузки).

Цифровые входы и выходы в микропроцессорных системах часто используются для организации межпроцессорного взаимодействия. В системе диагностики двигателя персональный компьютер используется для организации ин-

терфейса с оператором системы (ввод с клавиатуры, вывод на экран, запись на диск и т.п.)

Учет человеческого фактора при проектировании должен приводить к простоте системы и легкости ее использования. Эти цели достигаются посредством проектирования соответствующего системного интерфейса, обеспечивающего экономный обмен информацией между пользователем и системой. Это позволяет уменьшить затраты на обучение пользователя и не требует больших усилий при взаимодействии с системой. Человеческий фактор в вопросах проектирования микропроцессорных систем проявляется также в том, что некоторые характеристики существующих систем должны быть учтены при проектировании новых систем. Например, инженер, который использует генератор сигналов, привык вращать ручки для установки амплитуды и частоты формируемого сигнала. Поэтому, даже если эти функции находятся теперь под управлением микропроцессора, проектировщик должен оставить ручки, чтобы пользователю не пришлось переучиваться для того, чтобы выполнять хорошо знакомые ему функции, особенно если эти ручки хорошо спроектированы с точки зрения человека.

Рассмотренная система диагностики двигателя является достаточно простой (часть функций возложена на персональный компьютер), поэтому определение для нее функциональной спецификации не составило особого труда. В более сложных системах определение функциональной спецификации может оказаться достаточно непростой задачей. В таких случаях удобнее сначала разбить систему на несколько подсистем меньшего размера, а затем определить функциональную спецификацию для каждой из них.

3.2.3. Предварительное проектирование системы

В этом разделе должно быть произведено предварительное проектирование системы и осуществлен выбор соотношения между аппаратными и программными средствами. Вначале необходимо на основе функциональной спецификации определить набор модулей, реализующих выполняемые в системе функции. Этот набор функциональных модулей составляет первый уровень проектирования систем. После разбиения системы на модули надо отделить аппаратные модули от программных. В результате этого строится модульная структура аппаратных средств системы, которая оформляется в виде электрической структурной схемы.

Прежде чем начинать детальное проектирование, необходимо определить, какие функции лучше выполняются с помощью программного обеспечения системы, а какие – с помощью аппаратных средств. Все функции предварительно должны быть распределены между программными и аппаратными средствами. Часть функций может быть реализована с помощью как аппаратных, так и программных средств. Например, в системе диагностики двигателя необходимо задавать частоту дискретизации сигналов, и, следовательно, преду-

смотреть функцию генератора частоты. Эта функция может быть реализована одним из следующих способов:

- аппаратным способом, используя в качестве генератора частоты конфигурацию логических элементов на интегральных схемах;
- комбинированным способом, используя программируемый таймер на интегральной схеме, т.е. путем комбинации аппаратных и программных средств.

От выбора решения зависит, как проектировать аппаратные и программные средства системы. Во время детального проектирования аппаратных и программных средств часто становится очевидным, что некоторые аппаратные функции могут быть лучше выполнены с помощью программных средств и наоборот. Таким образом, во время последующих стадий процесса проектирования может иметь место модификация предварительного проектного решения.

Основу аппаратных средств системы составляет специализированная микроЭВМ, которая в общем случае включает (рис. 3.3):

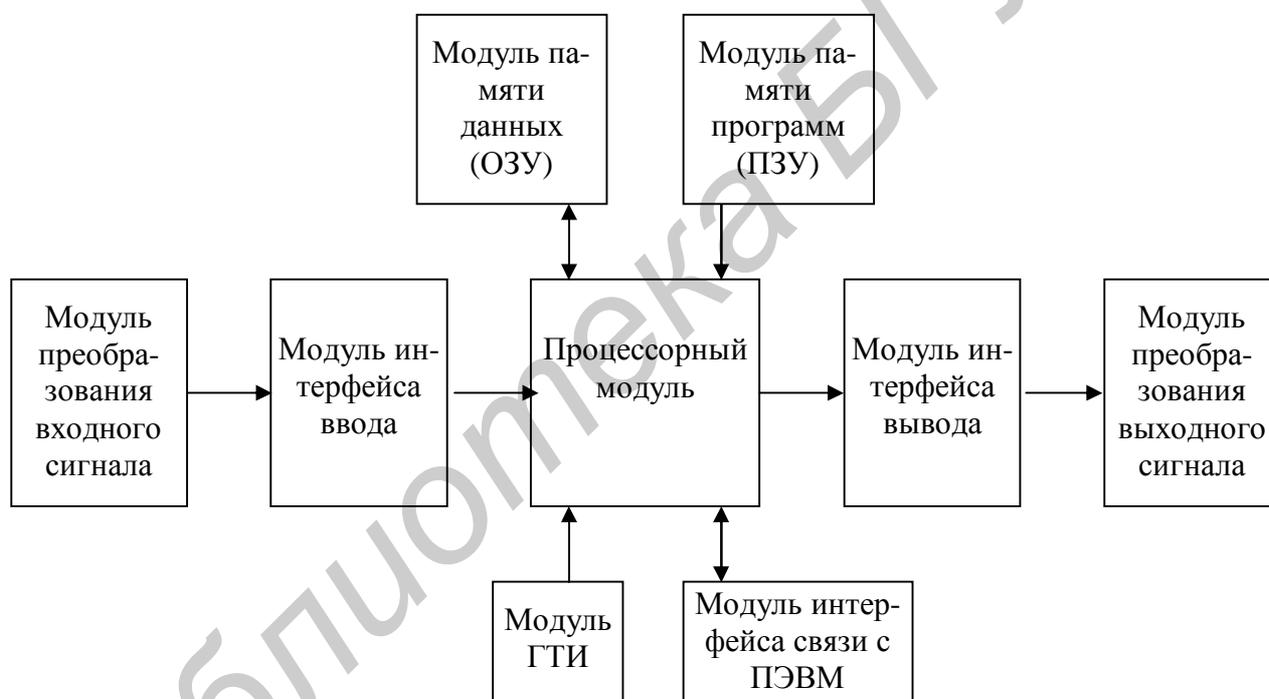


Рис. 3.3. Общая модульная структура аппаратных средств микропроцессорной системы

- модуль ЦИФРОВОГО ПРОЦЕССОРА СИГНАЛОВ (ЦПС);
- модуль ГЕНЕРАТОРА ТАКТОВЫХ ИМПУЛЬСОВ (ГТИ);
- модуль ПАМЯТИ ДАННЫХ;
- модуль ПАМЯТИ ПРОГРАММ;
- модуль ИНТЕРФЕЙСА СВЯЗИ С ПЭВМ;

– модуль ИНТЕРФЕЙСА ВВОДА и модуль ИНТЕРФЕЙСА ВЫВОДА, содержащие интерфейсные компоненты, необходимые для связи модуля ЦПС с другими модулями системы;

– модуль ПРЕОБРАЗОВАНИЯ ВХОДНОГО СИГНАЛА и модуль ПРЕОБРАЗОВАНИЯ ВЫХОДНОГО СИГНАЛА, которые содержат компоненты, необходимые для обмена входными и выходными сигналами с внешним окружением (различные датчики, детекторы, переключатели, кнопки, клавиатуры, устройства световой и звуковой сигнализации, цифровые индикаторы, аналого-цифровые (АЦП) и цифроаналоговые (ЦАП) преобразователи и т.п.).

Остальные модули системы могут быть реализованы с помощью как аппаратных, так и программных средств. Конкретное распределение модулей между аппаратной и программной частями системы зависит от ее целевого назначения (решаемой задачи). Модули, реализуемые с помощью аппаратных средств, совместно со специализированной микроЭВМ образуют аппаратную часть системы.

Рассмотрим разбиение системы диагностики двигателя на функциональные модули. Из рассмотрения функциональной спецификации следует, что система состоит из четырех частей: ВХОД, ВЫХОД, СВЯЗЬ и ФУНКЦИИ. В данном случае каждая из частей ВХОД и ВЫХОД может быть реализована в одном модуле, поскольку они являются относительно простыми:

– ВХОДНОЙ модуль (модуль датчиков). Выполняет считывание состояния датчиков оборотов, тока и напряжения и их хранение;

– ВЫХОДНОЙ модуль (модуль управления двигателем). Включение/выключение двигателя и задание нагрузки;

– модуль СВЯЗИ (модуль взаимодействия с ПЭВМ). Прием команд из ПЭВМ и выдача результатов измерений.

В более сложных системах может оказаться необходимым дальнейшее разбиение этих модулей на несколько входных и выходных.

Для части функциональной спецификации ФУНКЦИИ можно выделить шесть различных модулей.

1. Модуль ОЖИДАНИЯ. Выполняет проверку состояния регистра команд и ожидание команды из ПЭВМ, чтобы предпринять соответствующие действия.

2. Модуль ТАЙМЕРА. Выполняет управление таймером: запускает и останавливает таймер, задает период таймера и обрабатывает прерывания от таймера.

3. Модуль ВВОДА ДАННЫХ. Выполняет ввод измеряемых величин.

4. Модуль ВЫЧИСЛЕНИЙ. Выполняет расчет мощности по введенным значениям тока и напряжения.

5. Модуль УПРАВЛЕНИЯ. Выполняет включение и выключение двигателя и задание нагрузки на двигатель.

6. Модуль ВОССТАНОВЛЕНИЯ. Выполняет инициализацию системы (аппаратных и программных средств) при начальном включении (включение питания), восстановление системы после сбоя.

После разбиения на функциональные модули строится функционально-модульная структура системы диагностики двигателя. Каждый из модулей размещается на одном из четырех уровней нисходящей иерархии. Так как микропроцессор является последовательным устройством, которое выполняет операции одну за другой, то на высшем уровне модульной структуры должна находиться управляющая функция, обеспечивающая последовательное исполнение системой других функций. Средства, необходимые для реализации управляющей функции, содержит ИСПОЛНИТЕЛЬНЫЙ модуль. ВХОДНОЙ модуль, ВЫХОДНОЙ модуль и модуль ТАЙМЕРА располагаются на самом нижнем уровне. Модуль ВВОДА ДАННЫХ, модуль ВЫЧИСЛЕНИЯ и модуль УПРАВЛЕНИЯ должны находиться на более низком уровне по отношению к модулю ОЖИДАНИЯ и модулю ВОССТАНОВЛЕНИЯ. Полученная таким образом модульная структура системы диагностики двигателя показана на рис. 3.4.



Рис. 3.4. Функционально-модульная структура системы диагностики двигателя

Каждый модуль состоит из набора функций, принадлежащих этому модулю. Способ распределения функций по конкретным модулям зависит от личного предпочтения разработчика. Если модуль содержит слишком много функций, он может быть разделен на несколько модулей. С целью упрощения нескольких модулей, каждый из которых содержит небольшое число функций, могут быть объединены в один модуль. Модульная структура позволяет легко добавлять функции к системе, а также изменять или удалять часть системы в любой момент цикла ее проектирования.

Рассмотрим, как распределяются функции по модулям системы диагностики двигателя (рис. 3.5).

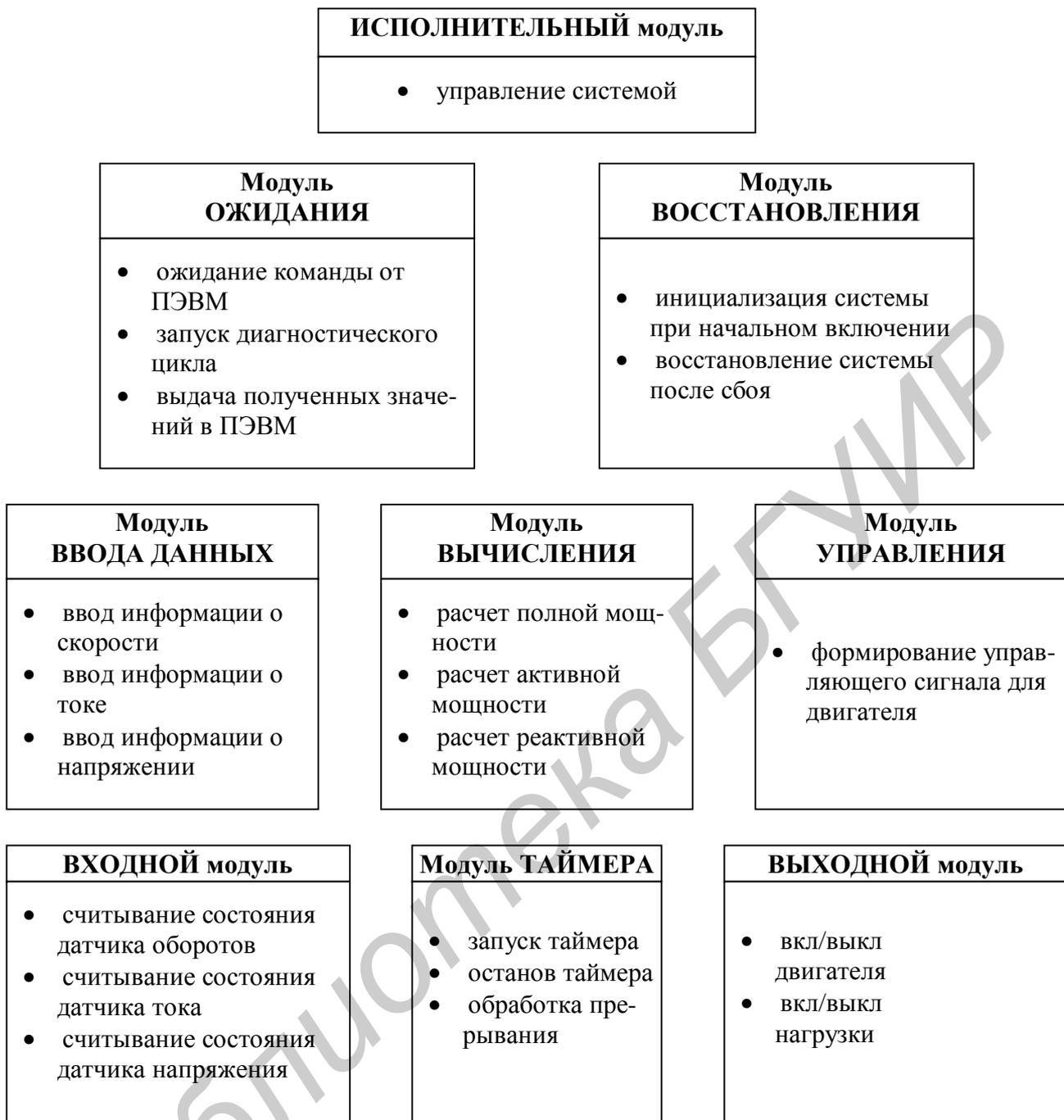


Рис. 3.5. Функционально-модульная структура системы диагностики двигателя, включающая функции для каждого модуля

1. **ИСПОЛНИТЕЛЬНЫЙ** модуль состоит из единственной управляющей функции.

