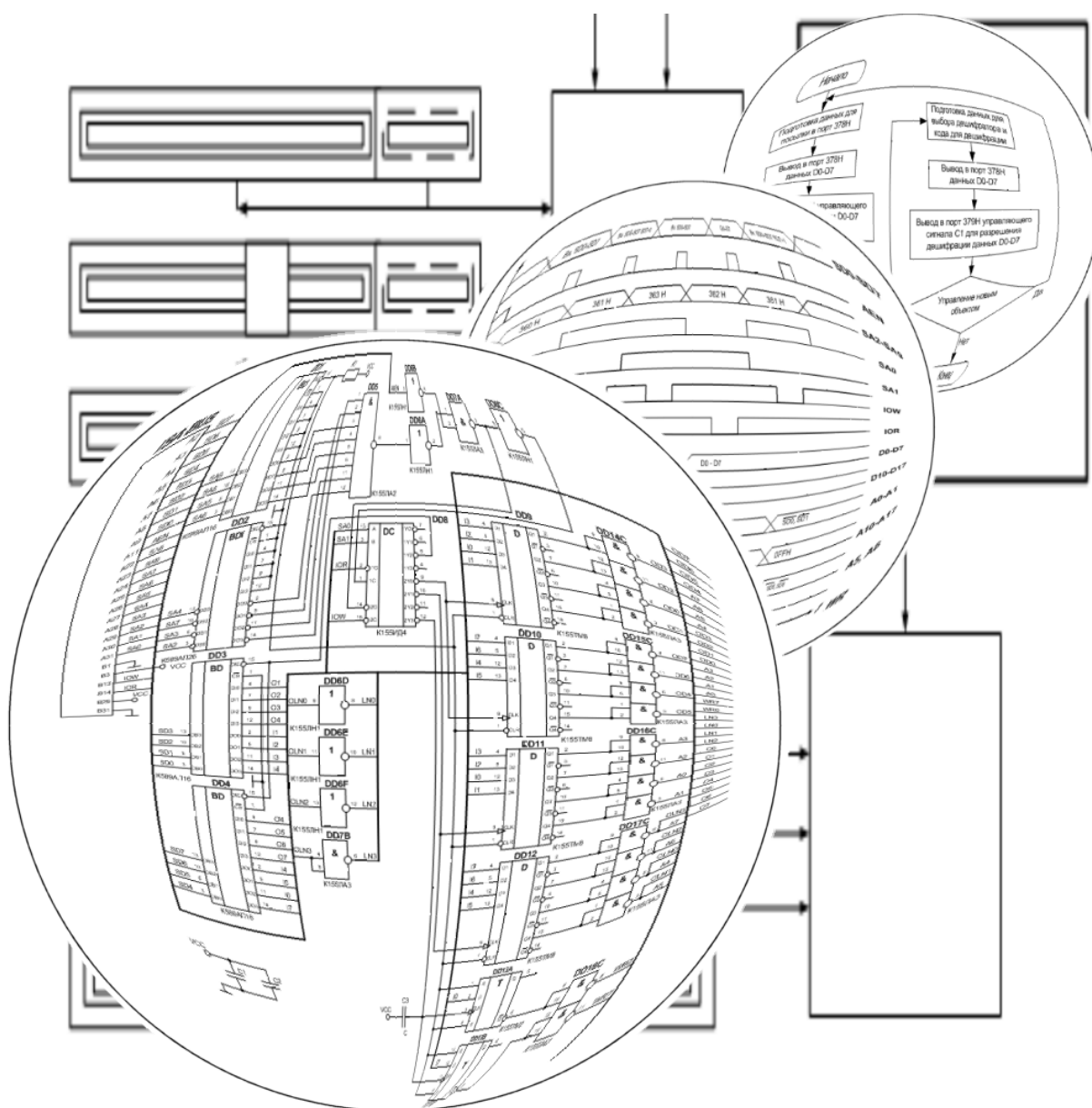


А.В. Меркулов

МИКРОПРОЦЕССОРНАЯ СИСТЕМА УПРАВЛЕНИЯ НА БАЗЕ ИНТЕРФЕЙСОВ ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА

Учебное пособие



Хабаровск
2004

РОССИЙСКОЙ ФЕДЕРАЦИИ
МИНИСТЕРСТВО ПУТЕЙ СООБЩЕНИЯ
ГОУ ВПО «ДАЛЬНЕВОСТОЧНЫЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ПУТЕЙ СООБЩЕНИЯ МПС РОССИИ»

А.В. Меркулов

**МИКРОПРОЦЕССОРНАЯ СИСТЕМА УПРАВЛЕНИЯ
НА БАЗЕ ИНТЕРФЕЙСОВ
ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА**

Учебное пособие

Рекомендовано методическим советом ДВГУПС в качестве учебного пособия для студентов четвертого курса дневной и пятого курса заочной форм обучения направления «Системы обеспечения движения поездов» специальности «Автоматика, телемеханика и связь на железнодорожном транспорте»

2004

УДК 004.326(075.8)++
ББК 3 973.26я73
М 523

Рецензенты:

Заведующий кафедрой «Автоматика и системотехника»
Хабаровского государственного технического университета,
доктор технических наук,
профессор *Чье Ен Ун*

Главный инженер службы СЦБ филиала ОАО РЖД
«Дальневосточная железная дорога»
С.Н. Рябов

Меркулов А.В.

М523 Микропроцессорная система управления на базе
интерфейсов персонального компьютера: Учебное посо-
бие. - Хабаровск: Издательство ДВГУПС, 2004. – 78 с.: ил.

Учебное пособие соответствует ГОС ВПО направления подго-
товки дипломированных специалистов 210700 (190400) «Системы
обеспечения движения поездов».

Рассмотрены основные принципы организации интерфейсов
микропроцессорных систем, основное внимание уделено техноло-
гии создания современных информационно-управляющих комплек-
сов с применением различных подходов.

Предназначено для студентов четвертого курса дневной и пятого
курса заочной форм обучения, обучающихся по специальности
210700 «Автоматика, телемеханика и связь на железнодорожном
транспорте», направлению «Системы обеспечения движения поез-
дов». Может быть использовано при дипломном проектировании
микропроцессорных информационно-управляющих систем.

© Издательство ГОУ ВПО «Дальневосточного
государственного университета путей сообщения»
(ДВГУПС), 2004

ВВЕДЕНИЕ

Практическое изучение интерфейсов вычислительных систем позволит студентам получить навыки и практический опыт создания устройств сопряжения с внешними объектами (УСО) с целью управления и контроля их состояния. Для достижения цели работы студент должен изучить общие принципы организации архитектуры ввода-вывода микропроцессорных систем, существующие интерфейсы, способы преобразования информации, защиты электрических схем и портов ввода-вывода.

Трудно найти область человеческой деятельности, где бы не использовались, в той или иной форме, микропроцессоры и разнообразные устройства на их основе: начиная от сложнейших систем автоматического управления вплоть до простейших датчиков. Системы на их основе представляют собой автоматизированные микропроцессорные комплексы управления и контроля. Они разрабатываются и применяются в программных комплексах диагностики, контроля и управления в различных отраслях. Программно-технический комплекс диагностики и контроля позволяет получать исчерпывающую информацию о состоянии устройств, подключенных к микропроцессорной системе и выдавать управляющие сигналы. В последние годы промышленностью налажен выпуск программного обеспечения и специальных сменных плат, позволяющих превращать компьютер в высококачественную измерительную и испытательную систему. Компьютеры, оснащенные подобным образом, могут использоваться в качестве запоминающих цифровых осциллографов, устройств сбора данных, многоцелевых измерительных приборов. Применение компьютеров в качестве контрольно-измерительных приборов более эффективно, чем выпуск в ограниченных количествах специализированных приборов с вычислительными блоками.

В ходе выполнения работы студент приобретет навык в подборе технических средств, элементной базы и программного обеспечения для построения оптимальной информационно-управляющей системы, научится разрабатывать функциональные и принципиальные схемы УСО, соответствующие поставленным целям разработки. Полученные знания позволят применять на практике методы создания автоматизированных измерительных и диагностических комплексов, компьютерных информационно-управляющих систем, технических комплексов научных исследований и экспериментов.

1. АРХИТЕКТУРА СИСТЕМ ВВОДА-ВЫВОДА

Вопросы организации ввода-вывода в вычислительной системе иногда оказываются вне поля зрения пользователей ЭВМ. Это приводит к тому, что при оценке производительности системы часто учитывается только производительность процессора, тогда как системой ввода-вывода пренебрегают. Такое отношение к системам ввода-вывода, как к не очень важным понятиям, исходит из термина «периферия», который применяется к внешним по отношению к процессору устройствам.

Однако компьютер без устройств ввода-вывода – это чаще всего микропроцессорный контроллер, а не полнофункциональный ПК. Основные составные элементы ПК в значительной степени определяют общий уровень функциональности и производительности. Уже сейчас можно наблюдать, что в компьютерах различного ценового класса - от рабочих станций до суперкомпьютеров (суперсерверов) - используется один и тот же тип микропроцессора. Различия в стоимости и производительности определяются практически только организацией систем памяти и ввода-вывода. Для таких систем одной из наиболее правильных оценок производительности является время ответа (время между моментом ввода пользователем задания и получением результата), которое учитывает все накладные расходы, связанные с выполнением задания в системе, включая ввод-вывод. Организация ввода-вывода существенно влияет на качественные показатели, причем в наибольшей степени тогда, когда речь идет о взаимодействии с оперативной памятью, дисковыми накопителями, видеоадаптерами, звуковыми и другими устройствами.

Эти устройства связаны посредством специальных интерфейсов, которые могут быть организованы как одна или несколько специально выделенных общих шин, к которым подключаются все подсистемы. Традиционно шины делятся на внутренние шины памяти и накопителей, магистральную шину ввода-вывода (системный интерфейс) и периферийное оборудование. Системный интерфейс имеет большую протяженность по сравнению с внутренней шиной, может поддерживать большее количество устройств, и, как правило, соответствует одному из шинных стандартов. Необходимость сохранения баланса по мере роста быстродействия микропроцессоров привела к многоуровневой организации шин системного интерфейса. На общей функциональной схеме организации ввода-вывода (рис.1.1.) приведены основные слоты стандартных интерфейсов ISA, EISA, PCI и AGP.

Количество и типы устройств ввода-вывода в вычислительных системах не фиксируется, что позволяет пользователю самому подбирать необходимую конфигурацию. Системный интерфейс может

рассматриваться как шина расширения, обеспечивающая постепенное наращивание устройств. Разработанные стандарты позволяют разработчикам внешних устройств работать независимо. Успех того или иного стандарта, в значительной степени, определяется принятием его такими организациями, как ANSI (Национальный институт по стандартизации США) или IEEE (Институт инженеров по электротехнике и электронике). Кроме того, стандарт шины может быть разработан одним из комитетов по стандартизации (например, Future-Bus).

Периферийные устройства, такие, как принтер, модем, мышь, сканер и другие включаются через устройства, называемые адаптерами. Взаимодействие осуществляется через интерфейс, определяющий тип соединения, уровень и длительность электрических сигналов протокола обмена. Стандартные последовательные и параллельные интерфейсы называют портом ввода-вывода. Последовательный порт обеспечивает передачу информационных бит данных последовательно один за другим, а параллельный - передает несколько бит данных одновременно. Эти особенности накладывают определенные требования на организацию протокола обмена информацией.

Набор внешних устройств может включать устройства ввода, вывода или двунаправленные. При этом каждое устройство может работать по-своему, а обмен информацией обеспечивает драйвер устройства или базовая система ввода-вывода. Направления и скорости обмена для периферийного оборудования находятся в большом диапазоне (табл.1.1.), и это должно быть учтено при обслуживании таких устройств.

Таблица 1.1

Устройства ввода-вывода и скорости обмена данными

Тип устройства	Направление передачи данных	Скорость передачи данных (Кбайт/с)
Клавиатура	Ввод	0.01
Мышь	Ввод	0.02
Голосовой ввод	Ввод	0.02
Сканер	Ввод	200.0
Голосовой вывод	Вывод	0.06
Строчный принтер	Вывод	1.00
Лазерный принтер	Вывод	100.0
Графический дисплей	Вывод	30000.00
(ЦПг буфер кадра)	Вывод	200.0
Оптический диск	ЗУ	500.00
Магнитная лента	ЗУ	2000.00
Магнитный диск	ЗУ	2000.00

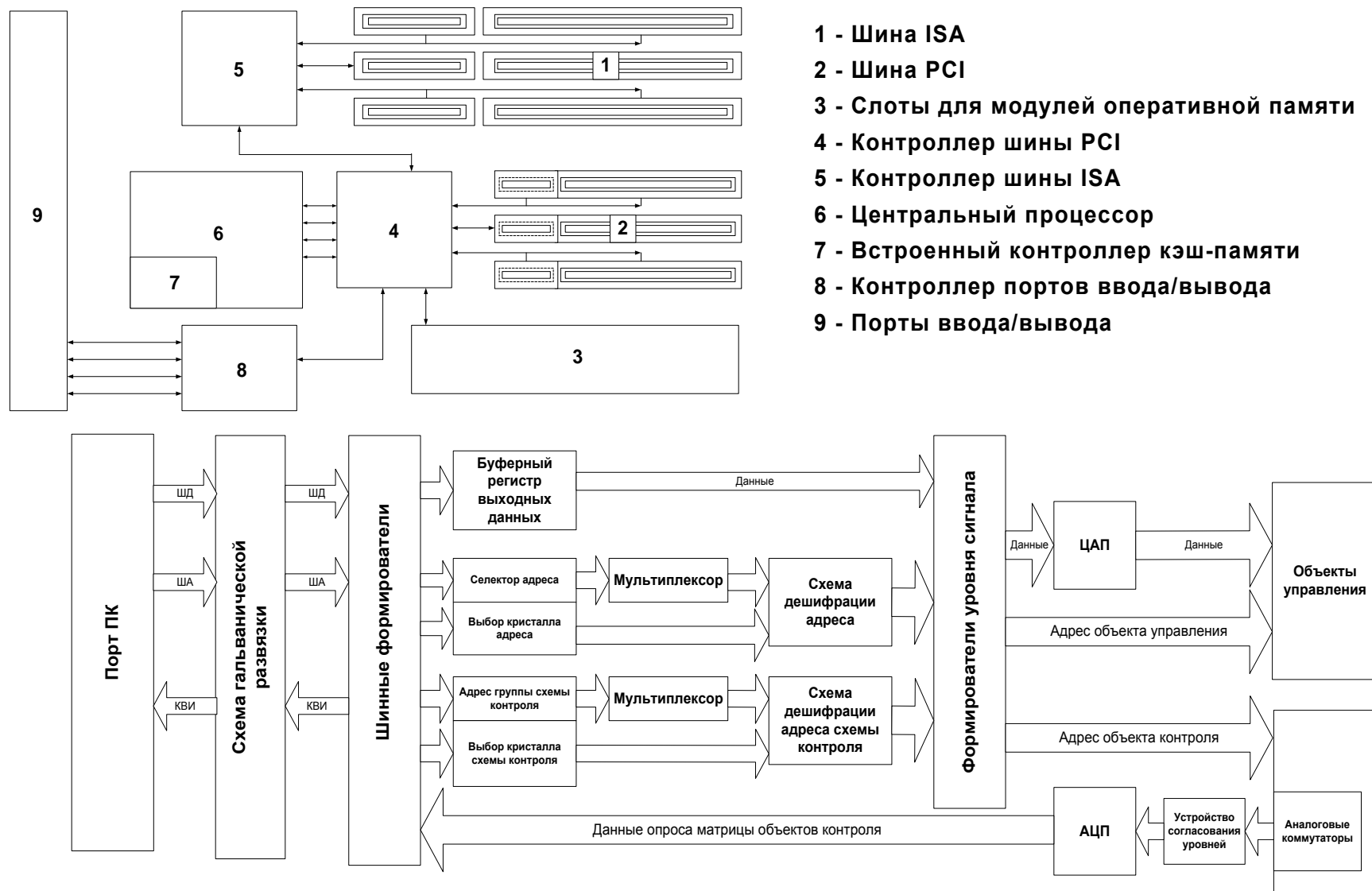


Рис.1.1. Организация системного интерфейса и структурная схема устройства ввода-вывода

С точки зрения пользователя, характеристики производительности устройств ввода-вывода являются субъективными, однако для разработчика это имеет существенное значение. При разработке системы управления объектами необходимо в первую очередь, определиться с характеристиками объектов, их количеством, характером информации, требуемой скорости передачи, длительностью управляющих сигналов. Эти и многие другие параметры определяют тип интерфейса устройства с компьютером.

Практически во всех компьютерах присутствуют стандартные порты ввода-вывода:

- параллельные (LPT1-LPT4), к ним обычно присоединяют принтеры и сканеры, что должно учитываться разработчиком устройства связи с объектом (УСО);
- асинхронные последовательные порты (COM1-COM4), к которым подключаются: мышь, модем и другие устройства;
- игровой порт для подключения джойстика;
- порт USB (универсальная последовательная шина). Он является наиболее перспективным и имеет высокую скорость ввода-вывода. К нему подключаются новые модели принтеров, сканеров, модемов. Главным из его достоинств является возможность подключения целой цепочки устройств. По скорости обмена наилучшие показатели имеет спецификация USB2.0. Скорость его работы превосходит параллельные порты, которые, в свою очередь, выполняют ввод-вывод с большей скоростью, чем последовательные за счет использования большего числа проводов в кабеле.

Помимо стандартных, в компьютере могут применяться системные интерфейсы:

- ISA посредством слота расширения работающая на частоте 8 МГц, что соответствует максимальной скорости передачи 16 Мбайт/с;
- EISA, обеспечивающая адресное пространство 4 Гбайта и 32-разрядную передачу данных. Она тактируется частотой 8 МГц и обеспечивает пропускную способность 33 Мбайта/с;
- PCI – поддерживает 32-разрядный канал передачи данных с тактовой частотой 33 МГц и имеет пропускную способность 120 Мбайт/с.

Кроме того, применяются шины MCA, VL-bus, VME, SCSI и др. Для подключения видеооборудования используется шина AGP.

При построении систем управления и контроля необходимо выбрать один из интерфейсов, разработать функциональные и принципиальные схемы, провести подбор и расчет элементов схем, создать программное обеспечение.

Приведем один из вариантов УСО на основе параллельного или системного интерфейса в виде структурной схемы (рис.1.1).

Внешние данные поступают в компьютер, минуя пользователя, непосредственно от измерительных приборов или других устройств, фиксирующих параметры объекта; сигналы управления переключают или устанавливают заданные параметры объекта. На представленной структурной схеме предполагается, что данные должны преобразовываться в аналоговую форму, усиливаться и сохранять объект в определенном состоянии, а измерительная часть связана с устройствами (датчиками), имеющими на своих выходах аналоговые величины. Следует отметить, что данная конфигурация может изменяться в зависимости от природы объектов и характера сигналов. Например, может потребоваться, чтобы устройства гальванической развязки находились бы на входах объектов управления или были бы подключены непосредственно к схемам аналоговых коммутаторов.

Основные функциональные узлы УСО, представленного на рис.1.1, следующие:

1. Порт ПК – стандартный или системный интерфейс компьютера.

2. Схема гальванической развязки – схема, обеспечивающая полное гальваническое разделение УСО и порта ПК во избежание влияния опасных напряжений и токов на системную плату ПК, что может привести к выходу из строя всего ПК.

3. Шинные формирователи - формируют цифровые сигналы, усиливают импульсы по току, обслуживают «энергоёмкие» цифровые нагрузки. Такими нагрузками являются, прежде всего, шины данных, состоящие из нескольких токоведущих дорожек на печатной плате.