2. **ВХОДНОЙ** модуль включает:

- считывание состояния датчика оборотов;
- считывание состояния датчика тока;
- считывание состояния датчика напряжения.

3. **ВЫХОДНОЙ** модуль включает:

- включение/выключение двигателя;

- включение/выключение нагрузки.
- 4. Модуль ТАЙМЕРА включает:
 - запуск таймера;
 - останов таймера;
 - обработку прерывания таймера.
- 5. Модуль ВВОДА ДАННЫХ включает:
 - ввод в память информации о скорости двигателя;
 - ввод в память информации о токе;
 - ввод в память информации о напряжении.
- 6. Модуль ВЫЧИСЛЕНИЯ включает:
 - расчет полной мощности;
 - расчет активной мощности;
 - расчет реактивной мощности.
- 7. Модуль УПРАВЛЕНИЯ состоит из единственной функции формирования управляющего сигнала для двигателя.
- 8. Модуль ОЖИДАНИЯ включает:
 - ожидание команды от ПЭВМ;
 - запуск диагностического цикла;
 - выдачу полученных и рассчитанных значений в ПЭВМ.
- 9. Модуль ВОССТАНОВЛЕНИЯ включает:
 - инициализацию системы при начальном включении (включение питания);
 - восстановление системы после сбоя.

Система диагностики двигателя является относительно простой и поэтому содержит небольшое число модулей. В более сложных системах модулей может быть значительно больше. Как указывалось выше, более сложные системы разбиваются на подсистемы, при этом функциональную спецификацию разрабатывают для каждой подсистемы. Поэтому разбиение на модули и проектирование каждой подсистемы должно выполняться так, как это делается для небольшой системы.

Рассмотрим соотношение между аппаратными и программными средствами системы диагностики двигателя.

Функция управления системой реализуется специализированной микро-ЭВМ в результате выполнения основной (управляющей) программы путем последовательного вызова функций соответствующих программных модулей системы.

Наличие двух аппаратных таймеров в процессоре позволяет независимо задавать частоту дискретизации для АЦП и ЦАП.

3.2.4. Проектирование аппаратных средств системы

В этом разделе должно быть произведено проектирование аппаратной части системы, осуществлен выбор микросхем памяти и разработана электриче-

ская функциональная схема, приведено описание работы системы по функциональной схеме.

Вопросы проектирования аппаратных средств микропроцессорной системы будут рассмотрены в следующем разделе данного учебного пособия.

3.2.5. Проектирование программного обеспечения

В данном разделе пояснительной записки должно быть выполнено проектирование прикладного программного обеспечения, разработана схема алгоритма работы системы и программы на ассемблере, приведено описание алгоритма работы системы и программы.

Вопросы проектирования программного обеспечения микропроцессорной системы будут рассмотрены в разд. 5 данного учебного пособия.

3.2.6. Выводы и заключение

Этот раздел должен содержать оценку результатов работы, в частности, с точки зрения их соответствия заданию на проектирование и требованиям пользователя. Кроме того, необходимо указать, чем закончена работа, а также наметить пути и цели дальнейшего совершенствования устройства.

3.2.7. Список литературы

В разделе приводится перечень книг, пособий, статей, Web-страниц, технической документации и нормативно-технических документов, которые использовались при выполнении курсового проекта.

3.2.8. Приложения

В приложения включаются графический материал в соответствии с заданием на проектирование, исходный текст программы на языке ассемблера, краткие технические характеристики компонентов (микросхем, датчиков, элементов индикации, исполнительных устройств и т.п.), используемых в системе.

4. ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ АППАРАТНЫХ СРЕДСТВ МИКРОПРОЦЕССОРНОЙ СИСТЕМЫ

Проектирование аппаратных средств микропроцессорной системы предполагает:

- проектирование микропроцессорного блока, включающего ПРОЦЕССОРНЫЙ модуль, модуль ГТИ, модули ПАМЯТИ, а также, возможно, и другие модули, например, модуль ТАЙМЕРА, модуль КОНТРОЛЛЕРА ПЕРЕРЫВАНИЙ и т.п.;

- проектирование интерфейса между микропроцессорным блоком и центральным (главным) процессором;

- проектирование интерфейса между микропроцессорным блоком и внешним окружением системы (периферийным оборудованием).

4.1. Проектирование микропроцессорного блока

Как видно из рассмотренного выше, разрабатываемые в курсовом проекте микропроцессорные системы по своей сути являются системами цифровой обработки сигналов (ЦОС) – основной функцией, реализуемой в них, является ввод, преобразование и обработка аналоговых сигналов с выдачей информации в цифровой или в аналоговой форме. Микропроцессорная система ЦОС содержит специализированную микроЭВМ, устройства связи с объектом (с датчиками и исполнительными устройствами) и периферийные устройства. Такие системы наиболее эффективно могут быть построены с помощью специализированных цифровых процессоров сигналов (ЦПС), ориентированных на выполнение цифровой обработки сигналов (цифровая фильтрация, быстрое преобразование Фурье и т.п.). ЦПС представляют собой приборы, конструктивно выполненные в виде БИС и включающие в себя все составные части обычной микроЭВМ: микропроцессор, память программ и данных, а также программируемые интерфейсные схемы для связи с внешней средой. Использование ЦПС в таких системах обеспечивает достижение высоких показателей эффективности при очень низкой стоимости – во многих применениях система может состоять только из одной БИС ЦПС. ЦПС характеризуется относительно небольшой емкостью внутренней памяти, физическим разделением памяти программ (ПЗУ, ОЗУ, флэш-память) и памяти данных (ОЗУ), ориентированной на задачи ЦОС системой команд, простыми способами адресации команд и данных, а также специфической организацией ввода/вывода. Все это предопределяет область их использования в качестве специализированных вычислителей, включенных в контур управления объектом или процессом. Структурная организация, набор команд и аппаратно-программные средства ввода/вывода ЦПС лучше всего приспособлены для решения задач ЦОС в приборах, устройствах и системах обработки данных. ЦПС не являются машинами классического фон-неймановского типа, так как память физически разделена на память программ и память данных. Наличие ОЗУ, используемого в качестве памяти программ, дает возможность модификации и/или замены прикладных программ ЦПС во время работы.

Типовая структура (рис. 4.1) системы диагностики, построенной на базе ЦПС, состоит из диагностируемого объекта, специализированного процессора ЦОС и аппаратуры их взаимной связи.

Процессор ЦОС путем периодического опроса осведомительных слов (ОС) формирует в соответствии с заданным алгоритмом последовательности управляющих слов (УС). Осведомительные слова – это сигналы состояния (СС) объекта, сформированные датчиками диагностируемого объекта, и флаги. Выходные сигналы датчиков вследствие их различной физической природы могут потребовать промежуточного преобразования на АЦП или на схемах формирователей сигналов (ФС), которые чаще всего выполняют функции гальванической развязки и формирования необходимых уровней сигналов.

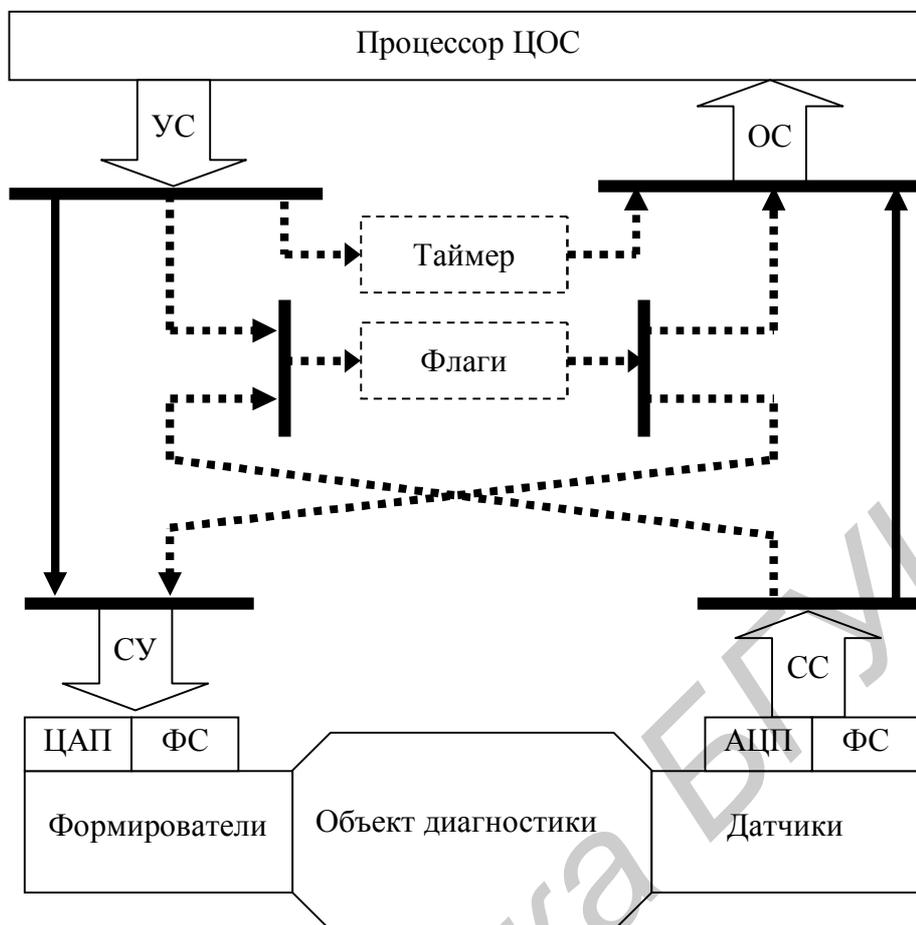


Рис. 4.1. Структура цифровой системы диагностики на основе процессора ЦОС

ЦПС с требуемой периодичностью обновляет управляющие слова на своих выходных портах. Некоторая часть управляющего слова интерпретируется как совокупность прямых двоичных сигналов управления (СУ), которые через схемы формирователей сигналов (усилители мощности, реле, оптроны и т.п.) поступают на исполнительные устройства и устройства индикации. Другая часть управляющего слова представляет собой двоичные коды, которые через ЦАП воздействуют на исследуемый объект аналогового типа. Если исследуемый объект использует цифровые датчики и цифровые исполнительные устройства, то АЦП и ЦАП будут отсутствовать.

В состав аппаратуры связи может входить регистр флагов, на котором фиксируется некоторое множество признаков как исследуемого объекта, так и процесса работы ЦПС. Регистр флагов используется в качестве аппаратного средства реализации механизма взаимной синхронизации относительно медленных и асинхронных процессов в исследуемом объекте или ПЭВМ и быстрых процессов в ЦПС. Регистр флагов доступен как ЦПС, так и датчикам или ПЭВМ. Поэтому он является удобным местом фиксации сигналов *готов/ожидание* при передачах с квитированием или *запрос прерывания/подтверждение* при взаимодействии ЦПС и объекта или ПЭВМ в режиме прерывания.

Для аппаратной реализации временных задержек, формирования сигналов требуемой частоты и скважности используют программируемые таймеры. Если в ЦПС их число недостаточно, программируемые таймеры включают в состав аппаратуры связи.

Функционирование системы на базе ЦПС полностью определяется прикладной программой, размещаемой в памяти программ ЦПС. Поэтому специализация системы типовой структуры на решение конкретной задачи ЦОС осуществляется путем разработки соответствующих прикладных программ и аппаратуры связи ЦПС с датчиками и исполнительными устройствами объекта.

Применение ЦПС в устройствах ЦОС привело к кардинальным изменениям в разработке аппаратных средств микропроцессорных систем. Поскольку ЦПС представляет собой функционально законченную однокристалльную микроЭВМ, то построение микропроцессорного блока полностью определяется выбранным типом ЦПС. Конкретный ЦПС допускает очень немного вариантов его системного включения, которые приводятся в фирменной технической документации. В свою очередь конкретный вариант включения выбранного ЦПС полностью определяется решаемой прикладной задачей (точнее методом ее решения). Таким образом, основным вопросом при проектировании микропроцессорного блока является выбор типа используемого ЦПС.

При выборе ЦПС необходимо определить:

- тип операций, используемых при решении прикладной задачи;
- емкость ОЗУ;
- емкость ПЗУ (флэш-памяти);
- необходимость использования функциональных преобразователей;
- число портов ввода/вывода и их назначение;
- число и тип запросов прерывания;
- необходимость использования таймера.

В каждом конкретном случае разработчик должен определить, какая часть названных средств и в каком объеме реализована в том или ином типе ЦПС, т.е. в какой степени ЦПС может удовлетворить поставленной задаче.

При выборе ЦПС важно правильно определить число и назначение портов ввода/вывода. Параметры и число портов ввода/вывода задаются особенностями внешних устройств, имеющих в составе ЦПС и подключаемых к нему. Для подключения внешних устройств могут использоваться как последовательные, так и параллельные порты ввода/вывода.

4.2. Проектирование внешней памяти микропроцессорного блока

В курсовом проекте по заданию к ЦПС необходимо подключить внешнюю память программ и данных заданного объема и конфигурации.