4. Буферный регистр выходных данных – по сигналу управления производит чтение битов данных и адреса, «записывает» их и хранит до следующего сигнала управления.

5. Селектор адреса – производит выбор (селекцию) определенного адреса объекта управления, на который будут передаваться данные. В случае большого количества ОУ селектор адреса дополняется схемой выбора кристалла, мультиплексором и дешифратором. В этом случае селектор адреса используется для выбора адреса группы управления.

6. Выбор кристалла адреса – используется для выбора конкретного кристалла в одной, определенной селектором адреса, группе адресов.

7. Мультиплексор – используется для увеличения количества групп управляемых объектов.

8. Дешифратор -- микросхема средней степени интеграции, предназначенная для преобразования двоичного кода в напряжение логического уровня на том выходе, десятичный номер которого соответствует двоичному коду.

9. Адрес группы схемы контроля. Выбор кристалла схемы контроля, мультиплексор и схема дешифрации адреса схемы контроля, которые используются аналогично схемам выбора адреса объекта управления, но только данные схемы выбирают адрес контролируемого объекта.

10. Формирователи уровня сигнала – устройства (как правило, шинные формирователи), которые обеспечивают формирование уровня выдаваемых сигналов до уровня, необходимого для управления объектами (при большом количестве объектов управления необходимо обеспечить допустимые коэффициенты разветвления сигналов в зависимости от используемой элементной базы).

11. АЦП, ЦАП – аналого-цифровые и цифро-аналоговые преобразователи.

12. Аналоговые коммутаторы – применяются для коммутации аналоговых величин в измерительных комплексах. Обычно это интегральные коммутаторы или релейные матрицы.

Анализ функциональных схем позволяет находить наиболее оптимальные решения по выбору элементной базы, а возможность программирования некоторых фаз в работе (например, исключение аппаратного генератора тактовых импульсов с заменой его на программный) приводит к упрощению схемы на этапе реализации УСО.

Разработка устройств сопряжения предусматривает *два способа взаимодействия компьютера и исполнительных устройств. Первый способ* основан на создании устройства для включения в слоты расширения ПК (ISA, EISA, PCI), т.е. реализуется на основе интерфейсного модуля, включаемого в шину системного интерфейса персонального компьютера.

В качестве недостатков этого способа можно отметить следующие:

- для устройства сопряжения (УСО) следует применять промышленный способ изготовления печатных плат (требуется высокая плотность монтажа, что не всегда возможно в лабораторных условиях);

- в современных ПК может возникнуть проблема, связанная с недостатком свободных аппаратных прерываний, не говоря уже о сложности настройки подобного устройства, особенно если устройство работает в режиме аппаратных прерываний;

- на материнских платах формата microATX/AT может просто не хватить места для платы сопряжения (на таких платах обычно устанавливается всего 1-2 PCI или ISA-разъема), а также возникнет проблема с физическим размещением платы в корпусе вследствие его малых габаритов;

- многие производители материнских плат для современных ПК не включают в состав своих изделий шину ISA (хотя она широко применяется в промышленных компьютерах), следовательно, раз-

работчик должен быть уверен в том, с какой компьютерной платформой ему придется работать.

В качестве достоинств можно назвать следующие:

- освобождение внешних интерфейсов ввода-вывода, что решает проблему подключения периферийных устройств;
- разрядность шин определяет большое адресное пространство, т.е. возможность подключения (адресации) большего (относительно COM и LPT) числа устройств;
- большая (относительно COM и LPT) скорость передачи данных.

Говоря о достоинствах и недостатках системного интерфейса, следует помнить, что если он и дает большие возможности работы с устройствами, но зато предъявляет более жесткие требования как к аппаратной части, так и к программному обеспечению.

Второй способ реализации УСО заключается в использовании внешних периферийных интерфейсов (параллельных – LPT-порт, последовательных – COM, USB). Он предусматривает применение стандартных портов и программирование контроллера ввода-вывода.

Отметим недостатки:

- количество внешних интерфейсов ввода-вывода ограничено (LPT, COM), однако на данный момент эти порты активно используются стандартными устройствами (модемы, мыши, принтеры, сканеры и др.);
- в большинстве случаев требуется внешний источник питания, если устройство потребляет много электроэнергии или порт не оснащен цепями источника питания;
- невысокая скорость передачи данных и малое адресное пространство ограничивают область применения и функциональность.

Достоинства:

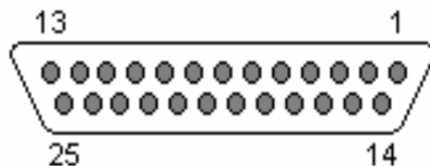
- практически неограниченные физические размеры УСО;
- простота физической реализации монтажных схем;
- более безопасны в случае нештатных ситуаций (короткое замыкание цепей питания на плате УСО), так как при самых неблагоприятных условиях выйдет из строя только устройство или контроллер порта ввода-вывода, но ни в коем случае не вся материнская плата.

Как видно из вышеизложенного, в наличии разработчика УСО имеются различные варианты использования портов ПК для подключения внешних устройств как по реализации (с точки зрения скорости подключения) и адресуемости подключаемых устройств, так и по удобству разработки и внедрения. Для создания УСО необходимо определиться с информационной емкостью объектов управления и контроля, с необходимой скоростью передачи данных и адресов, природой сигналов. Эти данные приведены в перечне вариантов задания на разработку УСО.

2. ОРГАНИЗАЦИЯ СТАНДАРТНЫХ ИНТЕРФЕЙСОВ ВВОДА-ВЫВОДА

2.1. Параллельный интерфейс LPT

Рассмотрим внешний вид разъема и обозначение контактов LPT-порта (рис.2.1.).



Pin	Name	Description	Register Bit
1	STROBE	Strobe	CR:0\
2	D0	Data Bit 0	DR:0
3	D1	Data Bit 1	DR:1
4	D2	Data Bit 2	DR:2
5	D3	Data Bit 3	DR:3
6	D4	Data Bit 4	DR:4
7	D5	Data Bit 5	DR:5
8	D6	Data Bit 6	DR:6
9	D7	Data Bit 7	DR:7
10	ACK	Acknowledge	SR:6
11	BUSY	Busy	SR:7\
12	PE	Paper End	SR:5
13	SEL	Select	SR:4
14	AUTOFD	Autofeed	CR:1\
15	ERROR	Error	SR:3
16	INIT	Initialize	CR:2
17	SELIN	Select In	CR:3\
18	GND	Signal Ground	-
19	GND	Signal Ground	-
20	GND	Signal Ground	-
21	GND	Signal Ground	-
22	GND	Signal Ground	-
23	GND	Signal Ground	-
24	GND	Signal Ground	-
25	GND	Signal Ground	-

\ - инвертированные сигналы

Рис.2.1. LPT-порт

Параллельный порт часто называют интерфейсом Centronics по имени фирмы разработчика. Понятие Centronics относится как к набору сигналов и протоколу взаимодействия, так и к 36-контактному разъему, устанавливаемому на принтерах.

Порт параллельного интерфейса был введен в РС для подключения принтера — отсюда и пошло его название LPT-порт (Line Printer — построчный принтер).

Адаптер параллельного интерфейса представляет собой набор регистров, расположенных в пространстве ввода-вывода. Регистры порта адресуются относительно базового адреса порта, стандартными значениями которого являются 378h и 278h (LPT1, LPT2). Порт может использовать линию запроса аппаратного прерывания, обычно IRQ7 или IRQ5. Внешний интерфейс порта имеет 8-битную шину данных, 5-битную шину сигналов состояния и 4-битную шину управляющих сигналов [3].

BIOS поддерживает до четырех (иногда до трех) LPT-портов (LPT1-LPT4) специальным сервисом (прерывание INT 17h), обеспечивающим связь с принтерами по интерфейсу Centronics. Этим сервисом BIOS осуществляет вывод символа (по опросу готовности, не используя аппаратных прерываний), инициализацию интерфейса и принтера, а также опрос состояния принтера.

Традиционный (стандартный) порт SPP (Standard Parallel Port) является однонаправленным портом, на базе которого программно реализуется протокол обмена Centronics. Сигналы порта выводятся на разъем DB-25S (розетка), установленный непосредственно на плате адаптера (или системной плате) или соединяемый с ней плоским шлейфом. Название и назначение сигналов разъема порта соответствуют интерфейсу Centronics.

Стандартный порт имеет три 8-битных регистра, расположенных по соседним адресам, начиная с базового адреса порта (BASE):

Data Register (DR) — регистр данных (адрес = BASE). Информация, записанная в этот порт, выводится на выходные линии интерфейса. Данные, считанные из этого регистра, в зависимости от схемотехники адаптера соответствуют либо ранее записанным данным, либо сигналам на тех же линиях, что не всегда одно и то же. Если в порт записать байт с единицами во всех разрядах, а на выходные линии интерфейса через микросхемы с выходом типа «открытый коллектор» подать какой-либо код (или соединить ключами какие-то линии со схемной землей), то этот код может быть считан из того же регистра данных. Таким образом, на многих старых моделях адаптеров можно реализовать порт ввода дискретных сигналов, однако выходным цепям передатчика информации придется «бороться» с выходным током логической единицы выходных буферов адаптера. Схемотехника ТТЛ такие решения строго не запрещает, но если внешнее устройство выполнено на микросхемах КМОП, их мощно-

сти может и не хватить в этом шинном конфликте. Однако современные адаптеры часто имеют в выходной цепи согласующий резистор с сопротивлением до 50 Ом. Выходной ток короткого замыкания выхода на землю обычно ограничен передатчиком на уровне около 30 мА. Простой расчет показывает, что в случае даже короткого замыкания контакта разъема на землю при выводе «единицы» на этом резисторе падает напряжение до 1,5 В, что входной схемой приемника будет воспринято как та же «единица». Так что не стоит рассчитывать на то, что такой способ ввода будет работать на всех компьютерах со стандартными портами.

Status Register (SR) — регистр состояния, представляющий собой 5-битный порт ввода сигналов состояния принтера (биты SR4-SR7, адрес = BASE+1). Бит SR7 инвертируется — низкому уровню сигнала соответствует «единичное» значению бита в регистре, и наоборот.

Назначение бит регистра состояния приведены в таблице 2.1.

Таблица 2.1.

Назначение бит регистра состояния

№ контакта	Назначение
10	<i>Ack</i> – сигнал отображения состояния линии <i>Ack</i>
11	<i>Basy</i> – инверсный бит отображения состояния линии <i>Basy</i> ; при низком уровне на линии устанавливается единичное значение бита – разрешение на вывод очередного байта
12	<i>Pe</i> – сигнал отображения состояния линии <i>Paper End</i> . Единичное значение соответствует высокому уровню линии – сигналу о конце бумаги в принтере
13	<i>Select</i> – сигнал отображения состояния линии <i>Select</i> . Единичное значение соответствует высокому уровню линии – сигналу о включении принтера
15	<i>Error</i> – сигнал отображения состояния линии <i>Error</i> , нулевое значение соответствует низкому уровню линии - сигналу о любой ошибке принтера

Control Register (CR) — регистр управления (адрес = BASE+2). Как и регистр данных, этот 4-битный порт вывода допускает запись и чтение (биты 0-3), но его выходной буфер обычно имеет тип “открытый коллектор”. Это позволяет более корректно использовать линии данного регистра как входные при программировании их в высокий уровень. Биты 0, 1, 3 инвертируются — единичному значению в регистре соответствует низкий уровень сигнала, и наоборот.

Назначение бит регистра управления приведено в табл. 2.2.

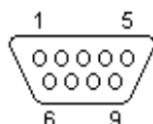
Таблица 2.2.

Назначение бит регистра управления

№ контакта	Назначение
1	<i>Strobe</i> – единичное значение бита соответствует низкому уровню на выходе <i>Strobe</i> – сигналу стробирования выходных данных
14	<i>AutoFD</i> – единичное значение бита соответствует низкому уровню на выходе <i>AutoLF</i> – сигналу на автоматический перевод строки (LF – Line Feed) по приему байта возврата каретки
16	<i>Init</i> – нулевое значение бита соответствует низкому уровню на выходе аппаратного сброса принтера
17	<i>Selin</i> – единичное значение бита соответствует низкому уровню на выходе <i>Select In</i> – сигналу, разрешающему работу принтера по интерфейсу Centronics

2.2. Последовательный интерфейс COM

Рассмотрим внешний вид и обозначения контактов (рис.2.2.).



Pin	Name	Description
1	CD	Carrier Detect
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	System Ground
6	DSR	Data Set Ready
7	RTS	Request to Send
8	CTS	Clear to Send
9	RI	Ring Indicator

Рис.2.2. Последовательный порт

Назначение контактов разъемов COM-портов приведено в табл. 2.3.

Назначение сигналов интерфейса RS-232C

№ контакта	Назначение
1	<i>Data Carrier Detect</i> – вход сигнала обнаружения несущей удаленного модема
2	<i>Reactive Data</i> – последовательные данные - вход приемника
3	<i>Transmit Data</i> – последовательные данные - выход передатчика
4	<i>Data Terminal Ready</i> - выход сигнала готовности терминала к обмену данными. Состояние «включено» поддерживает коммутируемый канал в состоянии соединения
5	<i>Signal Ground</i> – сигнальная (схемная) земля, относительно которой действуют уровни сигналов
6	<i>Data Set Ready</i> – вход сигнала готовности от аппаратуры передачи данных (модем в рабочем режиме подключен к каналу и закончил действия по согласованию с аппаратурой на противоположном конце канала)
7	<i>Request To Send</i> – выход запроса передачи данных: состояние «включено» уведомляет модем о наличии у терминала данных для передачи. В полудуплексном режиме используется для управления направлением – состояние «включено» служит сигналом модему на переключение в режим передачи
8	<i>Clear To Send</i> – вход разрешения терминалу передавать данные. Состояние «выключено» аппаратно запрещает передачу данных. Сигнал используется для аппаратного управления потоками данных
9	<i>Ring Indicator</i> – вход индикатора вызова (звонка). В коммутируемом канале этим сигналом модем сигнализирует о принятии вызова

Последовательный интерфейс для передачи данных в одну сторону использует одну сигнальную линию, по которой информационные биты передаются друг за другом последовательно. Такой способ передачи и определяет название интерфейса и порта, его реализующего. Эти названия соответствуют английским терминам Serial Interface и Serial Port. Последовательная передача данных может осуществляться как в асинхронном, так и синхронном режимах [3, 4].

При асинхронной передаче каждому байту предшествует стартовый бит, сигнализирующий приемнику о начале очередной посылки, за которым следуют биты данных и, возможно, бит паритета (кон-

троля четности). Завершает посылку стоп-бит, гарантирующий определенную выдержку между соседними посылками. Старт-бит следующего посланного байта может посылаться в любой момент после окончания стоп-бита, то есть между передачами возможны паузы произвольной длительности. Старт-бит, имеющий всегда строго определенное значение (логический «0»), обеспечивает простой механизм синхронизации приемника по сигналу от передатчика. Подразумевается, что приемник и передатчик работают на одной скорости обмена, измеряемой количеством передаваемых бит в секунду. Внутренний генератор синхронизации приемника использует счетчик-делитель опорной частоты, обнуляемый в момент приема начала старт-бита. Этот счетчик генерирует внутренние стробы, по которым приемник фиксирует последующие принимаемые биты. В идеале эти стробы располагаются в середине битовых интервалов, что обеспечивает возможность приема данных и при некотором рассогласовании скоростей приемника и передатчика. При передаче 8 бит данных, одного контрольного и одного стоп-бита предельно допустимое рассогласование скоростей, при котором данные будут распознаны верно, не может превышать 5 %.

Контроль формата позволяет обнаруживать обрыв линии: при этом обычно принимается логический «0», который сначала трактуется как старт-бит и нулевые биты данных, но потом сработает контроль стоп-бита. Для асинхронного режима принят ряд стандартных скоростей обмена: 50, 75, 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 и 115200 бит/с. Количество бит данных может составлять 5, 6, 7 или 8 (5- и 6-битные форматы малораспространены). Количество стоп-бит может быть 1, 1,5 и 2 («полтора бита» - здесь подразумевается только длительность стопового интервала).

Асинхронный обмен в РС реализуется с помощью СОМ порта с использованием протокола RS-232С. Дальность и скорость передачи информации зависит от типа применяемого соединительного кабеля (табл.2.1.).

Таблица 2.4

Зависимость дальности и скорости передачи по СОМ порту

Скорость передачи, бит/с	Максимальная длина для экранированного кабеля, м	Максимальная длина для неэкранированного кабеля, м
110	1525	915
300	1525	915
1200	915	915
2400	305	152
4800	305	76
9600	76	76

Синхронный режим передачи предполагает постоянную активность канала связи. Посылка начинается с синхробайта, за которым вплотную следует поток информационных бит. Если у передатчика нет данных для передачи, он заполняет паузу непрерывной посылкой байтов синхронизации. Очевидно, что при передаче больших массивов данных накладные расходы на синхронизацию в данном режиме обмена будут ниже, чем в асинхронном. Однако в синхронном режиме необходима внешняя синхронизация приемника с передатчиком, поскольку, даже малое отклонение частот, приведет к быстро накапливающейся ошибке и искажению принимаемых данных.

Последовательный интерфейс на физическом уровне может иметь различные реализации, различающиеся способами передачи электрических сигналов. Существует ряд родственных международных стандартов: RS-232C, RS-423A, RS-422A и RS-485. Наибольшее распространение в ПК получил простейший из этих — стандарт RS-232C.

2.3. Последовательный интерфейс USB

USB (Universal Serial Bus — универсальная последовательная шина) является промышленным стандартом расширения архитектуры PC, ориентированным на интеграцию с телефонией и устройствами бытовой электроники. Шина разработана сравнительно недавно - спецификация версии 1.0 была опубликована в январе 1996 г. - и в ней отразились современные достижения различных областей компьютерной техники. В 2002 году вышли первые продукты, использующие спецификацию шины версии 2.0.

Архитектура USB определяется следующими критериями:

- легко реализуемое расширение периферии PC;
- дешевое решение, поддерживающее скорость передачи до 12Мбит/с (версия 2.0 обеспечивает скорость передачи 480 Мбит/с);
- полная поддержка в реальном времени передачи аудио- и сжатых видео-данных;
- гибкость протокола для смешанной передачи изохронных данных и асинхронных сообщений;
- интеграция в технологию выпускаемых устройств;
- доступность в PC всех конфигураций и размеров;
- обеспечение стандартного интерфейса, способного быстро внедряться в продукцию;
- открытие новых классов устройств, расширяющих ПК.

С точки зрения конечного пользователя привлекательны такие свойства USB:

- простота кабельной системы и подключений;

- изоляция подробностей электрического подключения от конечного пользователя;
- самоидентифицирующаяся периферия, автоматическая связь устройств с драйверами и автоконфигурирование;
- возможность динамического подключения и реконфигурирования периферии.

С 1996 года многие фирмы выпускают ПК со встроенным контроллером USB, реализуемым чипсетом системной платы.

2.4. Структура и взаимодействие системы USB

USB обеспечивает обмен данными между хост-компьютером и множеством одновременно доступных периферийных устройств (до 127). Распределение пропускной способности шины между подключенными устройствами планируется хостом и реализуется им с помощью посылки маркеров. Шина позволяет подключать, конфигурировать, использовать и отключать устройства во время работы хоста и самих устройств — динамическое (или «горячее») подключение и отключение.