Поскольку процессоры семейства TMS320 обмениваются с памятью только 16-разрядными словами, то при использовании микросхем памяти с разрядностью 8 бит данных необходимо параллельное подключение как минимум двух микросхем. При необходимости получить большой объем памяти используются несколько пар микросхем, каждая из которых выбирается сигналами с дешифратора адресов. Например, для получения 8К слов ПЗУ программ на микросхемах с организацией 4К×8 требуется две пары микросхем, а для получения 6К слов ОЗУ данных на микросхемах с организацией 2К×16 потребуется три микросхемы. Схема подключения памяти данной конфигурации к процессору может выглядеть так, как показано на рис. 4.2.

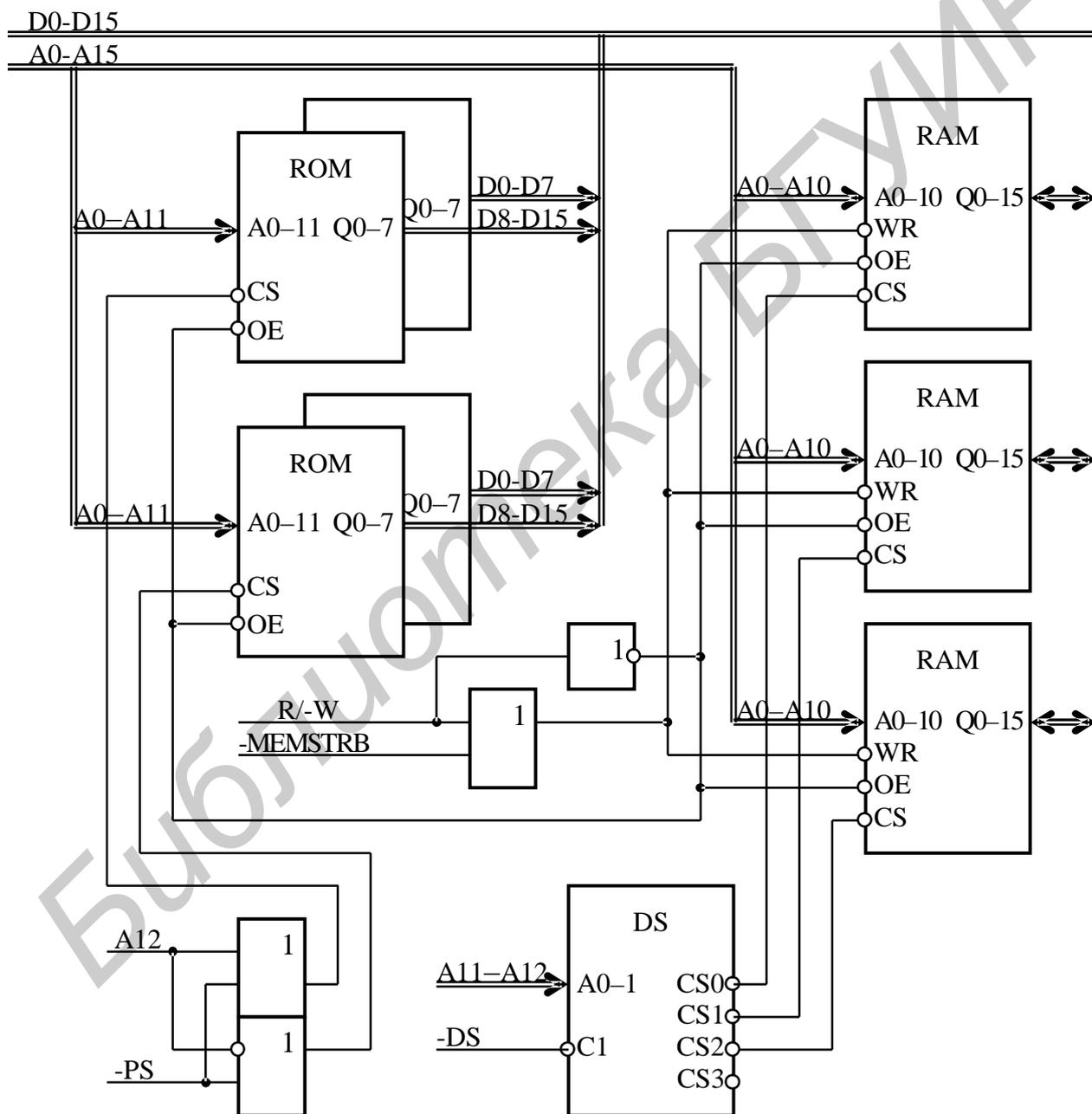


Рис. 4.2. Подключение внешней памяти программ и данных к ЦПС

Выборка конкретной микросхемы или пары микросхем осуществляется сигналами с дешифраторов адреса. При этом младшие разряды шины адреса подаются непосредственно на микросхемы памяти, следующие разряды – на дешифратор, а самые старшие разряды адреса могут не использоваться.

Простейший дешифратор на два выхода может быть построен на двух обычных элементах ИЛИ, как показано на рисунке для памяти программ. Для выбора из нескольких микросхем предпочтительнее использовать специальную микросхему дешифратора, как показано на рисунке для памяти данных. Каждый из дешифраторов стробируется соответствующим сигналом выбора адресного пространства памяти программ (-PS) или данных (-DS).

Первая пара микросхем ПЗУ выбирается при обращении к памяти программ (низкий уровень сигнала -PS) по адресам с низким уровнем сигнала A12, а вторая пара микросхем ПЗУ выбирается при обращении к памяти программ по адресам с высоким уровнем сигнала A12. Поскольку разряды адреса A13–A15 в дешифрации не участвуют, то блок памяти программ с адресами 0–01FFFh повторяется и по адресам 02000h–03FFFh, 04000–05FFFh, ..., 0E000–0FFFFh.

Первая микросхема ОЗУ выбирается при обращении к памяти данных (низкий уровень сигнала -DS) по адресам с низкими уровнями сигналов A12 и A11, вторая микросхема ОЗУ выбирается при обращении к памяти данных по адресам с низким уровнем сигнала A12 и высоким – A11 и третья микросхема ОЗУ выбирается при обращении к памяти данных по адресам с высоким уровнем сигнала A12 и низким – A11. Таким образом, блок ОЗУ данных занимает адреса 0–017FFh, адреса 01800h–01FFFh не используются (высокие уровни сигналов A12 и A11). Поскольку разряды адреса A13–A15 в дешифрации не участвуют, то блок памяти данных с адресами 0–01FFFh повторяется аналогично блоку памяти программ.

Для чтения данных из памяти (подключения выходов микросхем памяти к шине данных процессора) используется проинвертированный сигнал R/-W, который дополнительно может быть простробирован сигналом -MEMSTRB. Для записи данных в оперативную память используется сигнал R/-W, простробированный сигналом -MEMSTRB. В отличие от сигнала чтения памяти сигнал записи информации в память должен обязательно стробироваться сигналом -MEMSTRB [2].

4.3. Проектирование интерфейса микропроцессорного блока с главным процессором

Многие устройства ЦОС подключаются к персональному компьютеру или являются его составной частью. Поэтому важным вопросом является связь проектируемого устройства с ПЭВМ и их взаимодействие. Для этого используются два основных способа: подключение в качестве внешнего устройства через стандартные порты ПЭВМ (COM – интерфейс RS-232, LPT – интерфейс Centronix, USB) или в виде встраиваемой платы через системную шину ПЭВМ

(ISA, PCI). В зависимости от способа подключения изменяются аппаратная и программная части интерфейса с главным (HOST) процессором.

4.3.1. Подключение устройства ЦОС в качестве внешнего устройства

Подключение устройства к последовательному интерфейсу типа RS-232 может быть выполнено, как в лабораторном макете на базе процессора TMS320C50 [3]. При этом требуются только две микросхемы преобразователей уровня сигналов (75188 и 75189) для сопряжения TTL-уровней микросхемы процессора с уровнями интерфейса RS-232 (рис. 4.3).

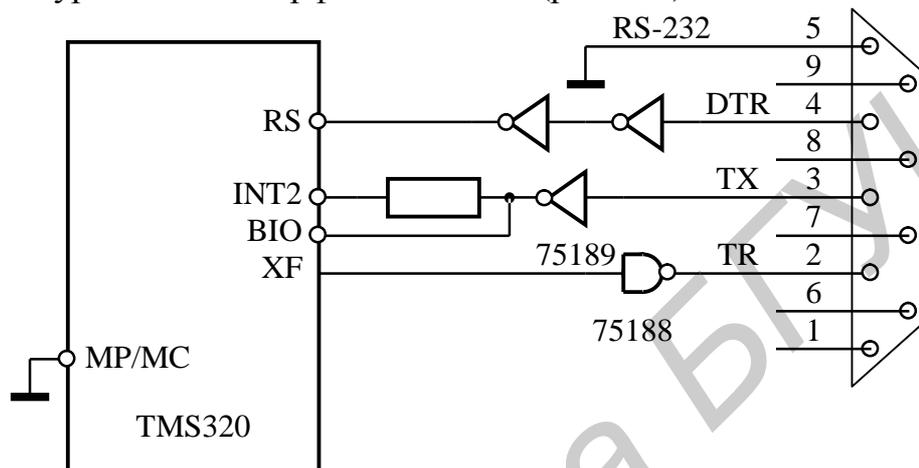


Рис. 4.3. Подключение ЦПС к интерфейсу RS-232

Более сложным является подключение к параллельному интерфейсу. В этом случае требуется использование специальной микросхемы PPC34C60, как в отладочном модуле на базе процессора TMS320VC5402 [4].

Аналогичным образом с использованием специальных микросхем можно подключить модуль ЦОС и к интерфейсу USB.

4.3.2. Подключение устройства ЦОС через системную шину

Рассмотрим подключение устройства ЦОС к системной шине ПЭВМ на примере шины ISA (рис. 4.4).

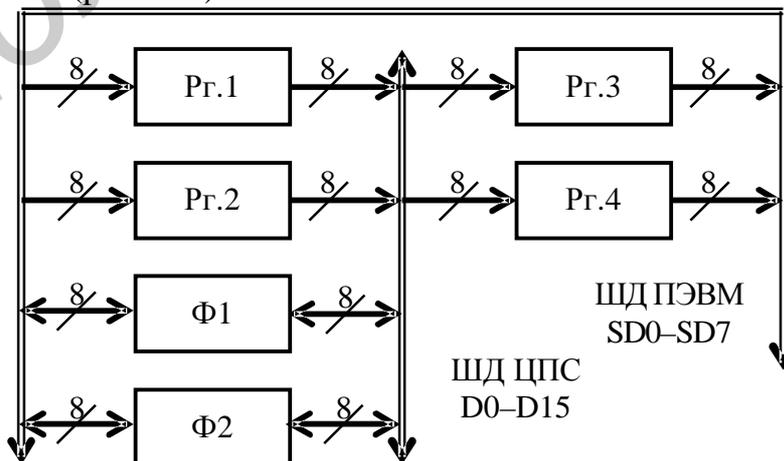


Рис. 4.4. Подключение ЦПС к системной шине ISA

Для подключения модуля ЦОС к системной шине ПЭВМ в байтовом режиме обмена с обеспечением возможности обмена информацией через порт ввода-вывода и непосредственный доступ к памяти модуля в общем случае необходимо четыре 8-разрядных регистра и два 8-разрядных формирователя.

При этом регистры Rг.1 и Rг.2 служат для приема информации из ПЭВМ и выдачи ее в ЦПС, а регистры Rг.3 и Rг.4 соответственно для приема информации от ЦПС и выдачи ее в ПЭВМ. Каждый регистр в паре соединяется со своей половиной шины данных ЦПС и одноименными разрядами шины данных ПЭВМ. Для обмена информацией между ПЭВМ и внешней памятью модуля предназначены двунаправленные формирователи Ф1 и Ф2. Регистры Rг.1–Rг.4 представляют собой пару 16-разрядных буферных регистров (один 16-разрядный двунаправленный регистр данных), которые позволяют передавать данные между модулем и HOST-компьютером под управлением программы ввода-вывода.

С каждым буферным регистром связан флаг (триггер) "буфер заполнен/буфер пуст". Каждый флаг устанавливается в 1 или сбрасывается в 0, когда к соответствующему буферному регистру выполняется обращение по чтению или записи. Состояние этих флагов отражается в регистре состояния модуля, который может быть проанализировано ЦПС и HOST-компьютером. Кроме того, эти флаги могут быть подключены к линиям запроса прерывания HOST-компьютера и процессора ЦПС.

Используя специализированные микросхемы 8-разрядных двунаправленных приемопередатчиков с регистрами и тремя состояниями на выходе, приведенную выше схему можно выполнить на двух микросхемах (рис. 4.5).

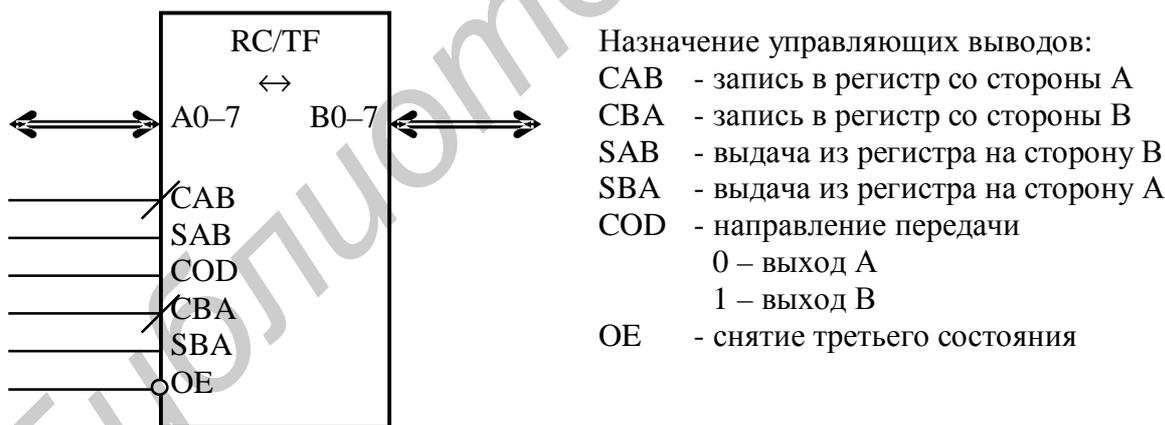


Рис. 4.5. Приемопередатчик с регистрами

Поскольку прямой доступ к памяти данных модуля используется довольно редко, рассмотрим схему связи модуля с шиной данных компьютера в режиме регистров данных. Данная схема включает два приемопередатчика с регистрами, два триггера готовности и два 2-разрядных однонаправленных формирователя (рис. 4.6). Для снижения нагрузки на системную шину ПЭВМ подсоединение всех схем к системной шине может быть выполнено через один 8-разрядный двунаправленный формирователь.