Устройства (Device) USB могут являться хабами, «функциями» или их комбинацией. Хаб (Hub) обеспечивает дополнительные точки подключения устройств к шине. «Функции» (Function) USB предоставляют системе дополнительные возможности: например подключение к цифровому джойстику, акустическим колонкам с цифровым интерфейсом и т. п. Устройство USB должно иметь интерфейс USB, обеспечивающий полную поддержку протокола USB, выполнение стандартных операций (конфигурирование и сброс) и стандартное представление информации, описывающей устройство. Многие устройства, подключаемые к USB, имеют в своем составе и хаб, и «функции».

Работой всей системы USB управляет хост-контроллер (host controller), являющийся программно-аппаратной подсистемой хост-компьютера. Физическое соединение устройств USB осуществляется по топологии многоярусной звезды (рис.2.3.).

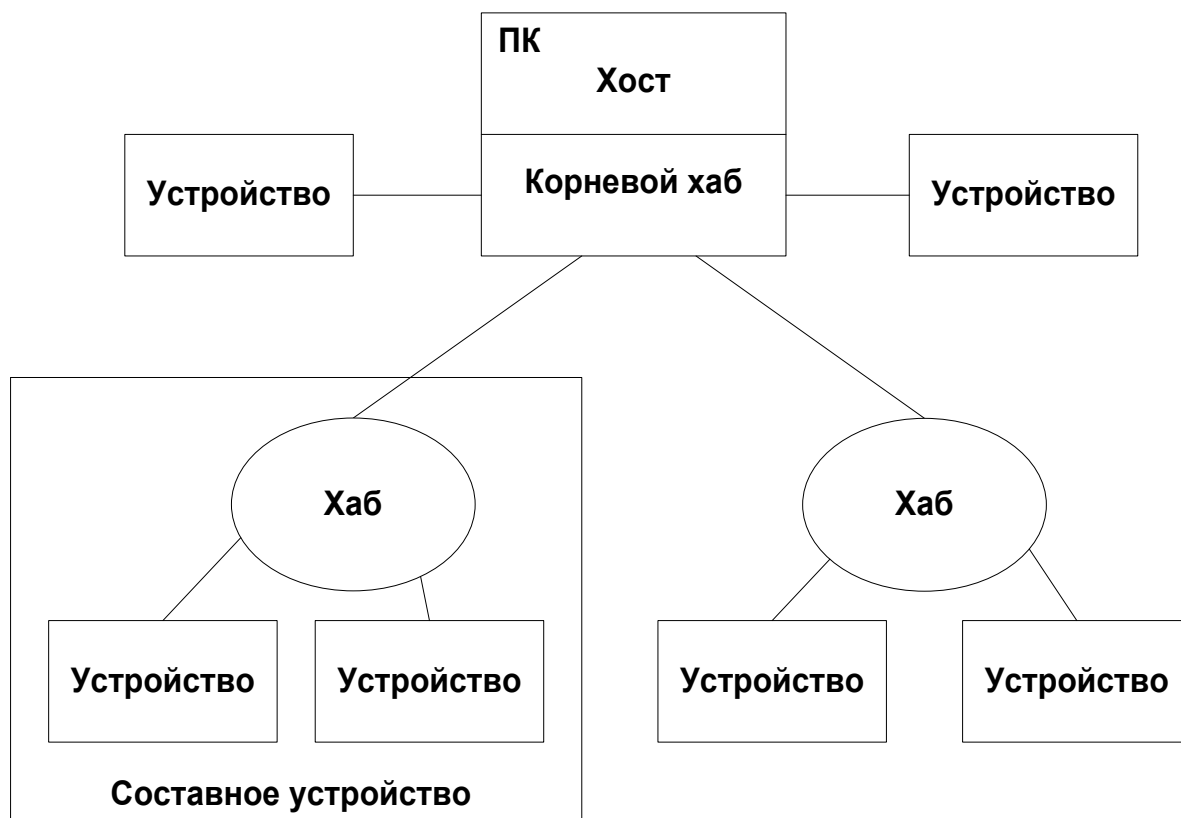


Рис.2.3. Способы подключения USB устройств

Центром каждой звезды является хаб, каждый кабельный сегмент соединяет две точки: хаб с другим хабом или с функцией. В системе с USB имеется один (и только один) хост-контроллер, расположенный в вершине пирамиды устройств и хабов USB. Хост-контроллер интегрируется с корневым хабом (root hub), обеспечивающим одну или несколько точек подключения — портов. Контроллер USB, входящий в состав чипсетов многих современных системных плат, обычно имеет встроенный двухпортовый хаб.

Логически устройство, подключенное к любому хабу USB и сконфигурированное, может рассматриваться как непосредственно подключенное к хост-контроллеру.

«Функции» представляют собой устройства USB, способные передавать или принимать данные или управляющую информацию по шине. Типично «функции» представляют собой отдельные периферийные устройства с кабелем, подключаемым к порту хаба. Однако физически в одном корпусе может быть и несколько «функций» со встроенным хабом, обеспечивающим их подключение к одному порту. Эти комбинированные устройства для хоста являются хабами с постоянно подключенными устройствами-«функциями».

Каждая «функция» предоставляет конфигурационную информацию, описывающую возможности и требования устройства к ресурсам. Перед использованием «функция» должна быть сконфигуриро-

вана хостом — ей должна быть выделена полоса в канале и выбраны специфические опции конфигурации.

Хаб — ключевой элемент системы Plug-and-Play в архитектуре USB. Хаб является кабельным концентратором, точки подключения называются портами хаба. Каждый хаб преобразует одну точку подключения в их множество. Архитектура подразумевает возможность соединения нескольких хабов.

У каждого хаба имеется один восходящий порт (upstream port), предназначенный для подключения к хосту или хабу верхнего уровня. Остальные порты являются нисходящими (downstream ports) и предназначены для подключения функций или хабов нижнего уровня. Хаб может распознать подключение или отключение устройств к этим портам и управлять подачей питания на их сегменты. Каждый из этих портов индивидуально может быть разрешен или запрещен и сконфигурирован на полную или ограниченную скорость обмена. Хаб обеспечивает изоляцию сегментов с низкой скоростью от высокоскоростных.

Хабы могут иметь возможность управления подачей питания на нисходящие порты, причем предусматривается управляемая установка ограничения на ток, потребляемый каждым портом.

Реализацию хаба рассмотрим на примере структурной схемы хаба 8х931НА фирмы Intel (рис.2.4.).

Система USB разделяется на три уровня с определенными правилами взаимодействия (аналогично семиуровневой модели OSI). Устройство USB делится на интерфейсную часть, часть устройства и собственно функциональную часть. Хост тоже делится на три части - интерфейсную, системную и ПО устройства. Каждая часть отвечает только за определенный круг задач.

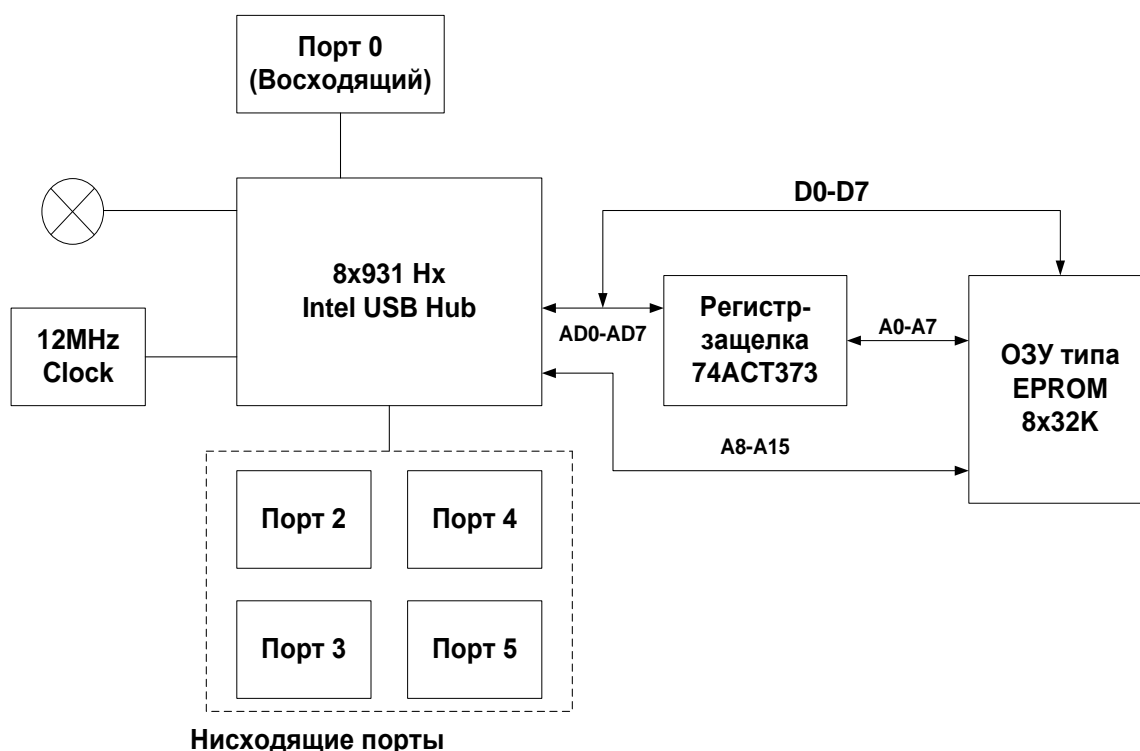


Рис.2.4. Структурная схема USB хаба

Стандарт USB определяет электрические и механические спецификации шины. Информационные сигналы и цепи питающего напряжения 5В передаются по четырехпроводному кабелю (рис.2.5.)

Вывод	Название	Цвет провода	Описание
1	VCC	Красный	+5В
2	D-	Белый	Данные -
3	D+	Зеленый	Данные +
4	GND	Черный	Земля

Рис.2.5. Назначение и обозначение выводов USB порта

Для сигнала используются дифференциальный способ передачи по двум проводам D+ и D-. Уровни сигналов передатчиков в статическом режиме должны быть ниже 0.3 В (низкий уровень) или выше 2.8 В (высокий уровень). Приемники должны выдерживать входное напряжение в пределах -0.5...+3.8 В. Передатчики должны иметь возможность перехода в высокоимпедансное состояние для обеспечения двунаправленной полудуплексной передачи данных по одной паре проводов.

Передача по двум проводам USB не ограничивается лишь дифференциальными сигналами. Кроме дифференциального приемника, каждое устройство имеет и линейные приемники сигналов D+ и D-, а передатчики этих линий управляются индивидуально. Это позволяет различать множество состояний линии, используемых для организации аппаратного интерфейса.

Шина имеет два режима передачи. Полная скорость передачи сигналов USB составляет 12 Мбит/с, низкая - 1,5 Мбит/с. Для полной скорости используется экранированная витая пара с импедансом 90 Ом и длиной сегмента до 5 м, для низкой - невитой и неэкранированный кабель при длине сегмента до 3 м. Низкоскоростные кабели и устройства дешевле высокоскоростных. Одна и та же система может одновременно использовать оба режима, переключение для устройств осуществляется прозрачно. Низкая скорость предназначена для работы с небольшим количеством устройств, не требующих высокой пропускной способности канала.

Скорость, используемая устройством, подключенным к конкретному порту, определяется хабом по уровням сигналов на линиях D+ и D-, смещаемых нагрузочными резисторами приемопередатчиков.

Кроме сигнальной пары, кабель имеет линии V и GND для передачи питающего напряжения 5 В к устройствам.

USB поддерживает подключение и отключение устройств в процессе работы. Нумерация устройств шины является постоянным процессом, отслеживающим динамические изменения физической топологии.

Все устройства USB подключаются через порты хабов. Хабы определяют подключение и отключение устройств к своим портам и сообщают состояние портов в ответ на запрос от контроллера. Хост разрешает работу порта и адресуется к устройству через канал управления, используя нулевой адрес — USB Default Address. Все устройства адресуются через нулевой адрес при начальном подключении или после сброса.

Хост определяет, является ли новое подключенное устройство хабом или функцией и назначает ему уникальный адрес USB. Хост устанавливает с этим устройством канал управления (control pipe), используя назначенный адрес и нулевой номер точки назначения.

Если новое устройство является хабом, хост определяет подключенные к нему устройства, назначает им адреса и устанавливает каналы. Если новое устройство является «функцией», уведомление о подключении передается диспетчером USB заинтересованному программному обеспечению.

Когда устройство отключается, хаб автоматически запрещает соответствующий порт и сообщает об отключении контроллеру, который удаляет сведения о данном устройстве из всех структур данных. Если отключается хаб, то процесс удаления выполняется для

всех подключенных к нему устройств. Если отключается «функция», уведомление посылается заинтересованному ПО.

Нумерация устройств, подключенных к шине (bus enumeration), осуществляется динамически по мере их подключения (или включения их питания) без какого-либо вмешательства пользователя или клиентского ПО.

Когда устройство отключается от шины, хаб уведомляет об этом хост и работа порта запрещается, а хост обновляет свою текущую топологическую информацию.

Хост – компьютер общается с устройствами через контроллер. Хост имеет следующие обязанности:

- обнаружение подключения и отключения устройства USB;
- манипулирование потоком управления между устройствами и хостом;

- управление потоками данных;

- сбор информации о состоянии и статистике;

- обеспечение энергосбережения подключенными устройствами.

Системное ПО контроллера управляет взаимодействием между устройствами и их ПО, функционирующим на хост-компьютере.

Области взаимодействия следующие:

- нумерация и конфигурация устройств;

- изохронные передачи данных;

- управление энергопотреблением;

- информация об управлении устройствами и шиной.

Что касается использования шины USB, то здесь следует отметить сложность создания контроллера, который бы отвечал поставленным задачам, а также наличие большого количества очень недорогих контроллеров USB, выпускаемых серийно многими западными фирмами. Тем более, что концепция «все в одном» подразумевает интеграцию как можно большего числа электронных элементов в одну микросхему

Предпринимаемые до этого попытки по созданию работоспособного устройства, связывающее микропроцессор с компьютером, у многих разработчиков окончились неудачей. Это связано со сложностью USB как со стороны аппаратных решений, так и со стороны математического обеспечения. В качестве доказательства можно отметить, что известная микросхема USBN9603/USBN9604 фирмы National Semiconductor имеет в своём составе около 50 регистров, которыми должен управлять микропроцессор для того, чтобы обеспечить связь с компьютером по USB [10].

Структурная схема контроллера на базе микросхемы USBN9602 приведена на рис.2.6. Это интегрированный USB-контроллер, совместимый со спецификациями версий 1.0 и 1.1. На одном кристалле содержатся все необходимые модули для поддержки USB-протокола: трансивер, контроллер доступа, FIFO, универсальный

восьмиразрядный параллельный интерфейс, интерфейс Microwire/Plus и встроенный тактовый генератор. Всего 7 буферов FIFO позволяют организовать поддержку различных USB сообщений: один двунаправленный FIFO - для принудительного контроля и шесть FIFO буферов - в качестве дополнительных шести каналов для поддержки прерываний и передачи большого объема данных в асинхронном режиме. Параллельный интерфейс поддерживает мультиплексный и демultipлексный режимы работы шин адреса и данных CPU. Благодаря программируемой схеме прерываний возможна настройка под различные требования к интерфейсу.

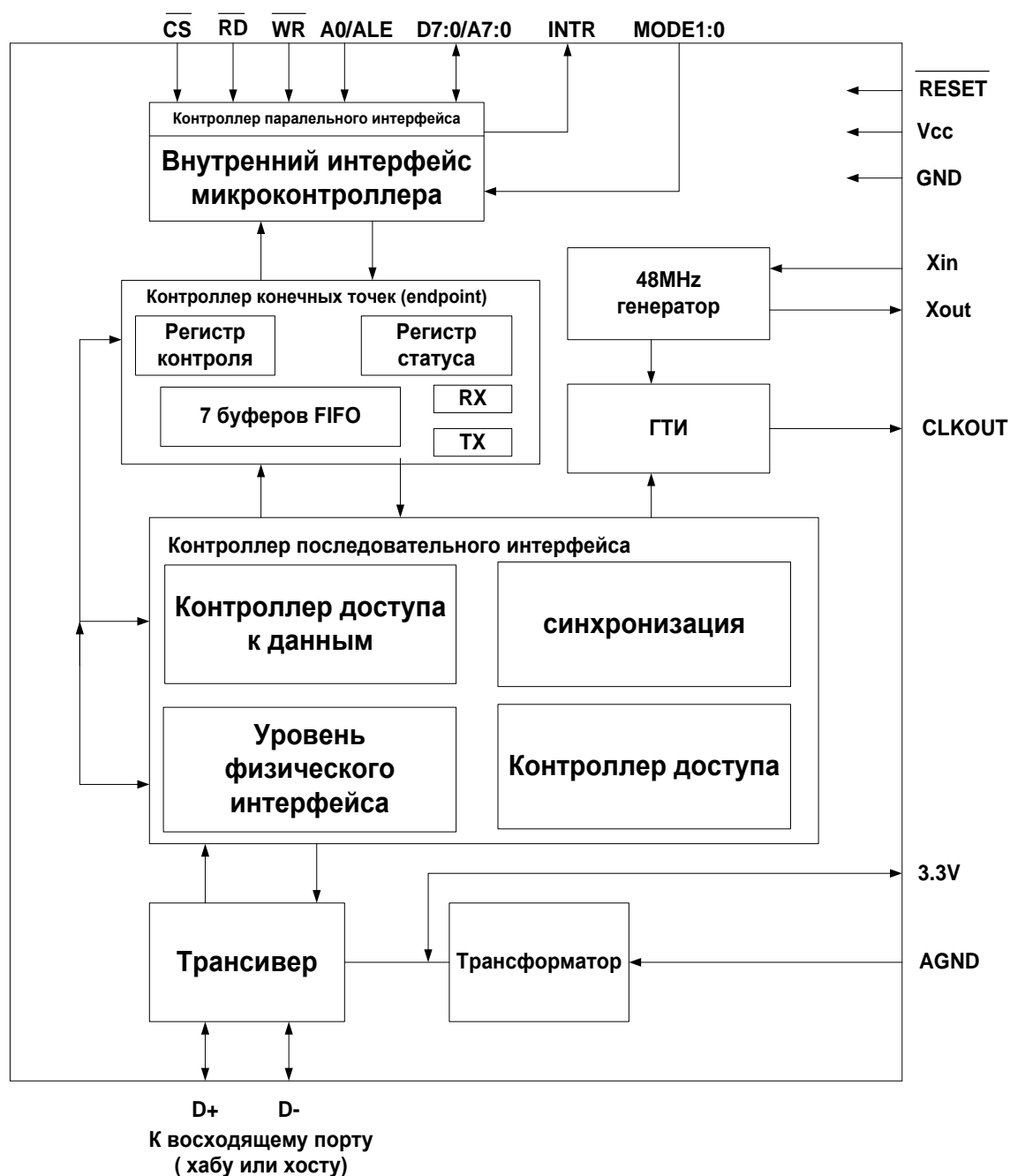


Рис.2.6. Структурная схема USB-контроллера

3. ПРИМЕНЕНИЕ ШИН СИСТЕМНОГО ИНТЕРФЕЙСА ПК

Одной из первых шин персонального компьютера является системная шина IBM PC/XT [5]. Она обеспечивает передачу данных по восьми разрядам. Кроме того, эта шина включает двадцать адресных линий, которые ограничивают адресные пространства пределом в один мегабайт. Для работы с внешними устройствами в ней предусмотрены четыре линии аппаратных прерываний и четыре линии запроса внешними устройствами прямого доступа к памяти. Для подключения плат расширения используется специальные 62-контактные щелевые разъемы. Микропроцессор и системная шина синхронизируются от одного тактового генератора с частотой 4.77МГц. Таким образом, теоретическая скорость передачи данных могла достигать не более 4 Мбайт/с.