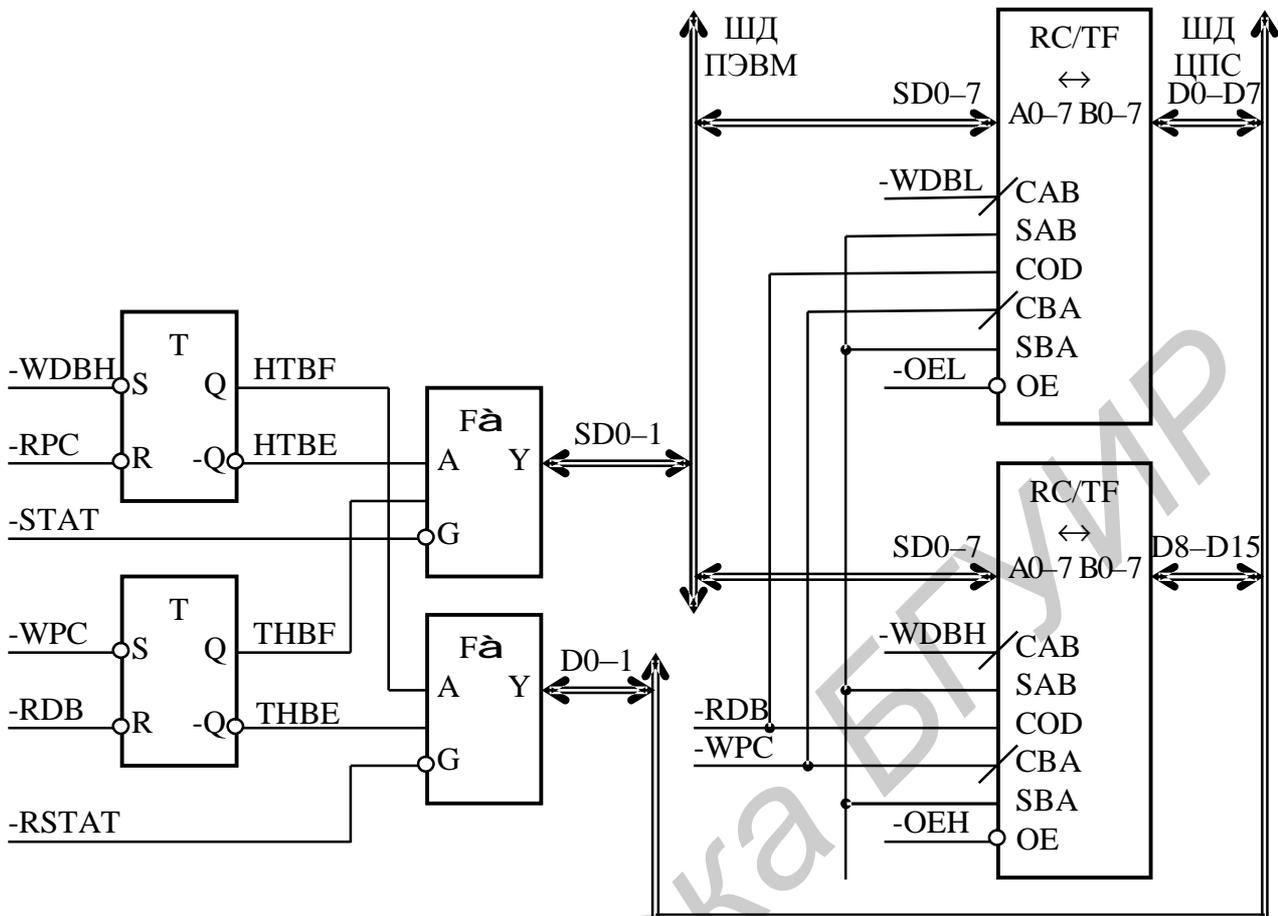


Рис. 4.6. Схема подключения регистров к системной шине ПЭВМ

Принцип работы схемы заключается в следующем. При записи данных со стороны ПЭВМ (сигнал -WDBH) устанавливается первый триггер. Его состояние читается в ПЭВМ и ЦПС как буферный регистр не пустой (-HTBE) и буферный регистр полон (HTBF) соответственно. При чтении данных со стороны ЦПС (сигнал -RPC) первый триггер сбрасывается и его состояние читается в ПЭВМ и ЦПС как буферный регистр пустой (HTBE) и буферный регистр не полон (-HTBF) соответственно. Аналогично работает и второй триггер при передаче данных в противоположном направлении.

Интерфейс шины HOST-компьютера обеспечивает подключение модуля к персональным компьютерам архитектурно и конструктивно совместимым с ПЭВМ IBM PC (386, 486, ...), имеющим стандартную AT-шину (рис. 4.7). Он осуществляет дешифрацию адресов пространства ввода-вывода и формирование соответствующих управляющих сигналов. Схема дешифрации портов ввода-вывода включает блок сравнения адресов и дешифратор портов.

Схема сравнения сравнивает адрес на шине ПЭВМ с адресом модуля, заданным переключателями. При совпадении адреса дешифратор по двум младшим разрядам адреса формирует сигналы обращения к регистрам.

Аналогичным образом строится и схема обращения к внешней памяти ЦПС через окно в адресном пространстве ПЭВМ.

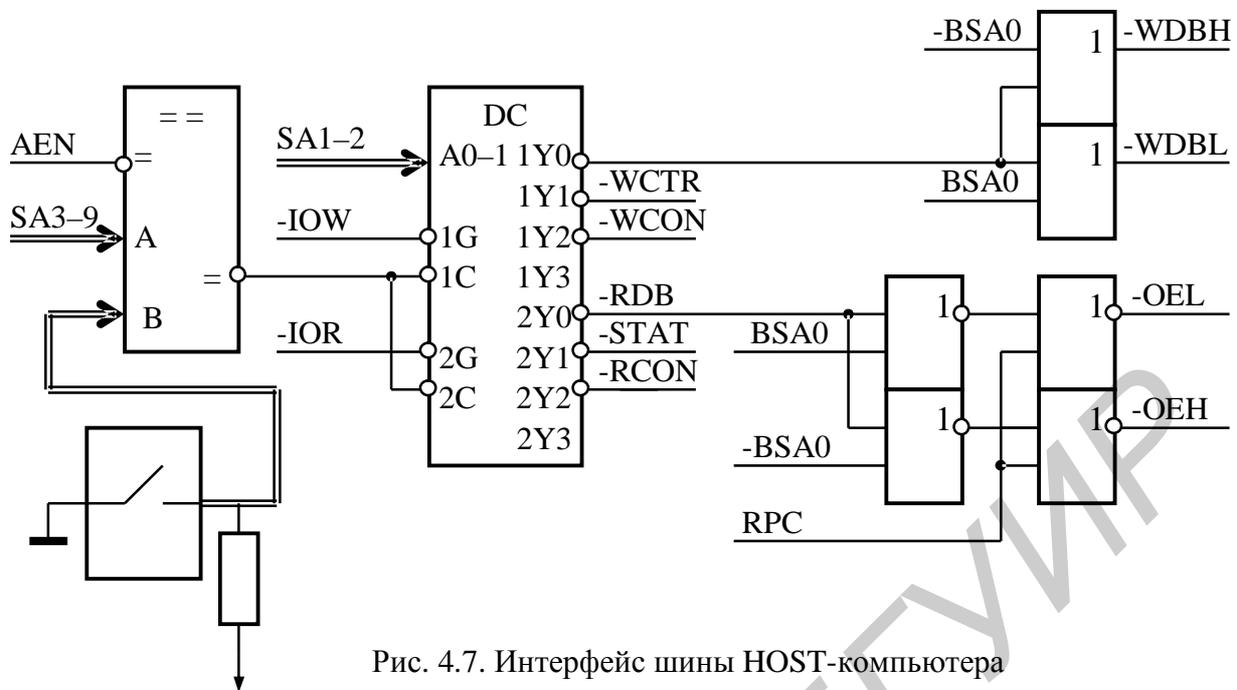


Рис. 4.7. Интерфейс шины HOST-компьютера

Подключение к системной шине PCI выполняется с использованием специальных интерфейсных микросхем, поскольку требования к параметрам сигналов шины более жесткие, чем у шины ISA.

4.4. Проектирование интерфейса микропроцессорного блока с внешними устройствами

В курсовом проекте ЦПС должен взаимодействовать с множеством внешних устройств. В основном это различные АЦП и ЦАП, а также датчики, клавиатуры, световые, цифровые и графические индикаторы, устройства звуковой сигнализации и т.д. [5].

4.4.1. Оценка погрешностей аналого-цифрового преобразователя

Погрешность аналого-цифрового преобразователя (АЦП) – это разность между теоретическим и фактическим значениями аналоговой величины, соответствующей заданному цифровому сигналу. Погрешность АЦП, связанная с дискретизацией непрерывного входного сигнала по уровню и времени, условно подразделяется на статическую и динамическую.

Оценка статической погрешности АЦП

Статическая погрешность описывает точностные свойства АЦП при постоянном (квазипостоянном) входном сигнале. Статическая погрешность преобразования входной величины при линейном (равномерном) квантовании идеальным АЦП определяется разрядностью последнего. Полная статическая погрешность преобразования реального АЦП может быть сведена к погрешности одного отсчета, которая включает погрешность квантования по уровню и инструментальную погрешность. Теоретически показано, что при квантовании по

уровню идеальный линейный АЦП имеет методическую погрешность, максимальное значение которой равно [6]:

$$\Delta_{\max} = \pm 0,5q,$$

где q – шаг квантования.

Значение погрешности квантования по уровню в конечной точке характеристики преобразования линейного n -разрядного униполярного нормализованного к единичному значению АЦП определяется как

$$\Delta_{FS} = 3(0,5q) = 3 \cdot 2^{-(n+1)},$$

где $q = 2^{-n}$.

Возникновение методической погрешности (шума квантования) связано с тем, что при квантовании по уровню текущее значение преобразуемого входного сигнала в пределах шага квантования заменяется ближайшим дискретным значением соответствующего уровня квантования. Следовательно, действительное значение входного сигнала, определенное на выборочном пространстве задаваемым шагом квантования, является случайной величиной. Так как закон распределения случайной величины при квантовании по уровню является равномерным, дисперсия погрешности квантования по уровню равна

$$s_{кв}^2 = q^2 / 12 = (0,5q)^2 / 3.$$

Другой составляющей статической погрешности реального АЦП является инструментальная погрешность, включающая как систематическую, так и случайную погрешности. Систематическая погрешность приводит к изменению характеристики преобразования, выражающейся в следующем:

- 1) общем сдвиге уровней квантования относительно начала координат;
- 2) изменении коэффициента преобразования, характеризующего погрешность преобразования в конечной точке характеристики преобразования;
- 3) нелинейности характеристики преобразования, приводящей к появлению в спектре преобразуемого входного сигнала многочисленных спектральных компонент, хаотично расположенных по всему спектру [7].

Характеристикой нелинейности может служить отношение сигнал/шум реального АЦП, которое рассчитывается по результатам дискретного преобразования Фурье [6].

Общую систематическую погрешность с учетом методической погрешности можно выразить в виде

$$d_c[N] = (N - 0,5)q - x_0,$$

где N – число, соответствующее выходному коду АЦП, x_0 – истинное значение входной величины, соответствующее данному выходному коду АЦП.

Кроме указанных составляющих погрешностей одного отсчета в реальных АЦП всегда имеются шумы и помехи, которые являются источником случайной составляющей инструментальной погрешности. При этом возникновение случайной погрешности связано с:

4) стационарными случайными процессами, вызванными как шумами входной цепи АЦП, так и помехами, проходящими по цепям питания;

5) нестационарными процессами типа «дрейф», «старение» и другие, порождаемые флуктуациями параметров элементов АЦП, являющиеся функцией времени;

б) случайными процессами, возникающими под действием изменений внешней среды, например, изменения температуры либо напряжения питания.

Для определения влияния случайной погрешности на закон распределения полной погрешности квантования по уровню ($\delta_n = \delta_c + \delta_{кв} + \delta_{сл}$) предположим:

а) систематическая погрешность δ_c не превышает значения, равного половине шага квантования, т.е. последняя не влияет на полную погрешность;

б) погрешность квантования $\delta_{кв}$ распределена по равномерному закону и имеет плотность вероятности, равную

$$j(d_{кв}) = 1/q \quad (\text{при } -0,5q \leq d_{кв} \leq +0,5q);$$

в) случайная погрешность $\delta_{сл}$ имеет нормальный закон распределения

$$j(d_{сл}) = \frac{1}{s\sqrt{2p}} \exp\left(-\frac{d_{сл}^2}{2s^2}\right),$$

где σ – среднее квадратическое отклонение случайной погрешности.

Тогда плотность вероятности суммы равномерного и нормального законов распределения двух случайных величин равна

$$j(d_{кв} + d_{сл}) = \frac{1}{q\sqrt{2ps}} \int_{-\infty}^{+\infty} \exp\left[-\frac{(d_{сл} - d_{кв})^2}{2s^2}\right] dd_{кв} = \frac{1}{q} \left\{ \frac{1}{2} \operatorname{erf}\left[\frac{1}{\sqrt{2}} \frac{(d_{сл} - d_{кв})}{s}\right] \right\}, \quad (1)$$

где erf – функция ошибок.