Основным промышленным стандартом стала применяться системная шина ISA Bus (Industry Standard Architecture) — шина расширения, применявшаяся в первых моделях PC [3]. Она отличается наличием второго 36-контактного дополнительного разъема, за счет чего количество адресных линий увеличено на четыре, данных — на восемь. Количество линий аппаратных прерываний увеличилось до 15, а каналов прямого доступа до 7. Конструктивно шина выполнена в виде двух щелевых разъемов (слотов) с шагом выводов 2,54 мм (0,1 дюйма). Подмножество ISA-8 использует только 62-контактный слот (ряды А, В), в ISA-16 применяется дополнительный слот (ряды С, D) (рис.3.1.).

Шина обеспечивает своим абонентам возможность отображения 8- или 16-битных регистров на пространство ввода/вывода и памяти. Диапазон адресов памяти ограничен областью UMA, но для шины ISA-16 специальными опциями BIOS Setup может быть разрешено и пространство в области между 15-м и 16-м мегабайтом памяти (правда, при этом компьютер не сможет использовать более 15 Мбайт ОЗУ). Диапазон адресов ввода/вывода сверху ограничен количеством используемых для дешифрации бит адреса, нижняя граница ограничена областью адресов 0—FFh, зарезервированных под устройства системной платы. В PC была принята 10-битная адресация ввода/вывода, при которой линии адреса A[15:10] устройствами игнорировались. Таким образом, диапазон адресов устройств шины ISA ограничивается областью 00h—3FFh, то есть всего 768 адресов 8-битных регистров. Впоследствии стали применять и 12-битную адресацию (диапазон 00h—FFFh), но при ее использовании всегда необходимо учитывать возможность присутствия на шине и старых 10-битных адаптеров, которые «отзовутся» на адрес с подходящими ему битами A[9:0] во всей допустимой области 12-битного адреса четыре раза.

В распоряжении абонентов шины ISA-16 11 линий запросов прерываний IRQx. Заметим, что при конфигурировании BIOS Setup часть из этих запросов могут отобрать устройства системной платы или шина PCI.

Абоненты шины могут использовать до трех 8-битных каналов DMA, а на 16-битной шине могут быть доступными еще три 16-битных канала. Сигналы 16-битных каналов могут использоваться и для получения прямого управления шиной устройством *Bus-Master*.

При этом канал DMA используется для обеспечения арбитража управления шиной, а адаптер Bus-Master формирует все адресные и управляющие сигналы шины, не забывая передать управление шиной процессору не более чем на 15 микросекунд (чтобы не нарушить регенерацию памяти).

Все перечисленные ресурсы системной шины должны быть бесконфликтно распределены между абонентами. Бесконфликтность подразумевает следующее:

- каждый абонент должен при операциях чтения управлять шиной данных (выдавать информацию) только по своим адресам или по обращению к используемому им каналу DMA. Области адресов для чтения не должны пересекаться;

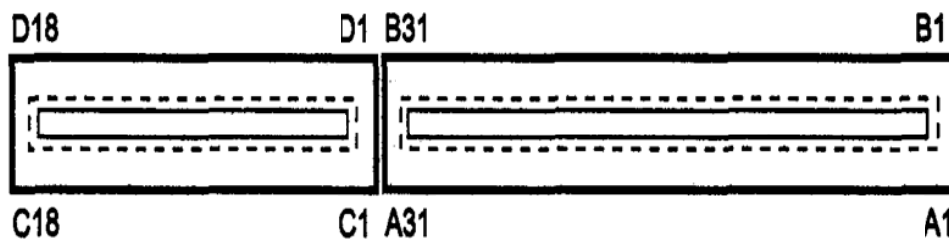
- назначенную линию запроса прерывания IRQx или прямого доступа DRQx абонент должен держать на низком уровне в пассивном состоянии и переводить в высокий уровень для активации запроса. Неиспользуемыми линиями запросов абонент управлять не имеет права, они должны быть электрически откоммутированы или подключены к буферу, находящемуся в третьем состоянии. Одной линией запроса может пользоваться только одно устройство.

Набор сигналов 16-битной шины ISA предельно прост. Непосредственно к программному обращению к ячейкам памяти и пространства ввода/вывода относятся следующие сигналы:

- Data[16:0] — шина данных;
- Addr[24:0] — шина адреса;
- IOWR# — запись в порт;
- IORD# — чтение порта;
- SMemWR# — запись в память (в диапазоне адресов 0-FFFFFh);
- SmemRD# — чтение памяти (в диапазоне адресов 0-FFFFFh).

Дополнительный разъем, расширяющий шину до 16-битной, содержит дополнительные линии данных, адреса запросов прерываний и каналов прямого доступа:

- Data [15:8] — шина данных;
- SBHE — признак наличия данных на линиях Data [15:8];



Pin	Name	Description	Pin	Name	Description
A1	I/O CH	I/O channel check	B1	GND	Ground
A2	D7	Data bit 7	B2	RESET	RESET
A3	D6	Data bit 6	B3	+5V	+5 VDC
A4	D5	Data bit 5	B4	IRQ2	Interrupt Request
A5	D4	Data bit 4	B5	-5VDC	-5 VDC
A6	D3	Data bit 3	B6	DRQ2	DMA Request
A7	D2	Data bit 2	B7	-12VDC	-12 VDC
A8	D1	Data bit 1	B8	NOWS	No WaitState
A9	D0	Data bit 0	B9	+12VDC	+12 VDC
A10	I/O CH	I/O Channel ready	B10	GND	Ground
A11	AEN	Address enable	B11	SMEMW	System Memory Write
A12	A19	Address bit 19	B12	SMEMR	System Memory Read
A13	A18	Address bit 18	B13	IOW	I/O Write
A14	A17	Address bit 17	B14	IOR	I/O Read
A15	A16	Address bit 16	B15	DACK3	DMA Acknowledge 3
A16	A15	Address bit 15	B16	DRQ3	DMA Request 3
A17	A14	Address bit 14	B17	DACK1	DMA Acknowledge 1
A18	A13	Address bit 13	B18	DRQ1	DMA Request 1
A19	A12	Address bit 12	B19	REFRESH	Refresh
A20	A11	Address bit 11	B20	CLOCK	System Clock
A21	A10	Address bit 10	B21	IRQ7	Interrupt Request
A22	A9	Address bit 9	B22	IRQ6	Interrupt Request
A23	A8	Address bit 8	B23	IRQ5	Interrupt Request
A24	A7	Address bit 7	B24	IRQ4	Interrupt Request
A25	A6	Address bit 6	B25	IRQ3	Interrupt Request
A26	A5	Address bit 5	B26	DACK2	DMA Acknowledge 2
A27	A4	Address bit 4	B27	T/C	Terminal count
A28	A3	Address bit 3	B28	ALE	Address Latch Enable
A29	A2	Address bit 2	B29	+5V	+5 VDC
A30	A1	Address bit 1	B30	OSC	High-speed Clock
A31	A0	Address bit 0	B31	GND	Ground
C1	SBHE	Sys. Bus High Enable	D1	MEMCS16	Mem. 16bit chip selec
C2	LA23	Address bit 23	D2	IOCS16	I/O 16-bit chip select
C3	LA22	Address bit 22	D3	IRQ10	Interrupt Request 10
C4	LA21	Address bit 21	D4	IRQ11	Interrupt Request 11
C5	LA20	Address bit 20	D5	IRQ12	Interrupt Request 12
C6	LA18	Address bit 19	D6	IRQ15	Interrupt Request 15
C7	LA17	Address bit 18	D7	IRQ14	Interrupt Request 14
C8	LA16	Address bit 17	D8	DACK0	DMA Acknowledge 0
C9	MEMR	Memory Read	D9	DRQ0	DMA Request 0
C10	MEMW	Memory Write	D10	DACK5	DMA Acknowledge 5
C11	SD08	Data bit 8	D11	DRQ5	DMA Request 5
C12	SD09	Data bit 9	D12	DACK6	DMA Acknowledge 6
C13	SD10	Data bit 10	D13	DRQ6	DMA Request 6
C14	SD11	Data bit 11	D14	DACK7	DMA Acknowledge 7
C15	SD12	Data bit 12	D15	DRQ7	DMA Request 7
C16	SD13	Data bit 13	D16	+5 V	
C17	SD14	Data bit 14	D17	MASTER	
C18	SD15	Data bit 15			

Рис.3.1. Системная шина ISA Bus

- LA[23:17] — нефиксированные сигналы адреса, требующие защелкивания по спаду сигнала BALE. Такой способ подачи адреса позволяет сократить задержку и разрешить декодирование схемам дешифратора адреса памяти плат расширения несколько раньше спада BALE;

- IRQ[10:12], IRQ[14:15] — дополнительные запросы прерываний;

- DRQ[5:7] — запросы 16-битных каналов DMA (положительным перепадом);

- DACK[5:7]# — подтверждение запросов 16-битных каналов DMA.

По адресованному ему спаду сигнала чтения устройство должно выдать на шину данных содержимое адресуемой ячейки и удерживать его, пока не произойдет подъем данного сигнала. Во время циклов записи процессор выставляет действительные данные несколько позже начала (спада) сигнала записи, и устройство должно для себя фиксировать эти данные в конце цикла по подъему сигнала записи. Обращение к портам ввода/вывода отличается тем, что сигналы LA[32:17] не используются.

Минимальная длительность цикла определяется чипсетом и может программироваться опциями BIOS Setup через количества тактов ожидания. При этом циклы обращения к памяти обычно короче циклов обращения к портам ввода-вывода.

Рассмотрим принципиальную и структурную схему УСО на основе шины ISA.

УСО на основе шины ISA является параллельным периферийным интерфейсным адаптером (контроллером) портов ввода-вывода, с выбранными адресами 360H-363H.

Объекты контроля в предлагаемом примере объединяются в матрицу (32x32) и представляют собой пары свободных контактов контролируемых реле. Все объекты объединяются в 32 группы по 31 объекту в каждой. Вместо 32 объекта устанавливается перемычка, необходимая для проведения проверок достоверности считывания информации.

На структурной схеме (рис.3.2.) поместим следующие элементы:

- селектор адреса – воспринимает сигналы системной шины ПК: AEN, SA2-SA9 и при наличии на адресной шине комбинации сигналов, соответствующей 360H-363H, выдает сигнал соответствия адреса, разрешающий работу селектора портов;

- селектор портов – воспринимает сигналы системной шины SA0, SA1, IOR, IOW и выдает сигналы управления регистрами и схемой формирования шины данных;

- формирователь шины данных – подключается к двунаправленной системной шине данных SD0-SD7, осуществляет передачу информации в направлениях: от ПК к регистрам, от устройства ввода данных (D0-D7) к ПК (SD0-SD7);



Рис.3.2. Структурная схема УСО на основе интерфейса ISA

- регистр данных – предназначен для фиксации, хранения, и формирования выходных сигналов D10-D17; комбинация D10-D17 является одним из альтернативных вариантов подачи информации на входы D0-D7;

- регистр и формирователь сигнала адреса используется для фиксации хранения информации, полученной по системной шине данных SD0-SD7;

- регистр управляющего сигнала выдает управляющий сигнал WR на вход устройства выбора группы объектов.

Устройство выбора группы объектов коммутирует выходные цепи таким образом, чтобы одну (и только одну) из групп внешних объектов подключить к прямому проводу контрольной батареи. Устройство выбора группы объектов включает в себя следующие схемы: шинный формирователь, селектор адреса, входной формирователь, формирователь управляющих сигналов, схему защиты от ложных срабатываний, регистры памяти.

Устройство ввода данных предназначено для считывания информации с внешних объектов передачи побайтно в системную шину данных ПК. Оно состоит из входных нормализаторов, гальванической оптоэлектронной развязки, входных и выходных цепей, усилителей, схемы коммутации выходных сигналов, накопителей, шинного формирователя.

Системный интерфейс и контроллеры периферийного оборудования ПК обеспечивают легкость подключения и отключения периферийных устройств. Специальные средства доступа к шине исключают конфликтные ситуации между двумя и более одновременно работающими передатчиками. Управление работой приемо-передатчиков осуществляется во время цикла чтения шины таким образом, что информация передается от исполнителя только в нужном направлении. Это исключает возможность потери данных, обусловленную импульсными помехами от источника питания в моменты переключения приемо-передатчиков шины. Системные интерфейсы ПК совершенствуются в различных направлениях, но при этом обеспечивается обратная совместимость с предшествующими версиями интерфейсов. Это касается не только рассматриваемой системной шины ISA, но и более новой системной шины PCI.

Выполнение циклов чтения и записи в периферийные порты осуществляется аналогично циклам обмена с памятью. Сначала выставляется адрес и управляющие сигналы, затем происходит сама запись. Сигналы системной шины, используемые рассматриваемым УСО, приведены в табл.3.1.

SA0-SA9(System_Address_Bus) – системная шина адреса. Используется микропроцессором для указания адресов периферийных портов, с которыми производится обмен данными.

Во время обращения к порту ввода/вывода шина адреса возбуждается в течение действия строба адреса BALE и должна быть зафиксирована селектором адреса YCO. Высокий уровень сигнала BALE указывает, что на шине сформировался адрес.

AEN(Address_Enable) – сигнал разрешения адреса прямого доступа к памяти. Активным является низкий уровень сигнала AEN. Активизация сигнала AEN свидетельствует о пересылке данных в циклах прямого доступа к памяти.

Таблица 3.1

Сигналы системной шины, используемые YCBO

Контакт	Обозначение	Назначение
B1, B31	GND	Общий, земля
B3, B29	+5V	Напряжение питания
A2	SD7	Линия данных 7
A3	SD6	Линия данных 6
A4	SD5	Линия данных 5
A5	SD4	Линия данных 4
A6	SD3	Линия данных 3
A7	SD2	Линия данных 2
A8	SD1	Линия данных 1
A9	SD0	Линия данных 0
A11	AEN	Возможность адресации
B13	IOW	Запись в порт ввода/вывода
B14	IOR	Чтение из порта ввода-вывода
A22	SA9	Адресная линия 9
A23	SA8	Адресная линия 8
A24	SA7	Адресная линия 7
A25	SA6	Адресная линия 6
A26	SA5	Адресная линия 5
A27	SA4	Адресная линия 4
A28	SA3	Адресная линия 3
A29	SA2	Адресная линия 2
A30	SA1	Адресная линия 1
A31	SA0	Адресная линия 0

IOW(Input/Output_Write) – сигнал записи в порт ввода/вывода. Активным является низкий уровень сигнала IOW. Принимается селектором порта YCBO.

SD0-SD7(System_Data_Bus) – системная шина данных. Она предназначена для пересылки байтов информации. Двухнаправленная передача байтов может производиться как в микропроцессор, так и из него. Подключается к схеме формирования шины данных.

Назначенные порты ввода-вывода имеют адреса:

- 360H: адрес порта вывода данных;
- 361H: адрес порта выбора объекта;
- 362H: адрес порта ввода данных;
- 363H: адрес порта вывода управляющего сигнала.

Ниже приведена принципиальная электрическая схема [6] УСО на основе шины ISA (рис.3.3.)

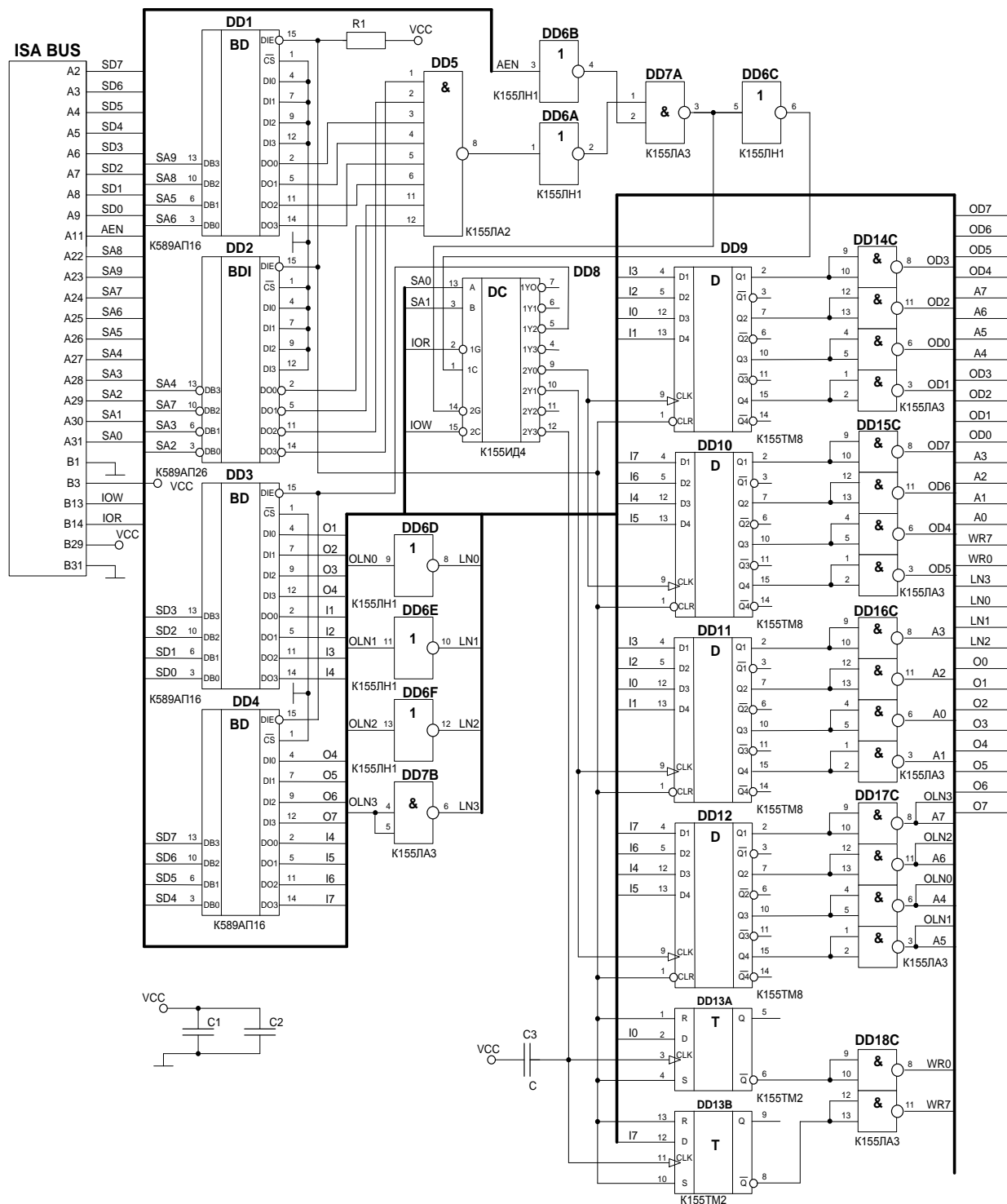


Рис.3.3. Принципиальная схема УСО на шине ISA

Принципиальная схема получена путем объединения функциональных узлов структурной схемы. Кроме того, ее нельзя построить без подбора определенной элементной базы. Рассмотрим основные компоненты представленного устройства.