Учитывая, что ошибка квантования равна $\delta_{кв} = \pm 0,5q = \pm 3\sigma$, выражение для плотности вероятности полной погрешности имеет вид

$$j(d_n) = \frac{1}{q} \left\{ \frac{1}{2} \operatorname{erf}\left[\frac{1}{\sqrt{2}} \frac{(d_{сл} - 3s)}{s}\right] \right\} = \frac{1}{q} \Phi(U), \quad U = \frac{(d_{сл} - 3s)}{s}, \quad (2)$$

где $\Phi(U)$ – табулированная функция интеграла вероятностей.

Выражение (2) показывает, что наличие при выполнении преобразования случайной погрешности приводит к фактическому увеличению размера зоны квантования и изменению закона распределения полной погрешности, а также к необходимости дополнительной статистической обработки входного сигнала.

Таким образом, полная статическая погрешность АЦП является суммой методической, систематической и случайной погрешностей. При этом при оценке общей систематической погрешности ошибки, вызванные причинами 1–3 подлежат алгебраическому суммированию, а при оценке общей случайной погрешности ошибки, вызванные причинами 4–6 всегда суммируются геометрически. Погрешность квантования по уровню идеального АЦП проявляется как шум квантования с равномерной спектральной плотностью. Наличие инст-

рументальной погрешности приводит к возникновению дополнительных гармонических составляющих в спектре преобразуемого входного сигнала, т.е. к увеличению шума АЦП, а также к изменению закона распределения преобразуемой входной величины.

Помимо обеспечения необходимой разрядности, выбираемый АЦП должен обеспечивать требуемый динамический диапазон, который определяется как отношение максимального сигнала к среднему квадратическому значению шумов квантования.

Для идеального АЦП, в котором отсутствуют внешние шумы и помехи, отношение сигнал/шум выраженное в децибелах, равно [8]

$$(C / Ш)_{ид} = 10 \lg \frac{P_{вх}}{P_{кв}},$$

где $P_{вх}$ – средняя мощность входного сигнала, $P_{кв} = q^2/12$ – средняя мощность шума квантования.

Отношение сигнал/шум для реального АЦП, на вход которого воздействуют как внутренние, так и внешние источники аддитивного шума, определяется как

$$(C / Ш)_{ра} = 10 \lg \frac{P_{вх}}{P_{кв} + P_{дон}}, \quad (3)$$

где $P_{дон}$ – средняя мощность дополнительных шумов, которая является результатом взаимодействия шумов (помех) как от внешних (внутренних) источников, так и шумов квантования.

Если выразить мощность дополнительного шума как $P_{дон} = k^2 P_{кв}$, где k – представляет отношение среднего квадратического значения шума в аналоговом канале на входе АЦП к среднему квадратическому значению шума квантования, тогда выражение (3) можно представить в виде

$$(C / Ш)_{ра} = 10 \lg \frac{P_{вх}}{P_{кв}(k^2 + 1)} = 10 \lg \frac{P_{вх}}{P_{кв}} - 10 \lg(k^2 + 1). \quad (4)$$

Первый член выражения (4) характеризует отношение сигнал/шум идеального АЦП, а второй член $10 \lg(k^2 + 1)$ представляет ухудшения динамического диапазона реального АЦП. Например, при значении $k = 1$, когда $P_{кв} = P_{дон}$ отношение сигнал/шум АЦП уменьшается на 3 дБ, а при $k = 2$ указанные потери составляют более 6 дБ, что означает потерю младшего разряда. Отсюда следует, что если отношение сигнал/шум реального АЦП начинает определяться $P_{дон}$, то нет смысла в данном случае увеличивать разрядность АЦП. При этом для достижения заданного динамического диапазона АЦП гораздо выгоднее уменьшать дополнительные шумы до уровня соответствующего младшему разряду преобразователя.

Для входного сигнала с нормализованным к единице максимальным значением входного униполярного сигнала выражение (4) может быть представлено в виде

$$(C / Ш)_{pa} = 10 \lg \left[\frac{12(2^n - 1)^2}{k^2 + 1} \right],$$

где n – разрядность АЦП.

Данное выражение позволяет определить требования к разрядности АЦП при заданном отношении сигнал/шум преобразователя или заданном отношении k .

Рассмотрим определение разрядности АЦП, необходимой для обеспечения требуемого диапазона изменения входного сигнала при заданной ошибке оценивания амплитуды минимального синусоидального сигнала по уровню.

При линейном квантовании количество уровней квантования, определяемое n -разрядным АЦП, связано с диапазоном изменения входного сигнала в пределах от X_{\min} до X_{\max} и шагом квантования q соотношением

$$2^n = \frac{X_{\max} - X_{\min}}{q} + 1 = \left\{ \left[\left(\frac{X_{\max}}{X_{\min}} - 1 \right) \frac{X_{\min}}{q} \right] + 1 \right\}. \quad (5)$$

При этом максимальное значение входного сигнала X_{\max} , ограничивается уровнем, равным полной шкале выбранного АЦП, а минимальное значение входного сигнала X_{\min} , ограничивается уровнем шума преобразователя.

Из (5) следует, что требуемая разрядность преобразователя, необходимая для обеспечения заданного диапазона изменения входного сигнала при постоянном шаге квантования определяется следующим образом:

$$n \approx 3,322 \left[\lg \left(\frac{X_{\max}}{X_{\min}} \right) + \lg \left(\frac{X_{\min}}{q} \right) \right]. \quad (6)$$

Учитывая, что динамический диапазон изменения входного сигнала определяется отношением (X_{\max}/X_{\min}), выражение (6) в децибелах имеет вид

$$n \approx 0,166D[\text{дБ}] + 3,322 \lg(X_{\min} / q). \quad (7)$$

Как было показано выше, разрядность выходного кода АЦП определяет, с одной стороны, динамический диапазон измерения входного сигнала, а с другой стороны, уровень шума квантования преобразователя.

Оценим минимальное значение входного сигнала X_{\min} , ограниченного уровнем шума квантования. Предположим, что в качестве входного сигнала используется синусоидальный сигнал

$$a(t) = A_m \sin(\omega t + j_0),$$

где A_m – амплитудное значение входного сигнала, ω – круговая частота, φ_0 – начальная фаза сигнала.

Амплитудное значение входного сигнала, выраженное через дисперсию:

$$A_m = \sqrt{2s_x^2}, \quad (8)$$

где s_x^2 – дисперсия входного сигнала при равномерном квантовании.

Обозначим величину допустимой относительной ошибки ε^2 оценивания односторонних спектральных плотностей шума квантования $G_{кв}$ и входного сигнала G_x как

$$e^2 = \int_0^{\infty} G_{кв}(f)df / \int_0^{\infty} G_x(f)df. \quad (9)$$

В соответствие с равенством Парсеваля дисперсия случайного процесса связана со спектральной плотностью соотношением

$$s^2 = \int_0^{\infty} G(f)df, \quad (10)$$

тогда с учетом (9) и (10) величина допустимой относительной ошибки определяется как

$$e^2 = s_{кв}^2 / s_x^2,$$

где $s_{кв}^2, s_x^2$ – соответственно дисперсии шума квантования и входного сигнала.

Дисперсия входного сигнала, определяемая через дисперсию шума квантования и величину допустимой относительной ошибки, равна

$$s_x^2 = \frac{s_{кв}^2}{e^2} = \frac{q^2}{12} \cdot \frac{1}{e^2}. \quad (11)$$

Минимальное значение X_{\min} , соответствующее амплитудному значению входного сигнала A_m , с учетом (8) и (11) определяется выражением

$$X_{\min} = \sqrt{2s_x^2} = \sqrt{2q^2/12e^2} \approx 0,408 q/e. \quad (12)$$

Подставляя полученное выражение в (7), определим требуемую разрядность выходного кода идеального АЦП:

$$n_{\sim} \approx 0,166D[\text{дБ}] - 3,322 \lg e_1 - 1,293, \quad (13)$$

где n_{\sim} – разрядность выходного кода АЦП, $D[\text{дБ}]$ – динамический диапазон изменения входного сигнала в децибелах, e_1 – ошибка оценивания амплитуды минимального синусоидального сигнала по уровню.

Из (13) следует, что ошибка оценивания минимального синусоидального сигнала по уровню для идеального АЦП, выраженная в процентах, равна

$$e_{\sim}[\%] \approx 10^{\{2 + \langle 0,3[0,166D[\text{дБ}] - (n+1,293)] \rangle\}}$$

Опуская промежуточные преобразования, можно показать, что ошибка оценки минимального уровня постоянного сигнала, выраженная в процентах, равна

$$e_{\sim}[\%] \approx 10^{\{2 + \langle 0,3[0,166D[\text{дБ}] - (n+1,792)] \rangle\}}$$

Оценка динамической погрешности АЦП

Динамические погрешности отражают инерционные свойства АЦП при изменяющемся во времени входном сигнале.

При рассмотрении идеального АЦП предполагалось, что процесс квантования входного сигнала по уровню происходит мгновенно. В действительности на квантование входного сигнала по уровню затрачивается время, зависящее как от метода преобразования, так и от инерционности элементов преобразователя. Общим для любого метода преобразования является сравнение текущего значения входного сигнала с эталонным с помощью схемы сравнения (компаратора). Между частотой входного синусоидального сигнала f_{ex} , относительной погрешностью передачи амплитудного значения этого сигнала δ_k и временем срабатывания компаратора t_k существует следующая зависимость [8]:

$$d_k = 1 - \cos 2\pi f_{ex} t_k,$$

которая при разложении в ряд Тейлора имеет вид

$$d_k \approx (2\pi f_{ex} t_k)^2 / 2.$$

При этом полоса пропускания компаратора по уровню -3 дБ равна

$$\Delta f_{-3\text{дБ}} = 1/(4t_k).$$

Динамическая погрешность компаратора, связанная с преобразованием непрерывной функции по времени в дискретную форму, является результатом ограниченной полосы пропускания компараторов и случайной задержкой отсчета, вызванной как временем срабатывания компаратора, так и скоростью изменения входного сигнала в пределах заданного уровня. Динамическая погрешность компаратора во многом определяет динамическую погрешность АЦП.

Под динамической погрешностью АЦП при изменяющихся во времени параметрах сигнала понимают разность $\Delta_{дин}(t_u)$ между зафиксированным на выходе АЦП значением выходной величины $X_n(t_u) \approx N$ и ее истинным значением $X_x(t_u)$ в рассматриваемый момент времени t_u без учета погрешности одного отсчета:

$$\Delta_{дин}(t_u) = X_n(t_u) - X_x(t_u).$$

Различают два вида динамической погрешности АЦП:

1) динамическая погрешность первого рода, обусловленная инерционностью отдельных узлов АЦП, участвующих в передаче сигнала от входа к выходу АЦП, и определяемая величиной и длительностью переходных процессов, происходящих в этих узлах;

2) динамическая погрешность второго рода, обусловленная изменением входного сигнала за время преобразования и определяемая скоростью изменения или частотным спектром преобразуемого сигнала.

Оценка значения динамической погрешности первого рода для определенного момента времени, связанная с наличием переходных процессов в узлах АЦП, является сложной задачей из-за необходимости учета множественных параметров, характеризующих переходной процесс. Поэтому единственно реальным путем определения динамической погрешности первого рода является моделирование процессов в АЦП [6, 9].

При работе с изменяющимися во времени сигналами возникают специфические погрешности, динамические погрешности второго рода, для оценки которых вводят понятие апертурной неопределенности [8]. Эффект апертурной неопределенности проявляется либо как погрешность мгновенного значения сигнала при заданных моментах измерений, либо как погрешность момента времени, связанная с отнесением данного отсчета к заданному моменту измерения. При равномерной дискретизации следствием апертурной неопределенности является возникновение амплитудных погрешностей, которые называются апертурными и численно равны приращению сигнала в течение апертурного времени [8].

Оценивание динамических погрешностей, возникающих в результате преобразования изменяющегося входного сигнала, зависит от способа представления апертурной погрешности.

Рассмотрим взаимосвязь между заданным значением динамической погрешности второго рода (апертурной погрешности), характеризующей изменение входного сигнала для заданного момента времени, и требуемой разрядностью АЦП.

Если принять, что для n -разрядного АЦП апертурная погрешность не должна превышать шага квантования, то между максимальной частотой входного сигнала f_{\max} , апертурным временем t_a и относительной апертурной погрешностью d_{au} имеет место соотношение

$$d_{au} = 2^{-n} = 2pf_{\max}t_a. \quad (14)$$

Отметим, что апертурное время для рассматриваемого случая является в первом приближении временем преобразования параллельного АЦП, т.к. не учитывает времени, необходимого для дешифрации дискретного значения входного сигнала в цифровой код. Значение разрядности АЦП, обеспечивающее выполнение (14), равно

$$n \approx \lceil -3,321g(2pf_{\max}t_a) \rceil. \quad (15)$$

При решении практических задач, связанных с измерением входной величины, обычно известны максимальное и минимальное значения сигнала, а также верхняя частота в спектре сигнала. Для определения динамической погрешности по этим данным необходимо иметь сведения о законе распределения спектра входного сигнала [6, 9]. Предположим, что преобразованию подвергается стационарный случайный сигнал с равномерным законом распределения мгновенного значения в пределах от X_{\min} до X_{\max} и равномерным спектром $S_x(\omega)$, имеющим верхнюю граничную частоту f_{\max} [9].

Дисперсию скорости изменения входного сигнала σ^2 можно связать с дисперсией входного сигнала s_x^2 выражением

$$s^2 = [(2pf_{\max})^2/3]s_x^2, \quad \text{где} \quad s_x^2 = \int_0^{f_{\max}} S_x(f)df.$$

При известной скорости изменения входного сигнала дисперсия динамической погрешности $s_{\text{дин}}^2$, обусловленная сигналом за время преобразования T_c ,

$$s_{\text{дин}}^2 = s^2 \cdot T_c^2 = \frac{(2pf_{\max}T_c)^2 (X_{\max} - X_{\min})^2}{36}. \quad (16)$$

Учитывая, что динамическая погрешность не должна превышать шага квантования, между частотой сигнала, временем преобразования и разрядностью АЦП, с учетом (5, 16), имеет место соотношение

$$1 = 1,05 f_{\max} T_c 2^n. \quad (17)$$

Требуемая разрядность АЦП, обеспечивающая выполнение соотношения (17), равна

$$n \approx \lceil -\langle 0,1 + 3,321g(f_{\max}T_c) \rangle \rceil. \quad (18)$$

Сравнение (15, 18) показывает, что полученное значение разрядности АЦП при равномерном законе распределения входной величины обеспечивает заданную динамическую погрешность, одинаковую как для случая применения апертурного времени, так и времени преобразования.

При оценивании погрешности преобразования, связанной с динамической погрешностью датирования отсчета, необходимо учитывать, что данная погрешность является результатом совместного влияния ограниченной полосы пропускания компараторов и случайной задержкой отсчета, связанной как со скоростью изменения входного сигнала, так и с наличием внутренних шумов или внешних помех. Например, для входного сигнала с равномерным спектром,

имеющим верхнюю граничную частоту f_{\max} , с граничными значениями погрешности датирования отсчета $\tau_{d \max}$, $\tau_{d \min}$ дисперсия динамической погрешности [6] определяется как

$$S_{\Delta t_d}^2 = \frac{S_x^2}{3} 2pf_{\max} \Delta t_d, \quad (19)$$

где $\Delta \tau_d = \tau_{d \max} - \tau_{d \min}$ – разброс погрешностей датирования отсчетов, определяемый разностью максимального и минимального значения времени срабатывания компараторов, S_x^2 – дисперсия входного сигнала.

Учитывая определенную сложность оценивания погрешности датирования отсчетов при решении практических задач, связанных с отсутствием в приводимых параметрах АЦП времен срабатывания компараторов, наиболее часто определение динамической погрешности второго рода выполняется исходя либо из апертурного времени, либо времени преобразования являющихся заданными характеристиками АЦП.

Апертурная погрешность, связанная с фазовой модуляцией сигнала дискретизации АЦП (фазовый джиттер), из-за наличия внешних помех приводит к паразитной частотно-фазовой модуляции преобразованного сигнала и появлению паразитных спектральных компонент. Следовательно, фазовый джиттер уменьшает динамический диапазон АЦП за счет увеличения шума. Примерно отношение сигнал/шум SNR_{jitter} в децибелах, вызванное флуктуацией (дрожанием) фазы, можно оценить [7]:

$$SNR_{jitter} \approx 20 \lg(1/(2pf_{\max} t_{aj})),$$

где f_{\max} – максимальная частота входного сигнала, t_{aj} – значение флуктуации фазы.

Из вышеизложенного следует, что динамическая погрешность второго рода при заданном апертурном времени или времени преобразования является функцией входного сигнала и не зависит от частоты дискретизации. Более точную оценку динамической погрешности второго рода можно дать с помощью величины среднего квадрата, которая в общем случае является функцией параметров АЦП и характеристик преобразуемого сигнала [6, 9].

Таким образом, для определения полной погрешности необходимо знать статистические характеристики преобразуемого сигнала и прежде всего – закон распределения величины входного сигнала по шкале АЦП.

4.4.2. Аналого-цифровые преобразователи

Аналого-цифровой преобразователь (АЦП) формирует на выходе двоичный код, прямо пропорциональный входному напряжению. Это одно из основных устройств, позволяющее передавать аналоговые сигналы в ЦПС. Существует три основных типа АЦП: параллельные, последовательного приближения

и с двойным интегрированием. Параллельные АЦП имеют наибольшую скорость преобразования; у АЦП с двойным интегрированием максимальная точность преобразования, но они менее быстрые. В качестве компромисса между скоростью и точностью выступают АЦП последовательного приближения. Существуют два типа интерфейса ввода/вывода между АЦП и внешними схемами: параллельный и последовательный. При параллельном интерфейсе выходные данные снимаются через параллельную шину, имеющую несколько линий данных, при последовательном – данные считываются по одной линии.

Фирма Analog Devices, один из лидеров на рынке высокоскоростных преобразователей, выпустила новую микросхему AD7680, которая представляет собой 16-разрядный АЦП последовательного приближения с 15-разрядным кодом без пропусков в корпусе SOT-23 с 6 выводами. Это первый промышленный 16-разрядный АЦП последовательного приближения в малогабаритном корпусе. Обладая высокой пропускной способностью (частота дискретизации 100 кГц) и низким уровнем рассеивания мощности (обычно 3 мВт при 3 В и 15 мВт при 5 В), этот преобразователь является новейшим пополнением семейства 16-разрядных АЦП последовательного приближения PulSAR™, выпускаемых Analog Devices Inc. (ADI). Поскольку AD7680 использует архитектуру последовательного приближения, то в нём отсутствуют задержки распространения.

Выбор 12- и 14-разрядных АЦП с частотой дискретизации 3 МГц ограничен АЦП с конвейерной архитектурой, что всегда означает высокое потребление мощности. Точные же 14-разрядные АЦП с интегральной нелинейностью 1 МЗР до сегодняшнего дня просто не существовали. AD748x – это 12- и 14-разрядные АЦП, обладающие частотой дискретизации до 3 МГц и потребляющие при этом менее 90 мВт.

Это высокоскоростные АЦП последовательного приближения с малым потреблением мощности. AD7485 – это 14-разрядный АЦП, обладающий последовательным интерфейсом с частотой дискретизации до 1 МГц. 14-разрядный AD7484 и 12-разрядный AD7482 используют две модели параллельного интерфейса и частоту дискретизации 3 МГц. Эти АЦП содержат следящий запоминающий усилитель, который работает на частотах до 50 МГц. AD7485, AD7484 и AD7482 обладают отличными характеристиками по постоянному и переменному току.

Благодаря передовой технологии проектирования в этих микросхемах достигнуто очень малое значение рассеиваемой мощности (приблизительно 80...90 мВт) при высокой пропускной способности. Существует два экономичных режима:

а) спящий режим, при котором потребляется ~5 мВт и цепь опорного напряжения поддерживается в активном состоянии для быстрого перехода в режим нормального потребления мощности;

б) ждущий режим, при котором потребление мощности снижается до 5 мкВт.

В АЦП имеется внутреннее опорное напряжение 2,5 В, но можно использовать и внешний источник опорного напряжения 2,5 В. Также с помощью 15-го бита допускается 8%-ный выход за пределы диапазона, таким образом, если диапазон аналогового входного сигнала отклоняется от номинала (что составляет до -200 мВ), то пользователь всё равно может точно оцифровывать сигнал.

Показатели защиты от помех и искажений могут быть улучшены с помощью применения дифференциальных сигналов. Помехи накапливаются в сигнале, когда он передаётся, например, по длинным линиям или печатной плате. Если сигналы дифференциальные, то помехи являются синфазными и могут быть отфильтрованы в АЦП.

AD7450 – это 12-разрядный АЦП последовательного приближения, использующий полностью дифференциальный аналоговый вход. Он может работать как в дифференциальном, так и в однопроводном режиме, в зависимости от питающего напряжения. AD7450 приспособлен для работы с напряжением питания 3 и 5 В и благодаря встроенному широкополосному малому шумящему следящему запоминающему усилителю может работать с входными сигналами, превышающими по частоте 1 МГц. Переменный диапазон изменения напряжения позволяет использовать эту микросхему в самых различных целях. С помощью дифференциальной структуры микросхемы и переменного опорного напряжения пользователь может подобрать различные диапазоны входных сигналов и отклонения. Потребление мощности можно уменьшить снижением пропускной способности или частоты дискретизации или используя экономичный режим работы микросхемы.

По мере того как всё больше требований при промышленном контроле процессов, в медицинской технике и аналитических приборах предъявляется к разрешающей способности, появляются требования и к стоимости используемых АЦП. Семейство микросхем PulSAR™ предоставляет пользователям выбор АЦП с разрешающей способностью в 16 разрядов и частотой дискретизации 1 МГц за приемлемую цену. Но в некоторых случаях низкая цена может оказаться важнее оптимальных характеристик.

Для таких клиентов были разработаны две группы 16-разрядных аналого-цифровых преобразователей: семейство AD7651/AD7652/AD7653 и AD7661/AD7666/AD7667. Все шесть моделей имеют встроенное опорное напряжение 2,5 В. Разница между этими двумя группами заключается в характеристиках интегральной нелинейности. AD765x обладает интегральной нелинейностью ± 6 МЗР, а AD766x — $\pm 2,5$ МЗР. Другими особенностями являются последовательный и параллельный интерфейсы, диапазон входных напряжений от 0 до 2,5 В, 15-разрядный код без пропусков, 48-контактный корпус типа LQFP и меньший 48-контактный корпус, адаптированный к размеру чипов. Все модели совместимы с ранее выпускавшимися АЦП семейства PulSAR, что позволяет клиентам модернизировать уже существующие разработки без их перепроектирования.

AD7708 и AD7709 – это 16-разрядные АЦП, а AD7718 и AD7719 – 24-разрядные. AD7708 и AD7718 могут быть сконфигурированы так, чтобы иметь 10 псевдодифференциальных аналоговых входов, что полезно в системах сбора данных. AD7709 и AD7719 имеют встроенные источники тока, которые устраняют потребность во внешних источниках тока в устройствах измерения температуры. В AD7719 также имеется вспомогательный АЦП, который полезен при реализации вспомогательных функций, таких как температурная компенсация. Использование буфера входных сигналов с размахом, равным напряжению питания, во всех микросхемах, обеспечивает согласование с источниками сигналов в широком диапазоне их выходных сопротивлений. Также при функционировании с опорным напряжением 2,5 В входной усилитель с программируемым усилением позволяет работать с амплитудами сигналов от 20 мВ до 2,56 В. Все микросхемы имеют цифровые входные и выходные контакты.

На рис. 4.8 и 4.9 приведены схемы подключения АЦП AD7851 по последовательному интерфейсу к ЦПС и схемы подключения аналоговых входов.

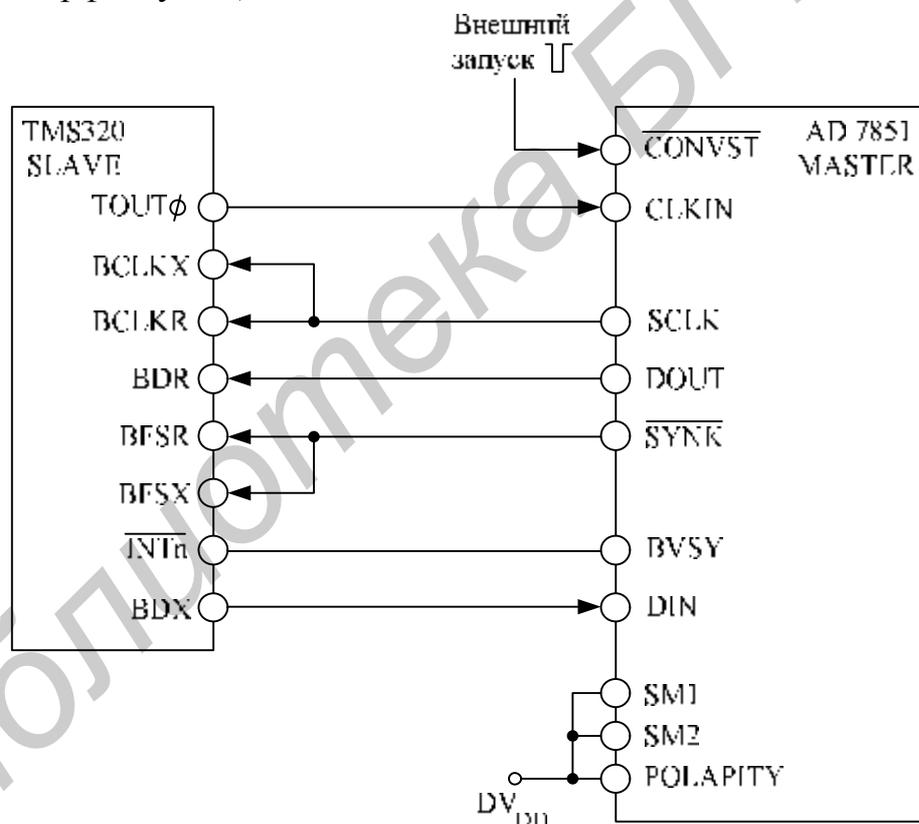


Рис. 4.8. Схема подключения АЦП AD7851 к ЦПС по последовательному интерфейсу

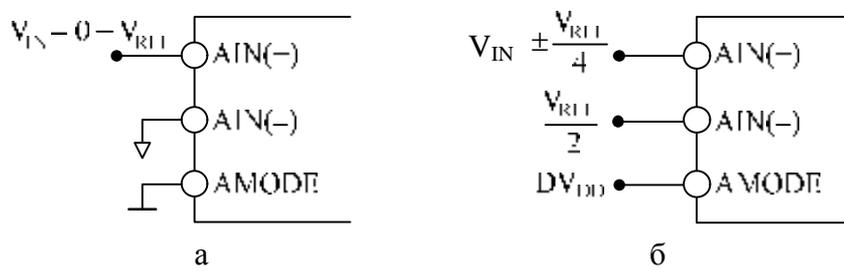


Рис. 4.9. Способы подключения аналоговых входов АЦП AD7851:
 а – униполярное входное напряжение; б – биполярное входное напряжение

На рис. 4.10 приведены схемы подключения аналоговых входов АЦП AD7880.