На микросхемах DD1, DD2, DD5, DD6 и DD7 выполнен селектор адреса (рис.3.4.), задача которого заключается в фиксации моментов, когда микропроцессор обращается к портам ввода-вывода по адресам 360H-363H. Он состоит из следующих микросхем:

- DD1, DD2 – шинные формирователи (DD1 – без инверсии, DD2 – с инверсией), настроенные на передачу информации только в одном направлении (DB→D0);

- DD5, DD6, DD7 – логические элементы, осуществляющие формирование разрешающих сигналов при обращении микропроцессора к портам с адресами 360H-363H и запрещающих сигналов при обращении микропроцессора к устройствам с другими адресами. Для каждого устройства могут быть назначены свои адреса портов, при этом во время разработки селектора адреса необходимо учитывать базовый адрес и диапазон портов ввода-вывода. Задача селектора адреса – выдавать управляющий сигнал только в пределах выбранного диапазона. Так, в приведенной схеме диапазон составляет 4 адреса, начиная с базового -- 360H. Это означает, что комбинация на шине адреса для заданного УСО представляет собой последовательность двоичных цифр следующего вида: 11011000XX. Младшие разряды (XX) для селектора значения не имеют, так как они определяют допустимый внутренний диапазон УСО. Эти разряды включены в схему дешифрации функций портов (DD8). И поскольку устройство рассчитано на десятибитный адрес, то все остальные (старшие) разряды включены в логическую схему селектора адреса.

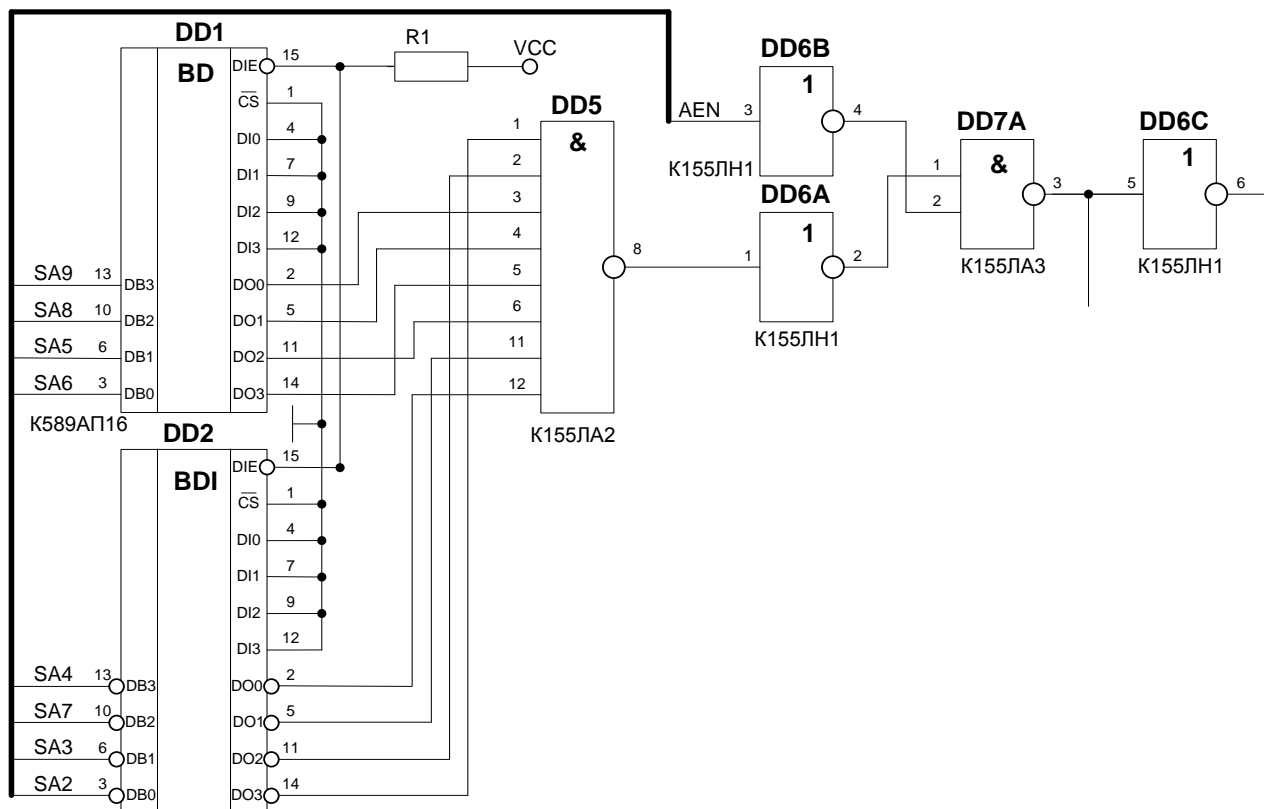


Рис.3.4. Принципиальная схема селектора адреса

На микросхеме DD8 выполнен селектор портов (рис.3.5.), который определяет, к какому из номеров портов, входящих в УСО, обращается микропроцессор в данный момент и формирует управляющие сигналы для переключения элементов УСО, настраивая их на выполнение заданной функции (при получении сигнала разрешения с селектора адреса).

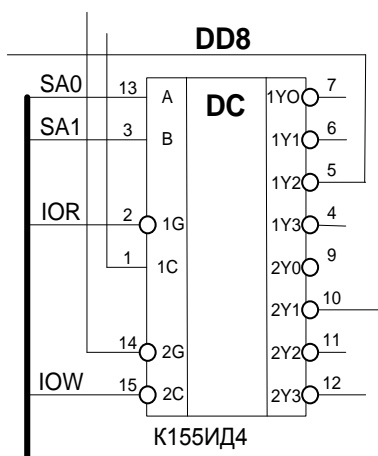


Рис.3.5. Принципиальная схема селектора портов

DD3, DD4 – формирователи шины данных, обеспечивающие развязку входов и выходов устройств, работающих на двунаправленную магистраль. Они повышают нагрузочную способность шины, защищают электрические цепи системного интерфейса. При обра-

щении к портам с адресами 360H, 361H, 363H формирователи шины данных (их еще называют канальными приемо-передатчиками) переключаются на передачу информации в направлении DB→DO, при адресе 362H – в направлении DI→DB. Принципиальная схема шинных формирователей приведена на рис.3.6.

DD9, DD10 – регистр данных, служит для фиксации, хранения и выдачи информации, полученной из шины данных системной магистрали, на выходы D10-D17. Принципиальная схема регистра данных приведена на рис.3.7. Управляющие сигналы поступают в цикле «запись», когда на выходе микросхемы DD8 формируется стробирующий сигнал. Следует отметить, что в рассматриваемом варианте схемы используются динамические D-триггеры. Для них достаточно перепада уровня сигнала, чтобы триггер зафиксировал и защелкнул информацию. Однако в случае применения статических регистров необходимо позаботиться об обеспечении удержания защелкнутой в триггерах информации. Это еще раз доказывает, что для того чтобы разрабатывать работоспособные устройства, необходимо хорошо представлять элементную базу и знать основные схемотехнические решения с выбранным набором элементов.

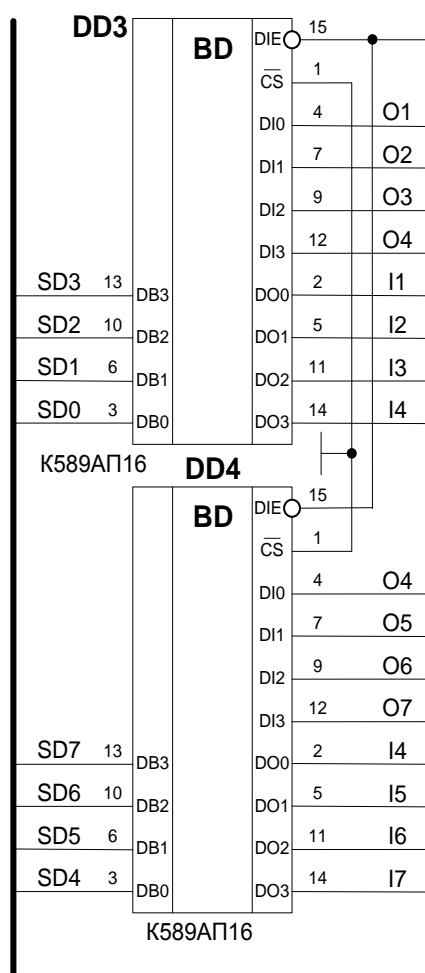


Рис.3.6. Принципиальная схема формирователей шины данных

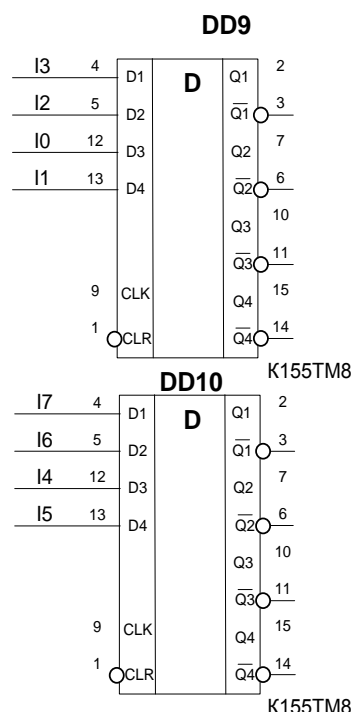


Рис.3.7. Принципиальная схема регистра данных

DD13 – регистр управляющего сигнала. Используется для выдачи записи на выход WR аналогично регистру данных, но выводит (записывает) информацию при адресе порта 363H. Этот сигнал используется для регистрации данных, предназначенных определенному исполнительному устройству. При этом предполагается, что данные для устройства выставлены на шине данных и к этому моменту зафиксированы регистром данных. Таким образом, можно реализовать многоступенчатую передачу информации, когда по одному адресу порта данных передается информация для множества устройств. Принципиальная схема регистра управляющего сигнала показана на рис.3.8.

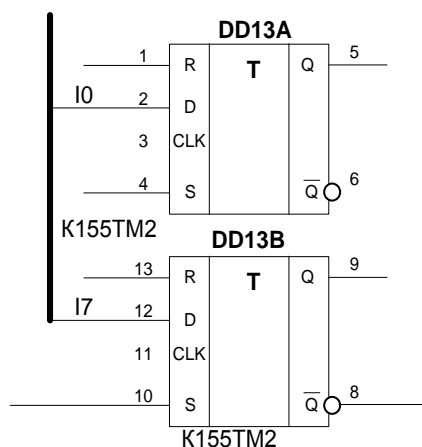


Рис.3.8. Принципиальная схема регистра управляющего сигнала

Микросхемы DD11, DD12 используются аналогично микросхемам DD9, DD10, но для вывода информации - на адресные выходы для устройств, подключаемых к УСО. При этом, старшие разряды дополнительно инвертируются и подаются на выход устройства. Это дает дополнительную гибкость при программировании управления большим количеством объектов без выделения дополнительных портов ввода-вывода. Установленные элементы DD14-DD18 обеспечивают повышенную нагрузочную способность УСО, что повышает его надежность и жизнеспособность. В общем виде порядок взаимодействия процессора со всеми узлами устройства иллюстрирует временная диаграмма (рис.3.9.).

Следует помнить, что микропроцессорная система требует определенной последовательности выполнения операций ввода-вывода, так как управляющие сигналы для каждого типа интерфейса имеют свой порядок и продолжительность. Разработка устройств на основе интерфейсов ISA и других системных шин должна включать программное обеспечение (как правило, написанное на языках низкого уровня), которое и будет регламентировать последовательность выполнения операций записи в порт и чтения с приемом данных и размещением их в памяти компьютера для дальнейшей обработки.

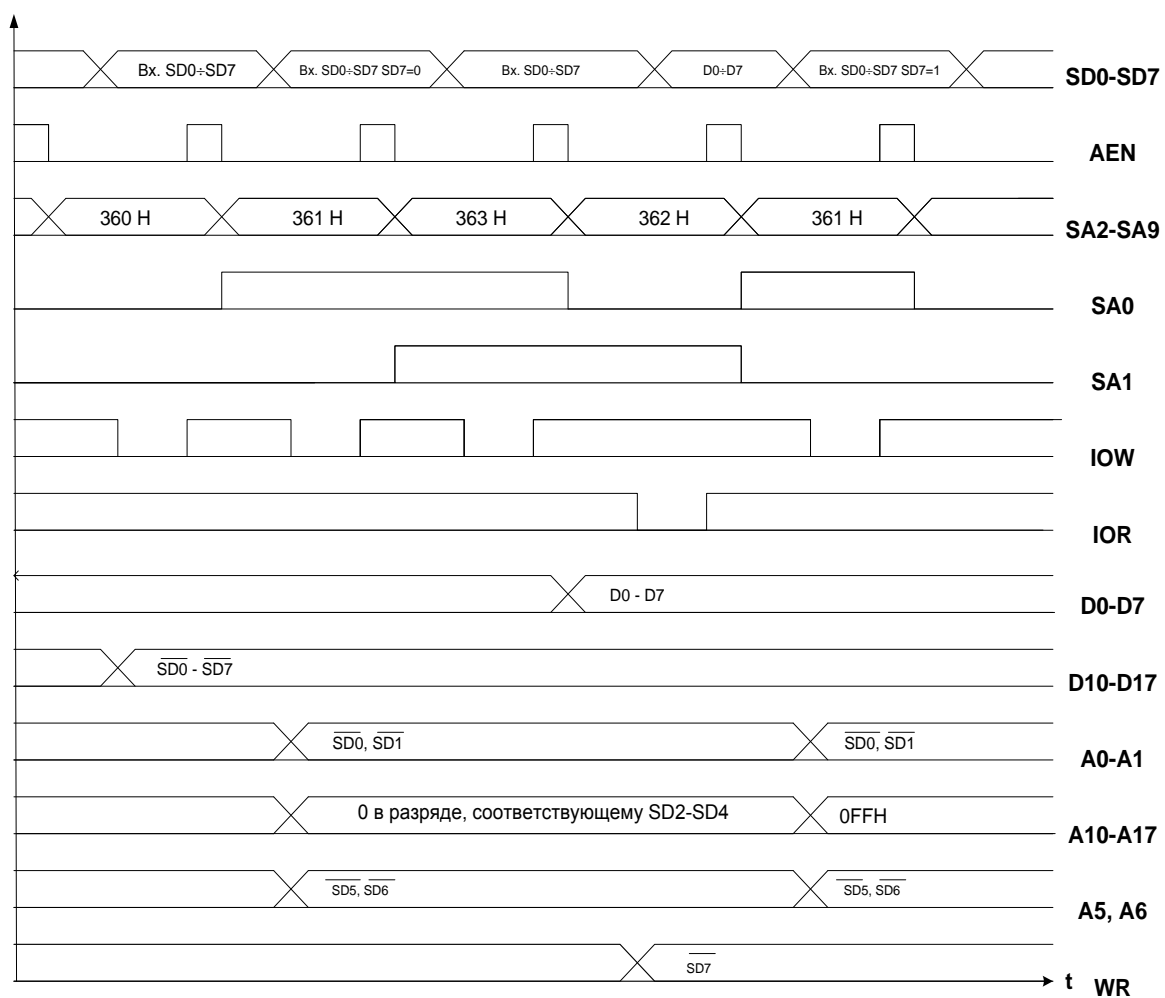


Рис.3.9. Временная диаграмма работы УСО на интерфейсе ISA

4. СОЗДАНИЕ УСО С ПАРАЛЛЕЛЬНЫМ ИНТЕРФЕЙСОМ

Создание интерфейса с параллельным портом значительно проще, чем с системным интерфейсом. Это обусловлено тем, что внутренняя организация LPT порта уже согласована с системной магистралью, выделены адреса ввода-вывода и данные уже хранятся в регистрах. Однако для построения УСО требуются определенные знания об этом интерфейсе, об элементной базе и навыки программирования портов. Как уже отмечалось, в адресном пространстве базовые адреса LPT портов 278H, 378H, 3BCH. Кроме того, для порта LPT1 предусмотрено аппаратное прерывание IRQ7, а для LPT2 – IRQ5. Как известно, параллельный порт имеет три регистра, один из которых является регистром данных, другой – регистр статуса и третий – регистр состояния.

Рассмотрим устройство, которое должно обеспечивать индикацию выносного табло, состоящего не менее чем из 350 цепей ламп, и контроль не менее 150 двухпозиционных объектов (кнопок или пар контактов). На основании этого и исходя из концептуальной структуры УСО, разработаем общую функциональную схему устройства (рис.4.1.). Интерфейс формирует сигналы данных, несущих информацию о состоянии матрицы кнопок, и принимает управляющие сигналы и информацию канала шины данных. Задача устройства сопряжения сводится к включению цепей индикации, определяемых сигналами на шине данных компьютера, и к контролю состояния кнопок с передачей этой информации на шину данных компьютера.

Функционально УС состоит из следующих основных компонентов:

- схемы интерфейса;
- схем опроса кнопок;
- схемы управления индикацией;
- блока питания 5V для питания интерфейсных схем;
- источника питания 24V для обеспечения индикации на табло;
- схемы защиты по питанию.

Функциональная схема работы устройства сопряжения компьютера через стандартный периферийный параллельный порт обеспечивает передачу данных на устройство сопряжения по трем шинам:

1. Шина данных (ШД)- 8-разрядная однонаправленная (от ПК к УСО);
2. Шина управления (ШУ) - 4-разрядная однонаправленная (от ПК к УСО);
3. Канал входной информации (КВИ) - 5-разрядный однонаправленный (от УСО к ПК).

Сигналы по шинам данных поступают на интерфейсную плату, которая обеспечивает взаимодействие с платами управления индикацией и платами опроса кнопок.

Для построения принципиальной схемы необходимо подобрать элементную базу и сформировать основные узлы устройства [1, 12, 13].

Мы видим, что в состав устройства входят, как и в случае интерфейса ISA, шинные формирователи, регистры, дешифраторы.

Полное описание этих микросхем можно получить из соответствующих справочников [2, 7]. Мы познакомимся только с некоторыми из них и рассмотрим способы их подключения.

Микросхема K555АП6 - шинный формирователь, предназначенный для обеспечения необходимого уровня мощности сигналов. Он содержит восемь цепей с тремя состояниями выходов: два входа разрешения E_{AB} (переключение направления каналов) и E_0 (перевод выхода канала в разомкнутое состояние). У микросхемы нет недопустимых сигналов управления. Она потребляет ток 90 мА, когда на всех входах присутствует напряжение низкого уровня, и 95 мА, если выходы каналов переведены в разомкнутое состояние. Время задержки выключения в разомкнутое состояние составляет 30...40 нс. Внешний вид и обозначения контактов приведены на рис.4.2.

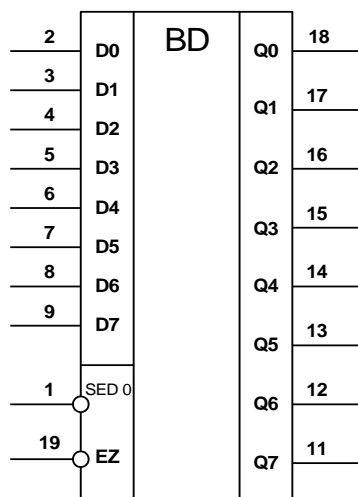


Рис.4.2. Микросхема K555АП6

Микросхема K155ИР13 - регистр данных однонаправленный (A→Q) универсальный, восьмиразрядный, синхронный регистр сдвига. Каждая операция продолжается в регистре не более 20 нс, поэтому он пригоден для обслуживания скоростных процессоров и в качестве запоминающего устройства как буферный накопитель байта.

Синхронную работу регистру обеспечивают специальные входы выбора режимов S0 и S1. Состоянием входов S0 и S1 определяется также прием тактового перепада от входа C. На входы S0 и S1 перепад от высокого уровня к низкому можно подавать, когда на входе C присутствует напряжение высокого уровня. При параллельной загрузке слово, подготовленное на входах D0-D7, появится на выходах Q0 - Q7 после прихода последующего положительного перепада тактового импульса. Сброс у регистра K155ИР13 - асинхронный; при подаче на вход R активного напряжения низкого уровня на выходах Q0-Q7 фиксируются низкие уровни. Регистр потребляет ток 116 мА, тактовая частота его может превышать 25 МГц. Внешний вид и обозначения контактов приведены на рис.4.3.