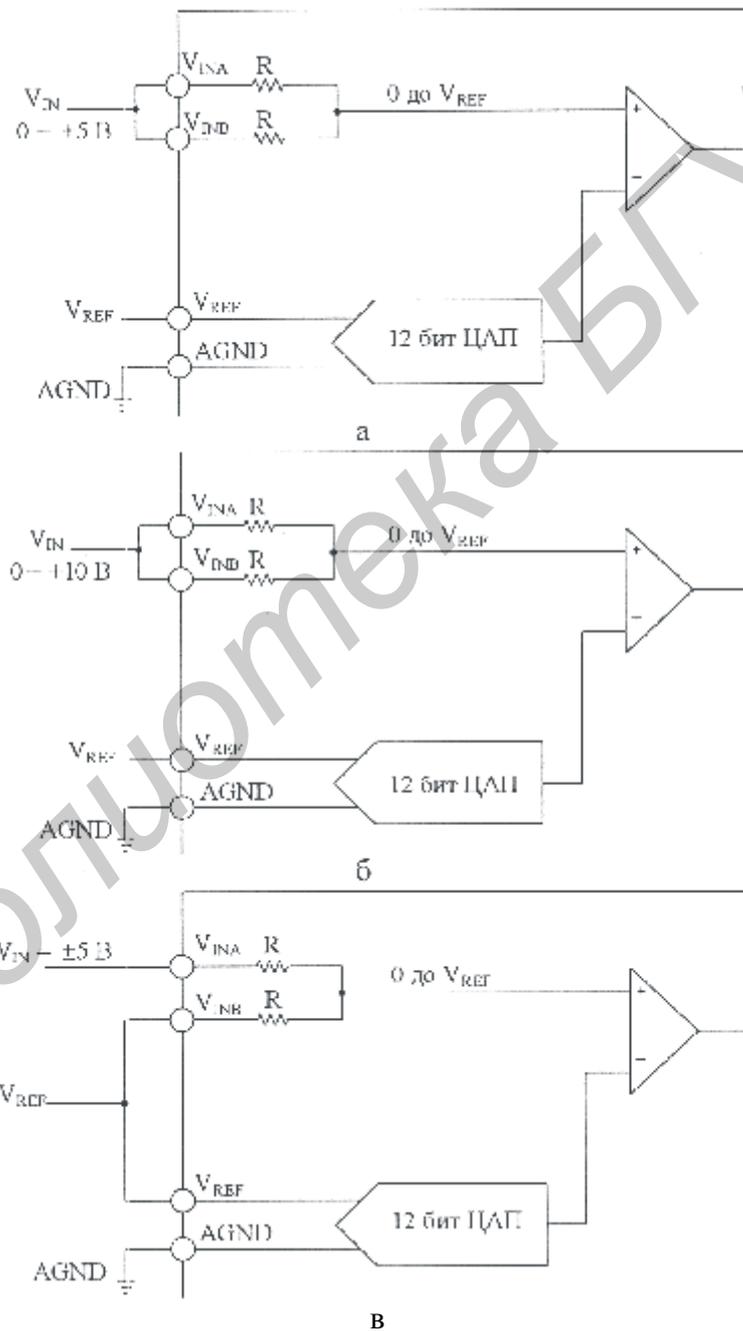


Рис. 4.10. Способы подключения аналоговых входов АЦП AD7880 ($V_{REF} = \pm 5V$):
 а, б – униполярное входное напряжение; в – биполярное входное напряжение

На рис. 4.11 приведена схема подключения АЦП AD7880 по параллельно-му интерфейсу к ЦПС.

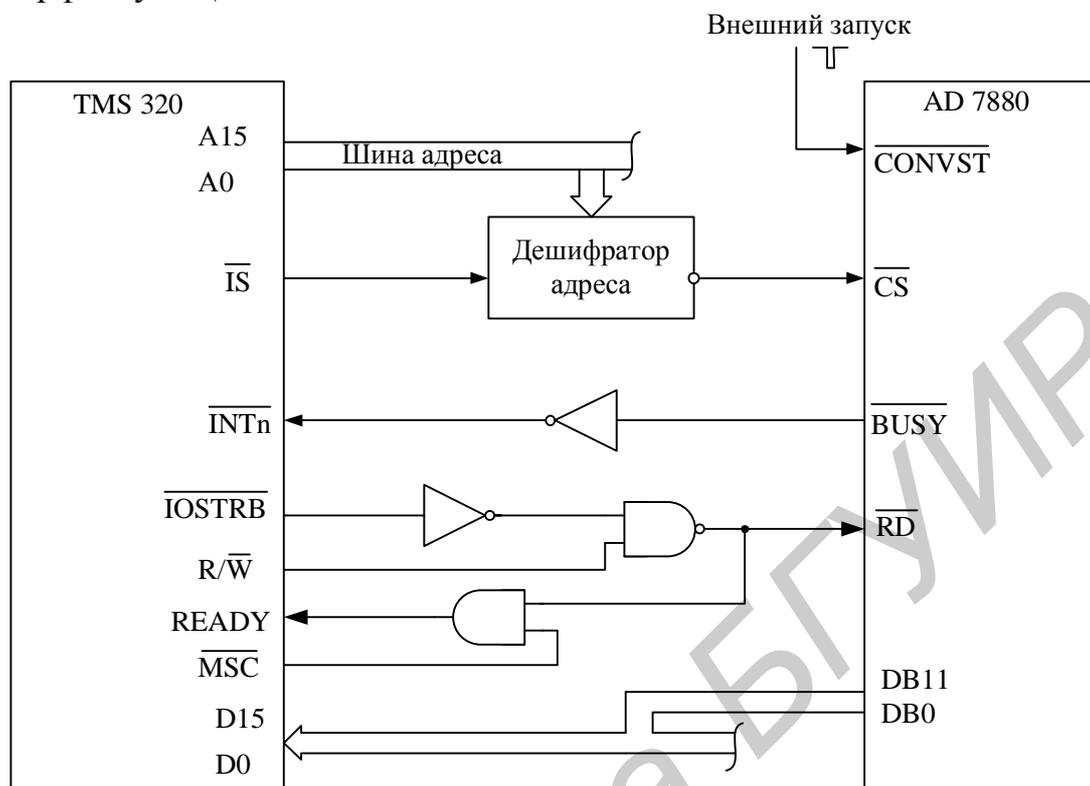


Рис. 4.11. Схема подключения АЦП AD7880 к ЦПС по параллельному интерфейсу

4.4.3. Цифроаналоговые преобразователи

Цифроаналоговый преобразователь (ЦАП) преобразует входной двоичный код в пропорциональное ему аналоговое напряжение или ток. ЦАП широко используется при синтезе речи, музыки и во многих других случаях, когда необходимо обеспечить компьютерное управление внешними аналоговыми устройствами в реальном масштабе времени.

Микросхема ZN428 (GEC-Plessey) представляет собой восьмиразрядный цифроаналоговый преобразователь с буфером-защелкой для входных данных. Он состоит из матрицы R-2R и быстродействующих коммутаторов, что обеспечивает время преобразования, равное 800 нс. Формируемое микросхемой опорное напряжение составляет 2,5 В, напряжение источника питания +5 В, ток потребления в режиме покоя 20 мА.

Микросхема DAC0854BIN (National Semiconductor) – это четырехканальный восьмиразрядный цифроаналоговый преобразователь с последовательным интерфейсом ввода/вывода. Напряжение источника питания микросхемы равно +5 В, ток потребления – 14 мА. DAC0854 содержит четыре ЦАП, для каждого из которых имеется вход опорного напряжения V_{REF} и выход аналогового напряжения V_{OUT} . Встроенный источник опорного напряжения формирует напряжение 2,65 В.

На рис. 4.12, 4.13 и 4.14 приведены схемы подключения ЦАП AD7538 по параллельному интерфейсу к ЦПС и схемы подключения аналоговых выходов.

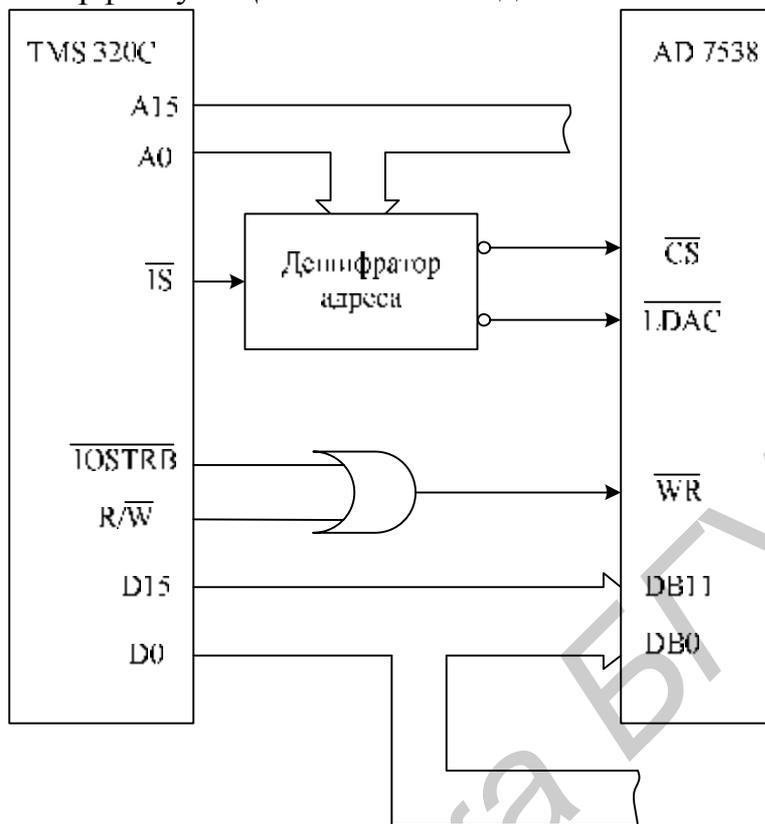


Рис. 4.12. Схема подключения ЦАП AD7538 к ЦПС по параллельному интерфейсу

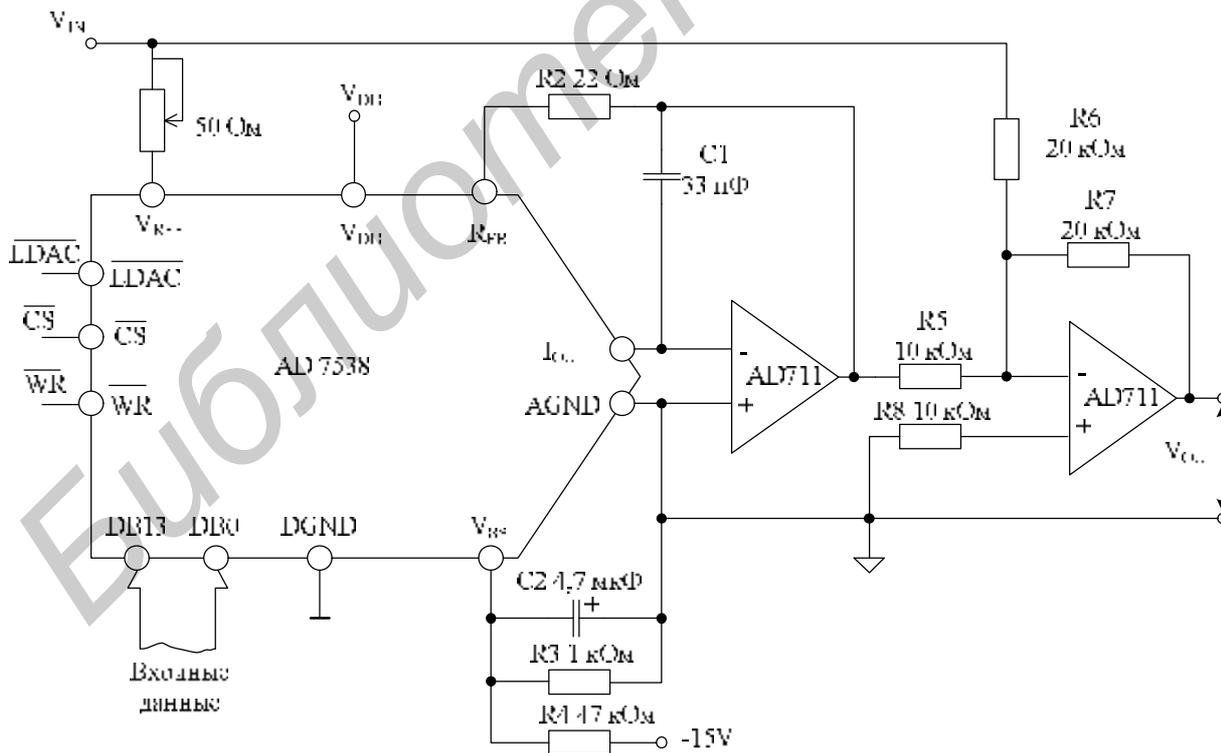


Рис. 4.13. Подключение аналоговых выходов ЦАП AD7538 для получения биполярного выходного напряжения

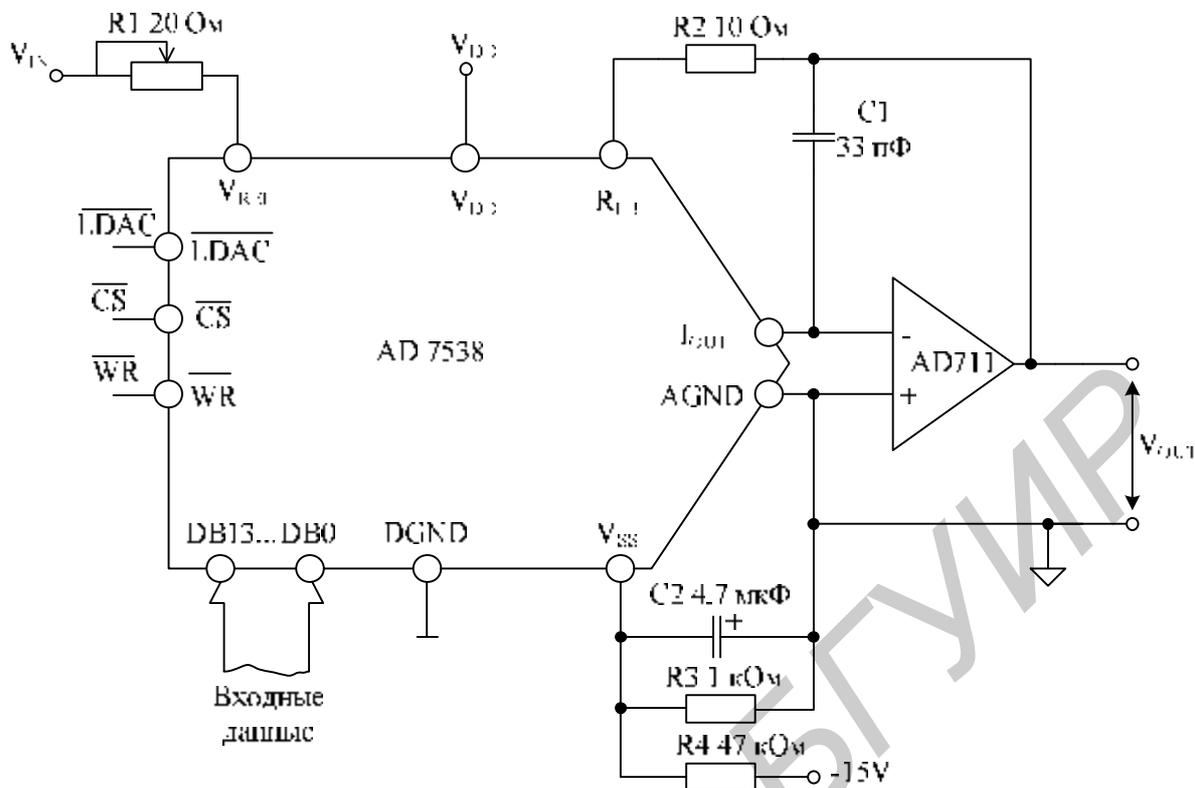


Рис. 4.14. Подключение аналоговых выходов ЦАП AD7538 для получения униполярного выходного напряжения

Подключение ЦАП по последовательному интерфейсу к ЦПС осуществляется аналогично подключению АЦП.