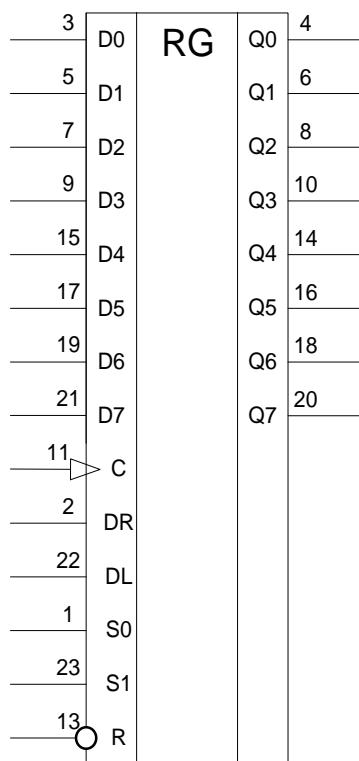


Рис.4.3. Микросхема K155ИР13

Микросхема K555ИР22 - восьмиразрядный регистр - защелка отображения данных. Регистр принимает и отображает информацию синхронно с положительным перепадом на тактовом входе. Регистр K555ИР22 потребляет ток 40 мА. Время задержки распространения сигнала для цепи вход – выход составляет 40 нс.

Если имеется несколько выходов, то суммарный ток всех выходов не должен превышать паспортного значения для данной микросхемы, даже если отдельные выходы микросхемы будут недогружены. Внешний вид и обозначения контактов приведены на рис.4.4.

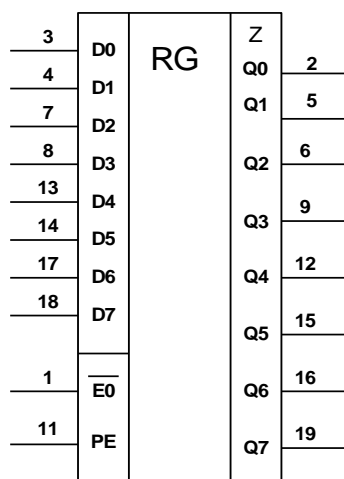


Рис.4.4. Микросхема K555IP22

Микросхема K155ИДЗ - дешифратор, позволяющий преобразовать четырехразрядный код, поступивший на входы АО-АЗ в напряжение низкого логического уровня, появляющееся на одном из шестнадцати выходов 0-15. Дешифратор имеет два выхода разрешения дешифрации: ЕО и Е1.

По входам Е0 и Е1 подаются сигналы разрешения выходов, чтобы устранять текущие выбросы, которыми сопровождается дешифрация кодов, появляющихся не строго синхронно (например, поступающих от счетчика пульсации). Чтобы разрешить прохождение данных на выходы, на входы Е0 и Е1 следует дать напряжение низкого уровня. Когда на входах Е0 и Е1 присутствуют напряжения высокого уровня, то на выходах 0-15 появляются также высокие уровни.

Дешифратор K155ИДЗ потребляет ток 56 мА. Время задержки распространения сигнала для цепи «вход А – выход» составляет 36 нс. Внешний вид и обозначения контактов приведены на рис.4.5.

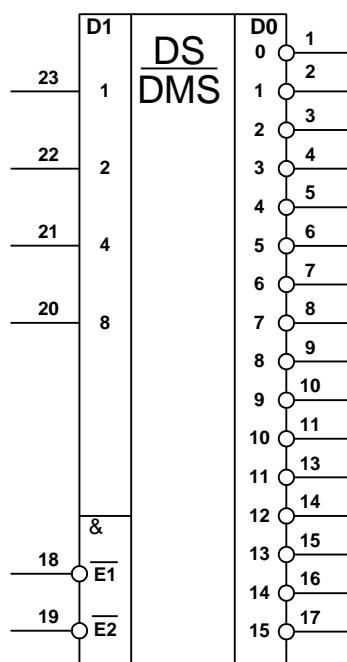


Рис.4.5. Микросхема K555ИД3

Микросхема K155ЛН1 содержит инверторы, снабженные двухтактным выходным каскадом. Время задержки распространения сигнала составляет 22 нс, соответствующий стекающий выходной ток для одного инвертора - 16 мА. Наибольший ток микросхемы ЛН1 потребляют тогда, если на всех шести входах присутствуют напряжения высокого уровня. Внешний вид и обозначения контактов приведены на рис.4.6.

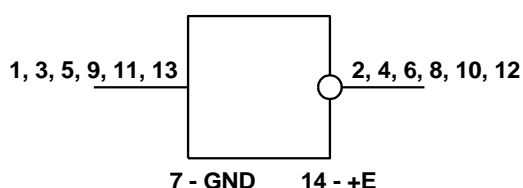


Рис.4.6. Микросхема K155ЛН1

Программируемый параллельный интерфейс K580ВВ55 является многофункциональным устройством, объединяющим в себе все необходимые компоненты для построения параллельных адаптеров внешних устройств. Функциональная схема устройства приведена на рис.4.7. Это устройство программируемое. Внешний вид и расположение контактов представлено на рис.4.8. Назначение выводов микросхемы – на рис.4.9, а временная диаграмма работы в режиме 0 – на рис.4.10.

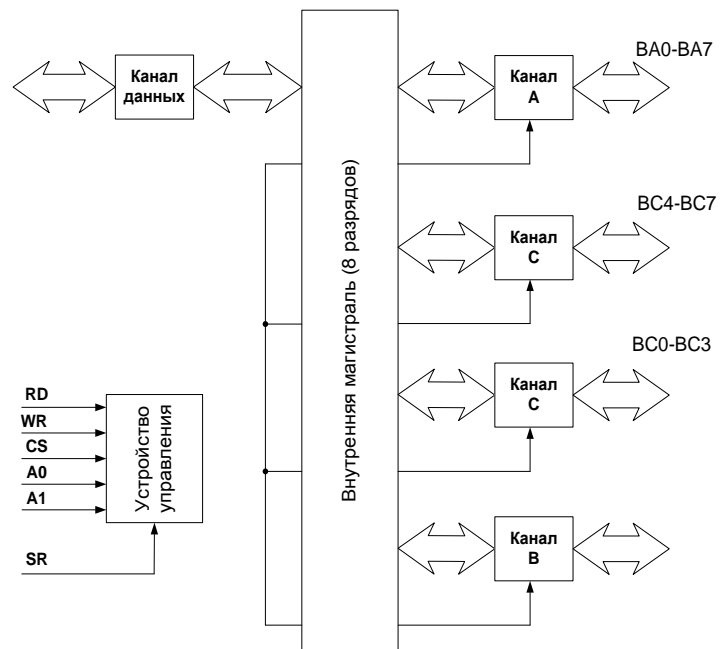


Рис.4.7. Функциональная схема K580BB55

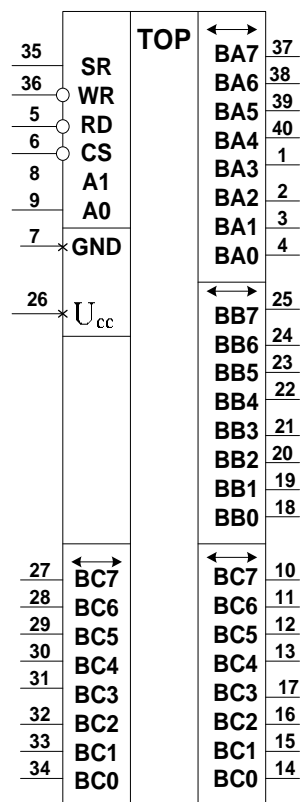


Рис.4.8. Микросхема K580BB55

Вывод	Обозначение	Функциональное назначение
1-4; 37-40	BA7-BA0	Информационный канал А
5	\overline{RD}	Чтение информации
6	\overline{CS}	Выбор микросхемы
7	GND	Общий
8-9	A1, A0	Младшие разряды адреса
10-17	BC7-BC0	Информационный канал С
18-25	BB7-BB0	Информационный канал В
26	Ucc	Питание 5В
27-34	D7-D0	Канал данных
35	SR	Установка в исходное состояние
36	\overline{WR}	Запись информации

Рис.4.9. Назначение выводов микросхемы K580BB55

Микросхема K580BB55 содержит три 8-разрядных канала ввода-вывода А, В, С.

Канал С может быть представлен в виде двух 4-разрядных каналов ввода/вывода, доступ к которым производится как к отдельным независимым каналам. Периферийные устройства подключаются к каналам А, В, С, а связь с микропроцессором осуществляется с помощью шины О через буфер данных. Каждый из каналов А, В, С состоит из 8-разрядного регистра и двунаправленных формирователей, имеющих на выходе состояние «Выключено». Устройство управления содержит регистр управляющего слова (РУС), в который предварительно производится запись информации, определяющей режим работы каналов, и формирует сигналы выбора канала и управления каналом С.

Микросхема может работать в одном из трех режимов: режим О - простой ввод/вывод; режим 1 - стробируемый ввод/вывод; режим 2 - двунаправленный канал. Режим работы каналов можно изменять как в начале, так и в процессе выполнения программы, что позволяет обслуживать различные периферийные устройства в определенном порядке с помощью одной микросхемы K580BB55А. Каналы А и В могут работать в различных режимах, а работа канала С зависит от режимов работы каналов А и В. Комбинируя режимы работы каналов, можно обеспечить работу микросхемы почти с любым периферийным устройством.

В режиме 0 осуществляется простой ввод/вывод данных по трем 8-разрядным каналам, причем канал С может использоваться как два 4-разрядных канала. Каждый из каналов может использоваться отдельно для ввода или вывода информации. В режиме О входная

информация не запоминается, а выходная хранится в выходных регистрах до записи новой информации в канал или до записи нового режима,

В режиме 1 передача данных осуществляется только через каналы А и В, а линии канала С служат для приема и выдачи сигналов управления. Каждый из каналов А и В независимо друг от друга может использоваться для ввода или вывода 8-разрядных данных, причем входные и выходные данные фиксируются в регистрах каналов,

В режиме 2 для канала А обеспечивается возможность обмена информацией с периферийными устройствами по 8-разрядному двунаправленному каналу. Для организации обмена используются пять линий канала С. В режиме 2 входные и выходные данные фиксируются во входном и выходном регистрах соответственно.

Подробное описание работы микросхемы можно получить из специальных справочников [1, 2].

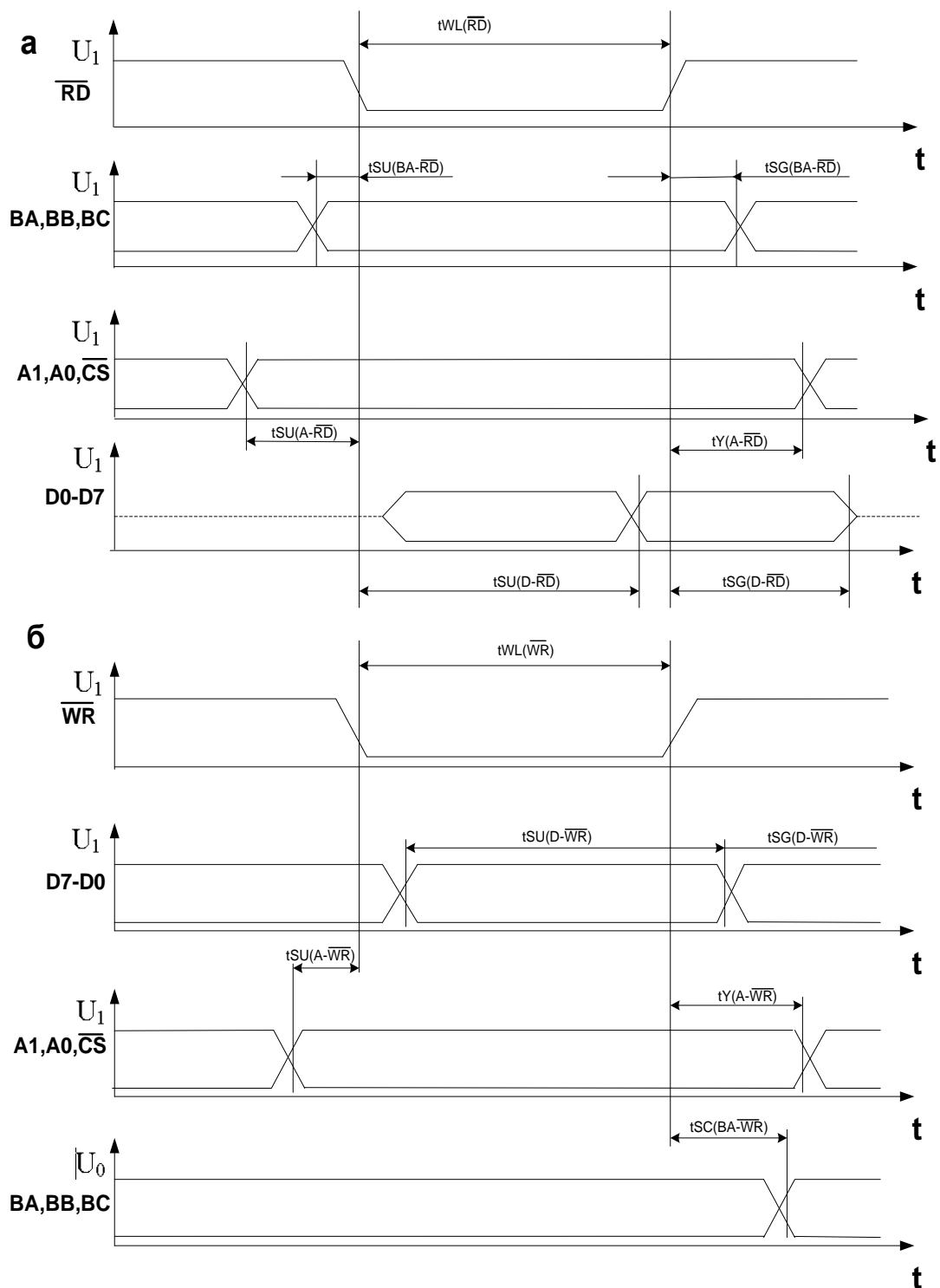


Рис.4.10. Временная диаграмма работы микросхемы K580BB55:
а) при вводе, б) при выводе информации

На рис.4.11. представлен фрагмент принципиальной схемы, построенной с применением описанной элементной базы (за исключением BB55) и на основе функциональной схемы.

С компьютера необходимые данные, сформированные программным путем, посылаются в порт на 8-разрядную шину данных D0-D7. Через интерфейсный кабель, связывающий ПК с управляющей схемой, они попадают на микросхему DD1 – шинный формирователь, выполняющий 2 функции: усиление сигнала; защиту LPT порта от высокого потенциала в случае пробоя одного из транзисторов схемы управления индикацией.

С выхода DD1 усиленный по уровню сигнал подается на микросхему DD9 – регистр данных, куда они записываются по управляющему сигналу (C0) из порта и хранятся до подачи нового управляющего сигнала. С выходов DD9 данные снова поступают на шинные формирователи DD4, где в дальнейшем они поступают на регистр данных выбранной группы устройств (рис.4.12.) Далее данные одновременно, через общую шину, подводятся к входам всех микросхем – регистров данных, расположенных в схемах управления индикацией.

Программа формирует следующий пакет данных, отвечающих за выбор микросхемы на платах управления, в которую будут записаны переданные ранее данные. Снова происходит посылка в порт, и данные через DD1 попадают одновременно на входы регистра данных DD4 (данные в регистр не записываются, так как не подается сигнал записи на управляющий вход С регистра) и входы дешифратора DD16, причем за разрешение дешифрации отвечает комбинация из 3 старших разрядов (2,3,4) и управляющего сигнала С1 из порта, а на информационные входы дешифраторов А0-А3 подаются младшие разряды 5,6,7,8. При получении сигнала разрешения дешифрации на управляющие входы W0 (C1), W1(2, 3, 4) четырехразрядный код, поступивший на информационные входы А0-А3 дешифратора преобразуется в напряжение низкого логического уровня, появляющееся на одном из шестнадцати выходов. Вследствие того, что выходы у дешифратора инверсные, а для управления регистрами на платах индикации требуется напряжение высокого логического уровня, то сигнал с выходов дешифратора подается на инверторы DD14 на базе микросхемы К155ЛН1. Каждый выход инвертора соединен с управляющим входом регистра платы управления индикацией (рис.4.13.), на входе которого уже находятся необходимые данные. При получении управляющего сигнала высокого уровня с инвертора данные записываются в регистр и передаются на его выходы, сохраняя текущее состояние до получения следующего управляющего сигнала.

На выходе регистров плат управления стоят транзисторные ключи, где, в соответствии с полученными данными, происходит их открытие, т.е. происходит коммутация питания на объекты управления (лампы индикации).

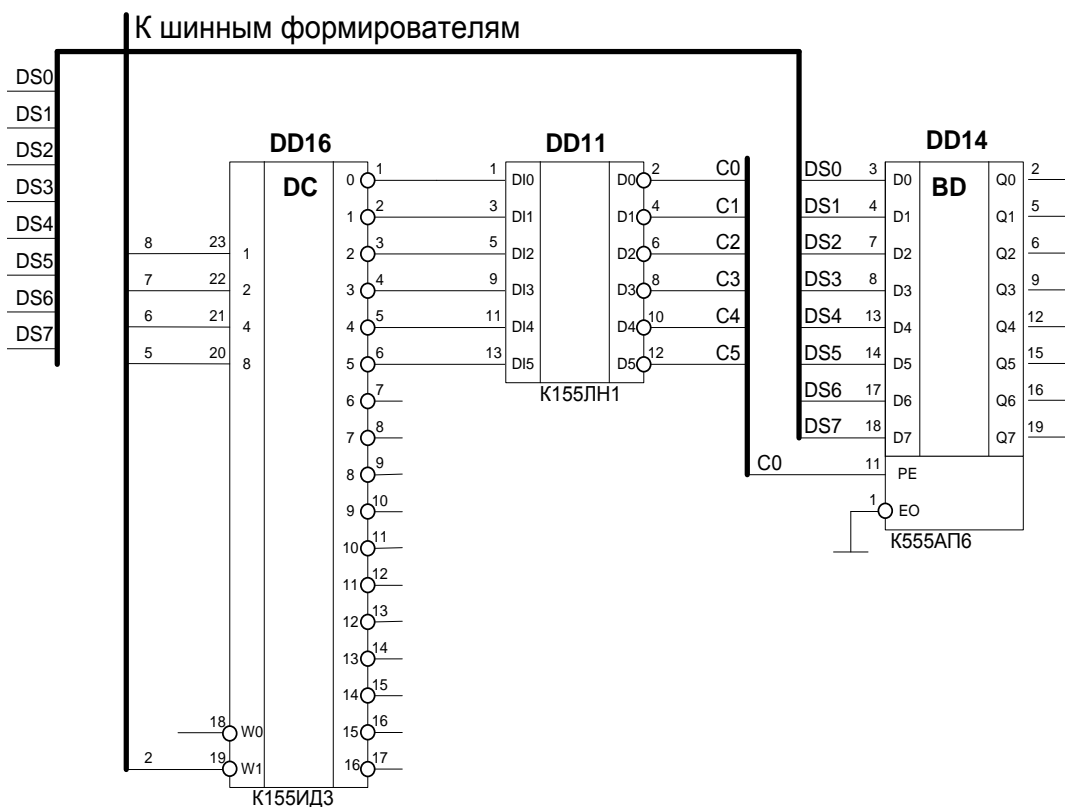


Рис.4.12. Фрагмент схемы дешифрации адреса группы объектов управления

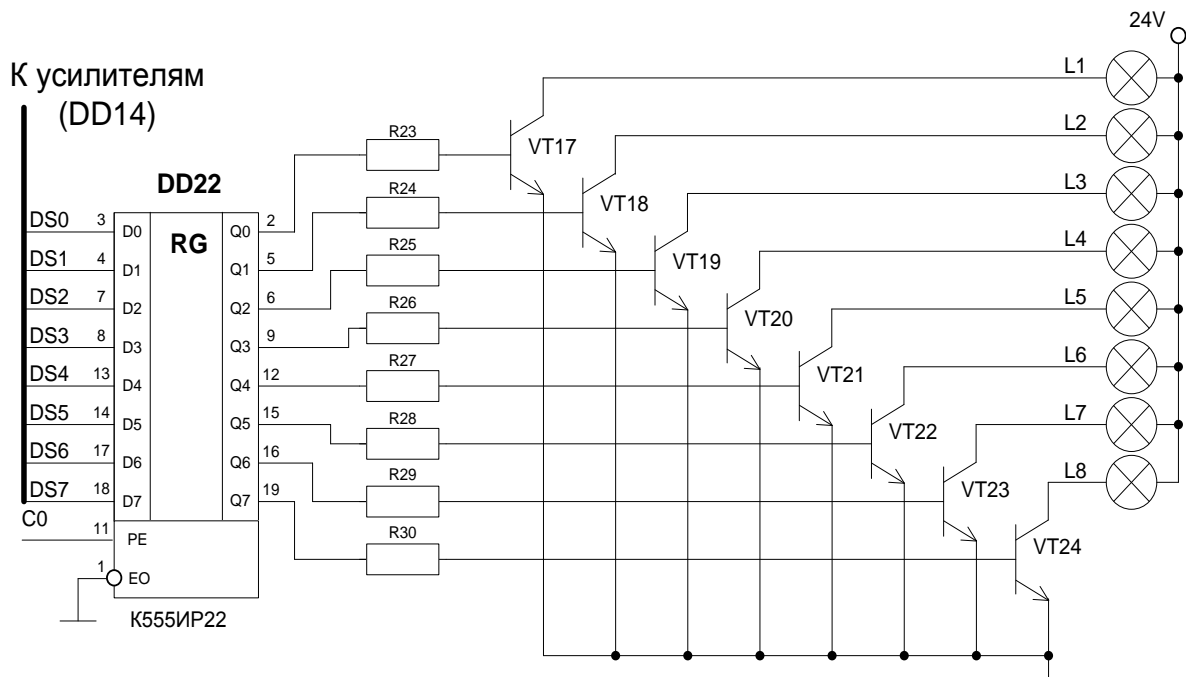


Рис.4.13. Фрагмент схемы управления индикацией

При контроле состояния объектов данные, сформированные компьютером, поступают через порт на DD1 – шинный формирова-

тель, а с него - на DD6 (микросхема K155ИД3) – дешифратор, расположенный на схеме опроса кнопок, причем за разрешение дешифрации отвечает один из 4 старших разрядов (W0,W1,W2,W3, что позволяет подключить еще три дешифратора напрямую, а со схемой дешифрации – еще 16), а на входы дешифратора подаются младшие разряды AK0-AK3. Выходы дешифратора (инверсные) являются вертикальными линейками матрицы опроса. Горизонтальными линейками являются входные линии шинного формирователя DD3. С одной стороны на горизонтальные линейки подается через резисторы сигнал логической 1 (+5В), а с другой стороны линейки через шинный формирователь соединены с портом С компьютера, откуда циклически считываются данные и происходит их программная обработка. В узлах матрицы находятся контакты кнопок (рис.4.14.).

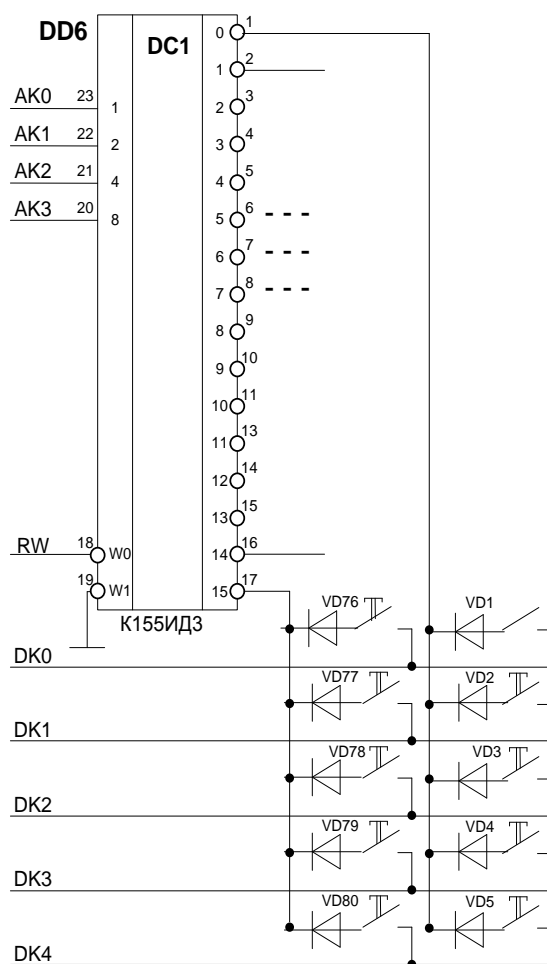


Рис.4.14. Фрагмент схемы контроля объектов

По заданному алгоритму циклически происходит выбор одной из вертикальных линеек путем дешифрации данных, полученных через порт, и, соответственно, на этой линейке появляется логический 0. Если в это время одна (или несколько) из кнопок в узлах этой ли-

нейки нажата, то потенциал логического «0» появляется на горизонтальной линейке этого узла, что вызывает изменение данных, посылаемых шинным формирователем в порт. Путем программной обработки считанных из порта данных и данных, посланных по определенному адресу, однозначно определяется положение нажатой кнопки. Далее формируется блок данных на выбор следующей вертикали, затем он посылается в порт, и процедура повторяется.

Такое устройство может контролировать до 1280 объектов (16x16x5) путем их циклического опроса. Обобщенная временная диаграмма работы УСО (рис.4.15.) показывает двухтактный режим управления объектами (данные – адрес) и режим непосредственного опроса объектов контроля.

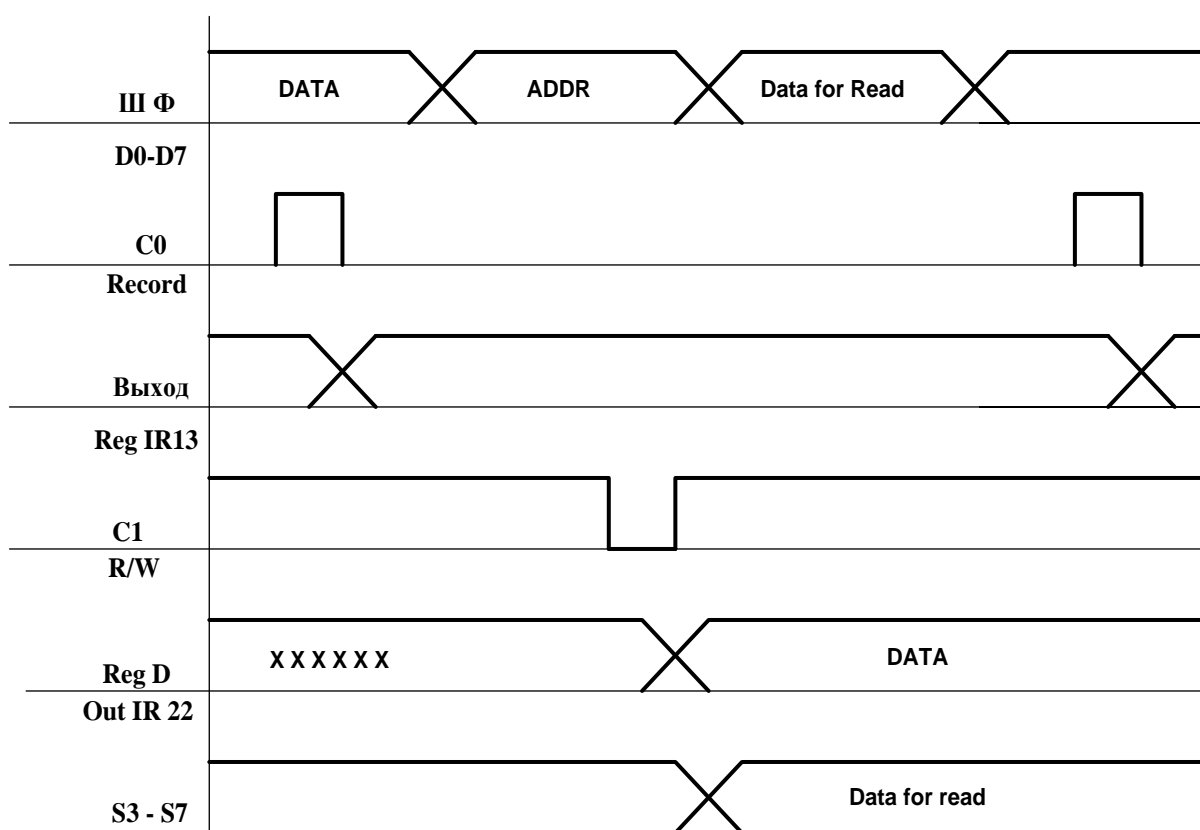


Рис.4.15. Временная диаграмма работы УСО на базе LPT порта

Следует помнить о том, что при большом количестве элементов возможно подключение большого количества входных устройств TTL уровней к одному выходному. Так, например, в рассматриваемом нами устройстве возможно подключение до 256 8-разрядных регистров данных. И это совсем не означает, что шина данных должна быть нагружена всеми этими устройствами [9].

Как уже отмечалось, суммарный ток нагрузки для регистров ограничен независимо от того, сколько выходов задействовано. Для

инверторов ЛН1 стекающий выходной ток для одного инвертора – микросхемы ЛН1 потребляют наибольший ток 16 мА, если на всех шести входах присутствуют напряжения высокого уровня. При входном напряжении 4,5 В эти токи составляют 33, 66 и 54 мА для микросхем ЛН1 серий К155, К555 и К531, соответственно. Если на всех входах присутствуют напряжения низкого уровня, ток потребления снижается в 2,2 раза.

Микросхемы серии К531 потребляют энергию на перенос 1 бита в 2,3 раза меньше микросхемы старой серии К131, вследствие чего она также стала неперспективной. Статические параметры логических элементов микросхем серий К155, К555, К531 и некоторые параметры для серии К1531 сведены в табл.4.1. Логические уровни и допустимые напряжения на входах и выходах микросхем этих серий отличаются незначительно. Однако для микросхем серий К155 и К531 велики входные токи низкого уровня соответственно 1,6 и 2,0 мА для одного входа. Сила этого тока для микросхем серии К555 и К1531 в 3 - 4 раза меньше. Допустимый стекающий выходной ток низкого уровня серии К531 в 2,5 раза больше, чем для серии К555. Кроме того, в составе каждой серии выпускаются так называемые буферные логические элементы, допустимый ток для которых увеличен еще примерно в 3 раза.

Таблица 4.1

Таблица динамических параметров микросхем ТТЛ

Серия ТТЛ	Параметр			Нагрузка	
	$P_{\text{пот}}, \text{мВт}$	$T_{\text{зд.р.}}, \text{нс}$	$\Sigma_{\text{пот}}, \text{пДж}$	$C_{\text{н}}, \text{пФ}$	$R_{\text{н}}, \text{кОм}$
К134	1	33	33	50	4
К155	10	9	90	15	0,4
К131	22	6	132	25	0,28
К555	2	9,5	19	15	2
К531	19	3	57	15	0,28
К1533	1,2	4	4,8	15	2
К1531	4	3	12	15	0,28

Для упрощения расчетов числа нагружающих входов указаны числа взаимной нагрузочной способности микросхем серий К531, К155 и К555 (табл.4.2). Например, обычный элемент серии К555 способен принять ток от четырех входов микросхем серии К531. Рассмотрим другой крайний случай взаимного применения микросхем ТТЛ. Буферный выход (самый мощный среди ТТЛ) микросхемы серии К531 может обеспечить стекание входного тока от 150 логических элементов серии К555. Отметим, что буферный элемент се-

рии К555 имеет более высокую нагрузочную способность, чем простой выход микросхемы серии К531.

Таблица 4.2

Таблица взаимной нагрузочной способности логических элементов ТТЛ разных серий

Нагружаемый выход	Число входов-нагрузок из серии		
	К555	К155	К531
К555	20	5	4
К555, буферная	60	15	12
К155	40	10	8
К155, буферная	60	30	24
К531	50	12	10
К531, буферная	150	37	30

При совместном использовании микросхем ТТЛ высокоскоростных, стандартных и микромощных следует учитывать, что микросхемы серии К531 дают увеличенный уровень помех по шинам питания в силу больших по силе и коротких по времени импульсов сквозного тока короткого замыкания выходных транзисторов логических элементов. Часть печатной платы с микросхемами серии К531 должна иметь отдельные, очень низкоомные шины питания. Токоведущие сигнальные дорожки должны быть кратчайшими, чтобы не излучались помехи.

Серии с повышенными входным и выходным сопротивлениями (например, К555) более чувствительны к помехам-наводкам и к помехам по питанию, чем мощные серии. Маломощную часть устройства требуется защитить экраном и заградительными фильтрами по питанию. Проводники на печатной плате, по которым передаются выходные сигналы микросхем серии К531, не должны проходить рядом с токоведущими дорожками входных сигналов микросхем серии К555. При совместном применении микросхем серии К155 и К555 помехи невелики.

Выходы однокристальных, т.е. расположенных в одном корпусе, логических элементов ТТЛ, можно соединять вместе. При этом надо учитывать, что импульсная помеха от сквозного тока по проводу питания пропорционально возрастет. Реально на печатной плате остаются неиспользованные входы и даже микросхемы (часто их специально «закладывают про запас»). Такие входы логического элемента можно соединять вместе, при этом ток не увеличивается. Однако для элементов серии К555 входы соединять не следует, чтобы не повышалась паразитная входная емкость элемента. На неиспользуемые входы можно подать напряжение высокого уровня

от выхода свободного логического элемента, заземлив при этом его вход.

Как правило, микросхемы ТТЛ с логическими функциями И, ИЛИ потребляют от источников питания меньшие токи, если на всех входах присутствуют напряжения низкого уровня. Из-за этого входы таких неиспользуемых элементов ТТЛ следует заземлять [8].

5. ПРИМЕНЕНИЕ АНАЛОГОВЫХ УСТРОЙСТВ

Часто возникает необходимость работы с аналоговыми устройствами (например, в целях регулирования и измерения; создания цифрового осциллографа с последующим программным спектральным анализом и многое другое), которые могли бы работать под управлением ПК. Для этой цели используются аналого-цифровые (АЦП) и цифро-аналоговые преобразователи (ЦАП) [11].

Цифроаналоговые преобразователи во многих случаях входят составной частью в АЦП. При выборе ЦАП необходимо определить разрядность прибора N , где 2^N — эквивалентно максимальному значению аналогового выходного сигнала. Промышленность выпускает ЦАПы, с разрешением 8-16 разрядов (256 - 65536).

Разрядность не дает определенного ответа на требуемую точность потому, что должны быть приняты во внимание и другие источники ошибки, такие, как:

1. *Интегральная нелинейность*, которая показывает отклонение функции передачи ЦАП от прямой линии — наилучшее приближение к фактической функции передачи между конечными точками. Для ЦАП это отклонение измерено в каждом шаге. В простых приборах интегральная нелинейность достигает 1 младшего значащего разряда (МЗР).

2. *Дифференциальная нелинейность* — различие между фактической высотой шага и идеальным значением 1МЗР. Заданная величина для дифференциальной нелинейности ($\leq 1\text{МЗР}$) гарантирует, что характеристика ЦАП является монотонной. Это означает, что никакие данные не потеряны, поскольку выходной сигнал всегда изменяется в соответствии с цифровым кодом на входе.

3. *Ошибка смещения* - отклонение фактического выходного значения от идеальной величины. Для ЦАП ошибка смещения равна выходному напряжению, когда цифровой код на входе нулевой. Эта ошибка остается константой для всех входных значений, и она может быть скомпенсирована при калибровке схемы. Ошибка смещения часто определяется как абсолютная величина в милливольт, а не МЗР. Приемлемая ошибка смещения — обычно меньше, чем $\pm 10\text{ мВ}$.

4. *Ошибка от коэффициента передачи* определяется как разность между идеальным выходным напряжением и фактическим

максимальным значением функции передачи после вычитания ошибки смещения. Так как ошибка от коэффициента передачи изменяет наклон всей функции передачи, то относительная ошибка будет на каждом шаге преобразования. Эта ошибка может быть выражена в единицах младшего значащего разряда или в милливольт-тах, а также в процентах от максимальной величины.

Идеальный ЦАП должен мгновенно выдать аналоговый сигнал при подаче на вход цифрового кода. Реально аналоговый сигнал на выходе появляется через какое-то время установки ($T_{уст}$), которое состоит из времени внутренней задержки ($t_{зад}$) распространения и ограниченной скорости нарастания выходного напряжения du/dt в выходном усилителе. $T_{уст}$ начинается с преобразования и заканчивается, когда на выходе ЦАП установится стабильное значение аналогового напряжения, включая любую статическую ошибку.

Характеристики ЦАП в большей степени определяются источником опорного напряжения, который может быть встроен в корпус преобразователя или применяться как внешний элемент. Если на выходе аналоговый сигнал не усиливается, то максимальный входной код соответствует $V_{оп}$. Опорное напряжение также определяет напряжение шага, то есть изменение выхода в ответ на 1 переход младшего значащего разряда на входе. Один шаг равен $V_{оп}/2^N$, где N - разрядность ЦАПа.

Преобразование двоичного входа кода в аналоговый вид может быть представлено двумя способами: током, который пропорционален значениям кода, или в виде напряжения. Большинство современных ЦАП имеют более простой выход - по напряжению.

Аналогово-цифровой преобразователь (АЦП) — электронная схема, которая измеряет аналоговые сигналы и преобразовывает их цифровую форму. Аналоговый электрический сигнал на входе преобразователя сравнивается с известным эталонным напряжением и производится цифровое представление этого сигнала. На выходе АЦП имеет обычно двоичный код, пропорциональный входному аналоговому значению. По своей природе АЦП вносит ошибку квантования. Это потерянная информация, поскольку для непрерывного аналогового сигнала должна быть бесконечная разрешающая способность преобразователя, а реально АЦП имеет конечное число разрядов кодирования. Чем выше разрядность АЦП, тем больше разрешающая способность, тем меньше приходится информации на ошибку квантования.

Точность АЦП зависит от нескольких ключевых условий, которые включают: ошибку интегральной нелинейности, ошибку от коэффициента передачи, стабильность опорного напряжения, температурный коэффициент, характеристики прибора по переменному току.

Дифференциальная нелинейность показывает, как изменение во входном аналоговом сигнале преобразовывается в единицу знача-

щего младшего разряда. Другими словами, на какую величину изменится аналоговый сигнал при очередном изменении выходного кода на 1МЗР. Нормально работающий АЦП - это значит нет никакого отсутствия или пропуска кода при подаче аналогового сигнала во всем диапазоне входного напряжения.