Сопряжение микропроцессоров и микроконтроллеров с другими устройствами приведено в пособии по курсовому проектированию по микропроцессорной технике [10].

5. ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ МИКРОПРОЦЕССОРНОЙ СИСТЕМЫ

Проектирование прикладного программного обеспечения микропроцессорной системы подразделяется на два основных этапа:

- 1) разработка алгоритма работы системы (решения поставленной задачи);
- 2) собственно программирование – разработка прикладных программ на некотором языке программирования с помощью системных средств поддержки проектирования.

Первый этап предполагает выполнение следующих последовательных действий:

- разработка общей блок-схемы алгоритма работы системы;
- разработка детализированных блок-схем алгоритма отдельных процедур, выделенных на основе модульного принципа составления программ;
- детальная проработка интерфейса микроконтроллера и внесение исправлений в общую и детализированные блок-схемы алгоритма.

Так как алгоритм есть точно определенная процедура, предписывающая микроконтроллеру однозначно определенные действия по преобразованию исходных данных в обработанные выходные данные, то разработка блок-схемы алгоритма требует предельной точности и однозначности. В основе разработки блок-схемы алгоритма лежит принцип модульного проектирования – метод декомпозиции, при котором функциональные модули, полученные на этапе предварительного проектирования, последовательно разделяются на меньшие функциональные модули, каждый из которых можно анализировать, разрабатывать и отлаживать отдельно от других. При выполнении прикладной программы в микроконтроллере управление однозначно (без всяких двусмысленностей) передается от одного функционального модуля к другому. Схема связности этих функциональных модулей, каждый из которых реализует некоторую процедуру, образует общую (системную) блок-схему алгоритма прикладной программы. Разделение на модули выполняется последовательно до такого уровня, который обеспечивает получение простого и понятного (прозрачного) алгоритма модуля.

Программные модули должны иметь только одну точку входа и одну точку выхода. Только в этом случае отдельные модули можно разрабатывать и отлаживать независимо, а затем объединять в законченную прикладную программу с минимальными проблемами их взаимосвязей. Источником подавляющего большинства ошибок программирования является использование модулей, имеющих один вход и несколько выходов. При необходимости организации множественных ветвлений в программе условные переходы или включают внутрь модуля, объединяя их с операциями обработки, или выносят в систему межмодульных связей, формируя тем самым блок-схемы алгоритма более высокого уровня, или используют систему программных флагов.

Разработка блок-схемы алгоритма функционального модуля программы имеет ярко выраженный итеративный характер, т.е. требует многократных проб, прежде чем возникает уверенность, что алгоритм реализации процедуры правильный и заверченный. Вне зависимости от функционального назначения процедуры при разработке ее блок-схемы алгоритма необходимо придерживаться следующей последовательности действий:

- 1) определить, что должен делать модуль;
- 2) определить способы получения модулем исходных данных (от датчиков через порты ввода, из таблиц в памяти, через рабочие регистры);
- 3) определить необходимость какой-либо предварительной обработки введенных исходных данных (маскирование, сдвиг, масштабирование, перекодировка);
- 4) определить способ преобразования входных данных в требуемые выходные;
- 5) определить способы выдачи из модуля обработанных данных (передать в память, в вызывавший модуль, в порты вывода);
- 6) определить необходимость какой-либо постобработки выводимых данных (изменение формата, перекодирование, масштабирование, маскирование);

7) проверить работоспособность алгоритма на бумаге путем подстановки в него действительных данных. Убедиться в его сходимости и результативности;

8) рассмотреть предельные случаи и попытаться определить граничные значения информационных объектов, с которыми оперирует алгоритм, за пределами которых он теряет свойства конечности, сходимости или результативности. Особое внимание при этом следует уделить анализу возможных ситуаций переполнения разрядной сетки микроконтроллера, изменения знака результата операции, деления на ноль и т.п.;

9) провести мысленный эксперимент по определению работоспособности алгоритма в реальном масштабе времени, когда асинхронные события, происходящие в объекте управления, могут оказать влияние на работу алгоритма. При этом самому тщательному анализу следует подвергнуть реакцию алгоритма на возможные прерывания с целью определения критических операций, которые необходимо защитить от прерываний. Кроме того, следует проанализировать логику алгоритма с целью определения таких последовательностей операций, при выполнении которых микроконтроллер может не заметить кратковременных событий в объекте управления. При обнаружении таких ситуаций в логику блок-схемы алгоритма следует внести изменения.

Преобразование разработанной блок-схемы алгоритма в исходный текст программы – относительно несложный процесс. Прежде чем приступить к написанию программы, необходимо выполнить распределение памяти и рабочих регистров.

Распределение памяти и рабочих регистров заключается в определении адреса первой команды прикладной программы, действительных начальных адресов стека, таблиц данных, переменных, областей передачи параметров между подпрограммами, подпрограмм обслуживания прерываний и т.п., при этом следует помнить, что в микроконтроллере память программ и память данных физически и/или логически разделены.

ЛИТЕРАТУРА

1. Фридмен М., Ивенс Л. Проектирование систем с микрокомпьютерами: Пер. с англ. – М.: Мир, 1986.
2. TMS320C54x DSP. CPU and Peripherals. Reference Set. Volume 1. – Texas Instruments, 1997.
3. TMS320C5x. DSP Starter Kit. User's Guide. – Texas Instruments, 1994.
4. Компьютерная программа CODE COMPOSER STUDIO. Справочная система. – Texas Instruments, 2001.
5. Ан П. Сопряжение ПК с внешними устройствами / Пер. с англ. – 2-е изд., стер. – М.: ДМК Пресс; СПб.: Питер, 2004.
6. Брагин А.А., Семенюк А.Л. Основы метрологического обеспечения аналого-цифровых преобразователей электрических сигналов. – М.: Издательство стандартов, 1989.
7. Шлеев С.Е. Элементная база и архитектура цифровых радиоприемных устройств //Цифровая обработка сигналов. 1999. №1. С.36 – 47.
8. Бахтиаров Г.Д., Малинин В.В., Школин В.П. Аналого-цифровые преобразователи /Под ред. Г.Д. Бахтиарова. – М.: Сов. радио, 1980.
9. Гитис Э.И., Пискулов Е.А. Аналого-цифровые преобразователи. – М.: Энергоиздат, 1981.
10. Петровский А.А., Качинский М.В., Давыдов А.Б., Ключ В.Б., Таранов Г.В. Микропроцессорная техника: Учеб. пособие по курсовому проектированию. – Мн.: БГУИР, 2005.
11. Пухальский Г.И. Проектирование микропроцессорных систем: Учеб. пособие для вузов. – СПб.: Политехника, 2001.
12. Бойко В. Схемотехника электронных систем. Микропроцессоры и микроконтроллеры. – СПб.: ВНУ, 2004.

БГУИР
(назва ВНУ)

Факультэт компьютерного проектирования

«ЗАЦВЯРДЖАЮ»

Загадчык кафедры _____
(подпіс)

«_____» сентября 20 05 г.

З А Д А Н Н Е
па курсавому праектаванню

Студэнту гр. 110701 Ф.И.О.

1. Тэма праекта **ФОРМИРОВАТЕЛЬ ПОЛИГАРМОНИЧЕСКОГО СИГНАЛА**

2. Тэрміны здачы студентам закончанага праекта 8 декабря 2005 г.

3. Выходныя даныя да праекта _____

3.1. Модуль ЦОС на базе TMS320VC5402

3.2. Память программ 16К (8К x 8), память данных 8К (2К x 16)

3.3. Частота дискретизации 10 кГц, динамический диапазон 48 дБ

3.4. Амплитуды и фазы гармоник задаются

3.5. Число гармоник – 64

4. Змест разлікова-тлумачальнай запіскі (пералік пытанняў, якія неабходна распрацаваць) _____

4.1. Введение. Постановка задачи.

4.2. Анализ задачи. Функциональная спецификация устройства. Список функций, выполняемых устройством, интерфейс пользователя.

4.3. Предварительное проектирование. Разбиение устройства на модули, выбор соотношения между аппаратными и программными средствами, построение структурной схемы устройства, разработка алгоритма работы.

4.4. Проектирование аппаратных средств. Разработка функциональной схемы устройства.

4.5. Проектирование программного обеспечения. Уточнение алгоритма работы устройства, разработка программного обеспечения устройства.

4.6. Заключение.

5. Пералік графічнага матэрыялу (з дакладным пазначэннем абавязковых чарцяжоў і графікаў)

5.1. Схема электрическая структурная устройства.

5.2. Схема электрическая функциональная устройства.

5.3. Блок-схема алгоритма работы устройства.

5.4. Листинги программ.

6. Кансультант па праекту (з пазначэннем раздзелаў праекта) Петровский А.А.

7. Дата выдачи задания 12 сентября 2005 г.

8. Каляндарны графік работы над праектам на ўвесь перыяд праектавання (з пазначэннем тэрмінаў выканання і працаемкасці асобных этапаў)

<u>8.1. Постановка и анализ задачи</u>	<u>10 %</u>	<u>22.09.2005</u>
<u>8.2. Предварительное проектирование</u>	<u>20 %</u>	<u>06.10.2005</u>
<u>8.3. Разработка алгоритма</u>	<u>15 %</u>	<u>20.10.2005</u>
<u>8.4. Проектирование аппаратных средств</u>	<u>15 %</u>	<u>03.11.2005</u>
<u>8.5. Проектирование программного обеспечения</u>	<u>30 %</u>	<u>24.11.2005</u>
<u>8.6. Оформление пояснительной записки</u>	<u>10 %</u>	<u>08.12.2005</u>

КІРАЎНІК _____
(подпіс)

Заданне прыняў да выканання 12 сентября 2005 г. подпись
(дата и подпіс студэнта)

СОДЕРЖАНИЕ

В В Е Д Е Н И Е	3
1. ЦЕЛИ И ЗАДАЧИ КУРСОВОГО ПРОЕКТИРОВАНИЯ.....	3
2. ОСНОВНЫЕ ТРЕБОВАНИЯ К КУРСОВОМУ ПРОЕКТУ	4
2.1. Тематика курсового проектирования	4
2.2. Задание на курсовое проектирование и исходные данные	4
2.3. Содержание и объем курсового проекта.....	5
2.4. Организация курсового проектирования и защита проекта	6
3. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ КУРСОВОГО ПРОЕКТА	7
3.1. Общие указания.....	7
3.2. Требования к построению и содержанию пояснительной записки.....	9
4. ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ АППАРАТНЫХ СРЕДСТВ МИКРОПРОЦЕССОРНОЙ СИСТЕМЫ	19
4.1. Проектирование микропроцессорного блока	20
4.2. Проектирование внешней памяти микропроцессорного блока.....	22
4.3. Проектирование интерфейса микропроцессорного блока с главным процессором	24
4.4. Проектирование интерфейса микропроцессорного блока с внешними устройствами	28
5. ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ МИКРОПРОЦЕССОРНОЙ СИСТЕМЫ	44
ЛИТЕРАТУРА.....	47
ПРИЛОЖЕНИЕ	48

Петровский Александр Александрович,
Клюс Владимир Борисович,
Давыдов Александр Борисович,
Качинский Михаил Вячеславович

ПРОЕКТИРОВАНИЕ ПРОБЛЕМНО-ОРИЕНТИРОВАННЫХ ВЫЧИСЛИТЕЛЬНЫХ СРЕДСТВ

Учебное пособие

по курсовому проектированию
для студентов специальности I-40 02 02
«Электронные вычислительные средства»
дневной формы обучения

Редактор Е.Н. Батурчик

Подписано в печать 18.11.2005.	Формат 60x84 1/16.	Бумага офсетная.
Гарнитура «Таймс».	Печать ризографическая.	Усл. печ. л. 3,14.
Уч.-изд. л. 3,0.	Тираж 150 экз.	Заказ 165.

Издатель и полиграфическое исполнение: Учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
Лицензия на осуществление издательской деятельности №02330/0056964 от 01.04.2004.
Лицензия на осуществление полиграфической деятельности №02330/0131518 от 30.04.2004.
220013, Минск, П. Бровки, 6