Интегральная нелинейность является составной частью дифференциальной нелинейности. Интегральная нелинейность определяется как интеграл ошибок дифференциальной нелинейности. Ошибка интегральной нелинейности показывает, как далеко от идеальной функции происходит передача результата преобразования.

Ошибка коэффициента передачи может включать в себя и ошибку от источника опорного напряжения (ИОН) АЦП. Как правило, ошибка коэффициента передачи намного хуже, если используется ИОН, встроенный в АЦП.

Как от ошибки смещения, так и от ошибки коэффициента передачи теряется динамический диапазон АЦП. Например, если входное напряжение составляет 4050 дискрет вместо идеальных 4096 (для 12-разрядного преобразователя), это определяется как отрицательная ошибка коэффициента передачи. В этом случае 46 дискрет не будут использованы.

Шум напряжения часто определяется как среднеквадратическая величина или как величина полного размаха. Если опорное напряжение 2,5В имеет полный размах шума величиной 500 мкВ, то это представляет ошибку 0,02%, что соответствует только 12-разрядному преобразованию.

Представленная информация относится к АЦП конвейерного типа, куда входят преобразователи на основе регистра последовательного приближения. Эти самые популярные приборы с разрешающей способностью от 8 до 16 разрядов имеют скорость выборки от единиц до сотни мегавыборок в секунду. Более точным приборам соответствует меньшая скорость дискретизации.

Если необходимо сверхбыстрое преобразование аналогового сигнала, то применяют АЦП с параллельным преобразованием. В основном это 8-разрядные преобразователи с частотой дискретизации 1 ГГц — 1,5 ГГц.

Интегрирующие преобразователи имеют высокую разрешающую способность (до 18 разрядов), но скорость преобразования от нескольких сотен Гц до нескольких кГц. Они предназначены для преобразования медленно меняющихся сигналов.

Отслеживающие АЦП используются в системах автоматического регулирования. При разрешающей способности 12-16 разрядов имеют полосу частот до 1 МГц, а 24-разрядный АЦП имеет скорость преобразования от сотен Гц до нескольких кГц. Этот тип преобразователей имеет самую высокую разрешающую способность. На рис.5.1. представлены отличительные характеристики АЦП с раз-

ными типами архитектуры. По этим кривым можно сделать первичный выбор типа прибора для конкретного применения. На сегодняшний день имеется большой выбор отечественных и зарубежных микросхем, а также законченных устройств АЦП и ЦАП. Многие из них имеют встроенные интерфейсы LPT, COM или USB. Выпускаются также устройства, которые подключаются к портам системного интерфейса. Информацию о них можно получить из специальной справочной литературы [2].

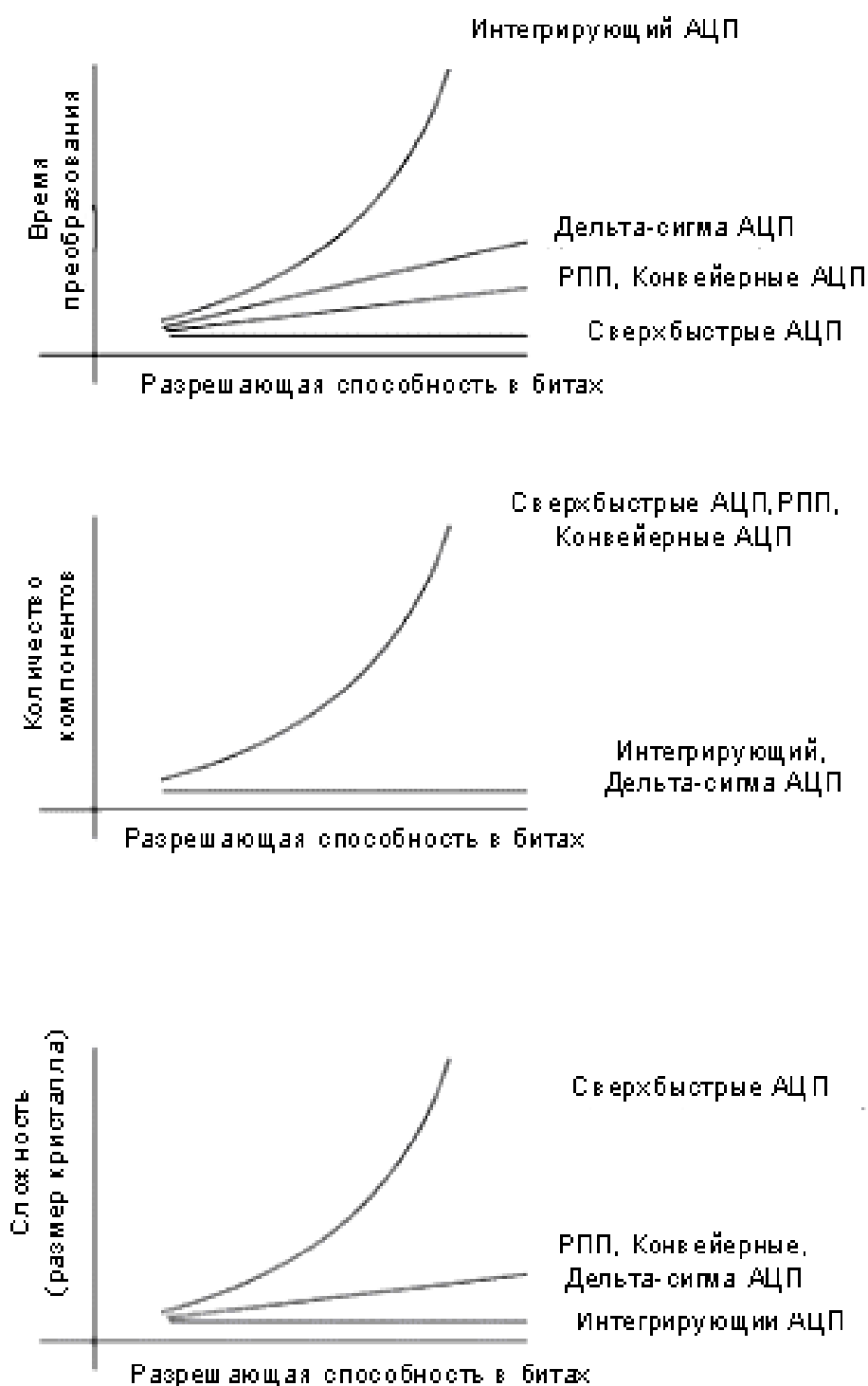


Рис.5.1. Отличительные характеристики АЦП с разной архитектурой

6. ЗАЩИТА ЭЛЕКТРИЧЕСКИХ ЦЕПЕЙ И ПОРТОВ ВВОДА-ВЫВОДА

Важной частью практически любой схемы УСО, особенно применительно к устройствам ЖАТ, является блок гальванической развязки, который обеспечивает гальваническое разделение ПК от внешних устройств и объектов управления. Это особенно актуально при управлении устройствами, работающими с большими токами и напряжениями. Гальваническая развязка плат ввода-вывода большинства производителей средств сбора данных используется для согласования интерфейсных линий шины данных персонального компьютера и узлов самой платы. Обычно при использовании параллельного способа передачи данных между шиной компьютера и платой ввода-вывода введение гальванической развязки понижает надёжность и увеличивает стоимость системы.

В настоящее время наиболее широкое распространение получили схемы, использующие два варианта гальванической развязки:

- трансформаторной;
- оптоэлектронной.

Гальваническая развязка трансформаторного типа предполагает использование магнитоиндукционного элемента с сердечником или без него, напряжение $U_{\text{вых}}$ на вторичной обмотке которого пропорционально напряжению $U_{\text{вх}}$ на входе устройства. При применении трансформаторной гальванической развязки необходимо учитывать следующие её недостатки:

- несущий сигнал может создавать помехи, влияющие на выходной сигнал развязки;
- полоса пропускания ограничена частотой модуляции развязки;
- сравнительно большие габаритные размеры компонентов, реализующих развязку.

Оптронами называют такие оптоэлектронные приборы, в которых имеются источник и приемник излучения (светоизлучатель и фотоприемник) с тем или иным видом оптической и электрической связи между ними, конструктивно связанные друг с другом.

Практически распространение получили лишь оптроны, у которых имеется прямая оптическая связь от излучателя к фотоприемнику и, как правило, исключены все виды электрической связи между этими элементами.

Достоинства этих приборов:

- возможность обеспечения идеальной электрической (гальванической) развязки между входом и выходом; для оптронов не существует каких-либо принципиальных физических или конструктивных ограничений по достижению сколь угодно высоких напряжений и сопротивлений развязки и сколь угодно малой проходной емкости;
- возможность реализации бесконтактного оптического управле-

ния электронными объектами и обусловленные этим разнообразие и гибкость конструкторских решений управляющих цепей;

- широкая частотная полоса пропускания оптрона, отсутствие ограничения со стороны низких частот (что свойственно импульсным трансформаторам); возможность передачи по оптронной цепи как импульсного сигнала, так и постоянной составляющей;

- физическая и конструктивно-технологическая совместимость с другими полупроводниковыми и микроэлектронными приборами.

Однако, наряду с этим, оптронам присущи и определенные недостатки:

- значительная потребляемая мощность, обусловленная необходимостью двойного преобразования энергии и невысокими КПД этих переходов;

- сложность реализации обратных связей, вызванная электрической разобщенностью входной и выходной цепей;

Оптрон, как элемент связи, характеризуется коэффициентом передачи K_i , определяемым отношением выходного и входного сигналов и максимальной скоростью передачи информации F . Практически вместо F измеряют длительность нарастания и спада передаваемых импульсов $t_{нар(сп)}$ или граничную частоту. Возможности оптрона как элемента гальванической развязки характеризуются максимальным напряжением и сопротивлением развязки $U_{разв}$ и $R_{разв}$ и проходной емкостью $C_{разв}$.

Выделяют три основные группы приборов оптронной техники; ранее названные как *оптопары* (элементарные оптроны), использующие блоки светоизлучатель - оптическая среда - фотоприемник; *оптоэлектронные* (оптронные) микросхемы (оптопары с добавлением выходного, а иногда и входного устройства); *специальные виды оптронов* - приборы, существенно отличающиеся функционально и конструктивно от элементарных оптронов и оптоэлектронных ИС.

Основным наиболее универсальным видом излучателя, используемым в оптронах, является полупроводниковый инжекционный светоизлучающий диод - светодиод. Это обусловлено следующими его достоинствами: высокое значение КПД преобразования электрической энергии в оптическую; узкий спектр излучения (квазимонохроматичность); широта спектрального диапазона, перекрываемого различными светодиодами; высокое быстродействие; малые значения питающих напряжений и токов; линейность ватт-амперной характеристики в более или менее широком диапазоне входных токов; высокая надежность и долговечность; малые габариты; технологическая совместимость с изделиями микроэлектроники. Диодные оптопары представлены на рис.6.1.

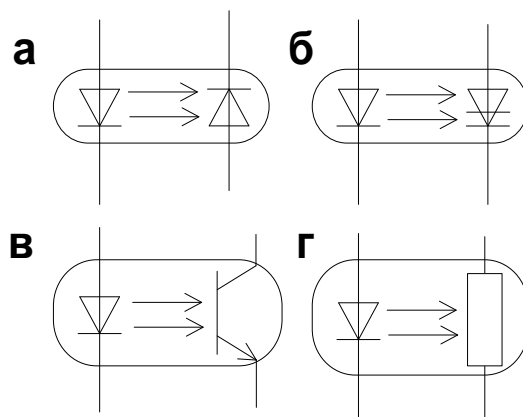


Рис.6.1. Условные обозначения оптопар: а – диодные, б – тиристорные, в – транзисторные, г – резисторные

Диодные оптопары (рис.6.1, а) в большой степени, чем какие-либо другие приборы, характеризуют уровень оптронной техники.

Транзисторные оптопары (рис.6.1, в) рядом своих свойств выгодно отличаются от других видов оптронов. Это, прежде всего, схемотехническая гибкость, проявляющаяся не только в том, что коллекторным током можно управлять как по цепи светодиода (оптически), так и по базовой цепи (электрически), но и в том, что выходная цепь может работать и в линейном, и в ключевом режимах.

Тиристорные оптопары (рис.6.1, б) наиболее перспективны для коммутации сильнотоковых высоковольтных цепей.

При передаче информации оптроны используются в качестве элементов связи, и, как правило, не несут самостоятельной функциональной нагрузки. Их применение позволяет осуществить весьма эффективную гальваническую развязку устройств управления и нагрузки (рис.6.2.), действующих в различных электрических условиях и режимах. С введением оптронов резко повышается помехоустойчивость каналов связи; практически устраняются "паразитные" взаимодействия по цепям "земли" и питания.

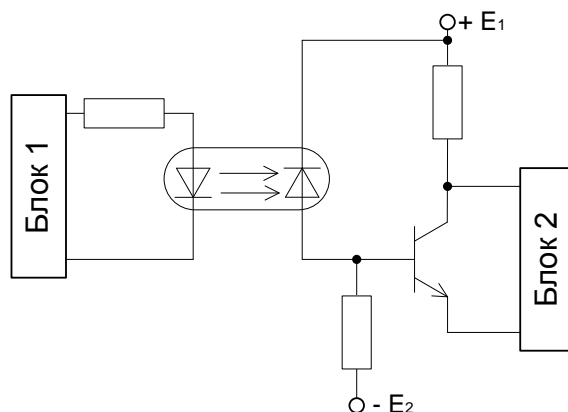


Рис.6.2. Схема межблочной гальванической развязки

7. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УСТРОЙСТВ

На стадии развития и совершенствования операционных систем программистам предоставлялись различные способы доступа к аппаратным средствам ПК.

Под MS-DOS приложение управляет всем компьютером. Максимальная скорость работы достигается непосредственным доступом к аппаратным средствам.

Под Windows3.x эта свобода отчасти была ограничена. Проблема объясняется тем, что с тех пор, как пользователь смог запускать любое количество приложений, не было никакой гарантии, что приложения не получали одновременно те же самые аппаратные средства. Другая проблема – необходимо было считаться с параллельно запущенными задачами, а не требовать у компьютера в свое распоряжение все ресурсы. Win3.x осуществляет кооперацию параллельных задач, означая, что каждое приложение должно исходить из концепции совместного существования и не монополизировать ресурсы, а пользоваться услугами специализированного диспетчера. Но, тем не менее, монополизированный доступ к аппаратным средствам также возможен, но вся ответственность за работу других приложений ложится на программиста. Получается борьба приложения с системой: если вы захватываете все рабочее время CPU, контроль над портами или работу с памятью, то система ждет, пока приложение не вернет контроль системе, при этом другие приложения могут конфликтовать, выдавая на экран сообщения об ошибках.

Тенденция блокирования прямого доступа к портам была заложена в основу платформы Win32 (Windows NT/2000/XP и Windows 95/98/ME). Это операционные системы с истинной многозадачностью. Каждый поток (выполняемый модуль) получает определенный квант процессорного времени. Когда лимит процессорного времени исчерпан или появляется поток с более высоким приоритетом, то система прекращает обслуживать первый поток, даже в случае, если он не завершен. Это переключение между потоками может произойти между двумя ассемблерными инструкциями, и нет никакой гарантии, что поток сможет завершить определенное количество инструкций, прежде чем у него отнимут процессорное время. К тому же, неизвестно, как долго ждать следующей порции процессорного времени. Это приводит к проблеме с прямым доступом к аппаратным средствам. Оптимальное решение - создание драйвера устройства, который единолично владеет портами/памятью. Однако, создание драйвера устройства – это достаточно сложная задача. Он должен быть создан с помощью ассемблера или Си и может быть труден в отладке. Более того, в целях безопасности драйверы устройств для Windows 95/98 (VxD) не совместимы с драйверами

для Windows NT (VDD, virtual device driver - виртуальный драйвер устройства).

Проблему удалось решить, заложив в Windows 95/98 совместимость с Windows 3.x. Это означает, что директивное использование I/O портов также возможно, поскольку до сих пор находятся в эксплуатации множество 16-битных программ, которые просто не могут работать по-другому. В этом случае при кодировании необходимо применение ассемблера, использующего стандартные операторы In и Out.

Инструкции ввода-вывода позволяют всем процессорам 80x86 общаться с другими устройствами в системе. Однако использование In или Out в приложении Windows NT даст сообщение об исключении в связи с привилегированной инструкцией и предложит выбор – завершить или отлаживать приложение. Если попытаться производить операции ввода-вывода из 16-битного DOS приложения в окне консоли Windows NT, то ввод-вывод игнорируется либо эмулируется виртуальными драйверами устройств NT, т.е. не произойдет исключения, но прямого ввода-вывода также не произойдет.

Это не является ошибкой системы. NT и должна работать таким образом. Проектировщики NT решили, что было бы слишком рискованно разрешать приложениям иметь прямой доступ к системным устройствам. При неограниченном доступе к портам ввода-вывода, приложение может запретить все прерывания, перехватить управление системой и испортить содержимое дисплея или жёсткого диска. Программа с ошибками может сделать то же самое непреднамеренно. Архитектура NT требует, чтобы доступ ко всем устройствам происходил через kernel-mode драйверы устройств – специальные, доверенные программные блоки, которые, практически, становятся частью операционной системы после того как были загружены. Эти драйверы устройств имеют полный доступ ко всей памяти системы, всем аппаратным устройствам и всем привилегированным инструкциям процессора. С другой стороны, приложения запущенные в режиме пользователя имеют ограниченный доступ к памяти, и процессор не может исполнить определённые привилегированные инструкции, включая команды ввода-вывода.

Для работы с последовательным портом обычно используется драйвер виртуального COM порта (VCP – Virtual COM Port). Они представлены в двух вариантах. Один - для устройств, подключаемых через преобразователь интерфейса и поддерживающих технологию PnP (Plug and Play). Другой - для аналогичных устройств без такой поддержки. В качестве инструмента программирования можно использовать семейство стандартных функций VCOMM API. Набор этих функций позволяет успешно проводить операции с шиной USB через созданный для нее драйвер виртуального последовательного порта.

При программировании системного интерфейса под Windows95/98/ME программисту предоставляется возможность обойти блокирование портов ввода-вывода системой путем применения низкоуровневого программирования на ассемблере, что существенно упрощает задачу управления устройствами и ускоряет быстроедействие. В тоже время в среде Windows NT/2000/XP прямое обращение к портам полностью блокируется системой, и программисту необходимо создавать kernel-драйвер устройства, при этом теряя в производительности и быстродействии. При создании драйвера программист сталкивается с множеством проблем, поэтому более приемлемым решением будет использование прямого обращения к портам с помощью ассемблера (а значит, на этом ПК должна быть установлена операционная система, не поддерживающая архитектуру NT:Microsoft Windows 95/98/ME). Пример листинга программы прямого чтения и записи в порт через использование инструкций ассемблера приведен ниже.

```
// Переменные
PortAdr:Word; // адрес порта ввода/вывода
Data:byte; // данные, передаваемые или принимаемые из порта
// запись в порт данных
    Push dx;
    Mov dx,PortAdr;
    Mov al,Data;
    Out dx,al;
    Pop dx;
// чтение данных из порта
    Push dx
    Mov dx,PortAdr;
    In al,dx;
    Pop dx;
    Mov Data,al;
```

Так как обращение к портам идет напрямую, без включения ресурсов ОС, то данный код позволяет достичь максимального быстрогодействия при минимальной загрузке ресурсов ОС.

Все программное обеспечение обслуживания функций управления и контроля может быть написано на любом языке высокого уровня с применением вставки ассемблерного кода. Ниже приведен примерный алгоритм управления объектами через рассмотренный нами интерфейсный блок УСО, подключенный к LPT порту (рис.7.1.).

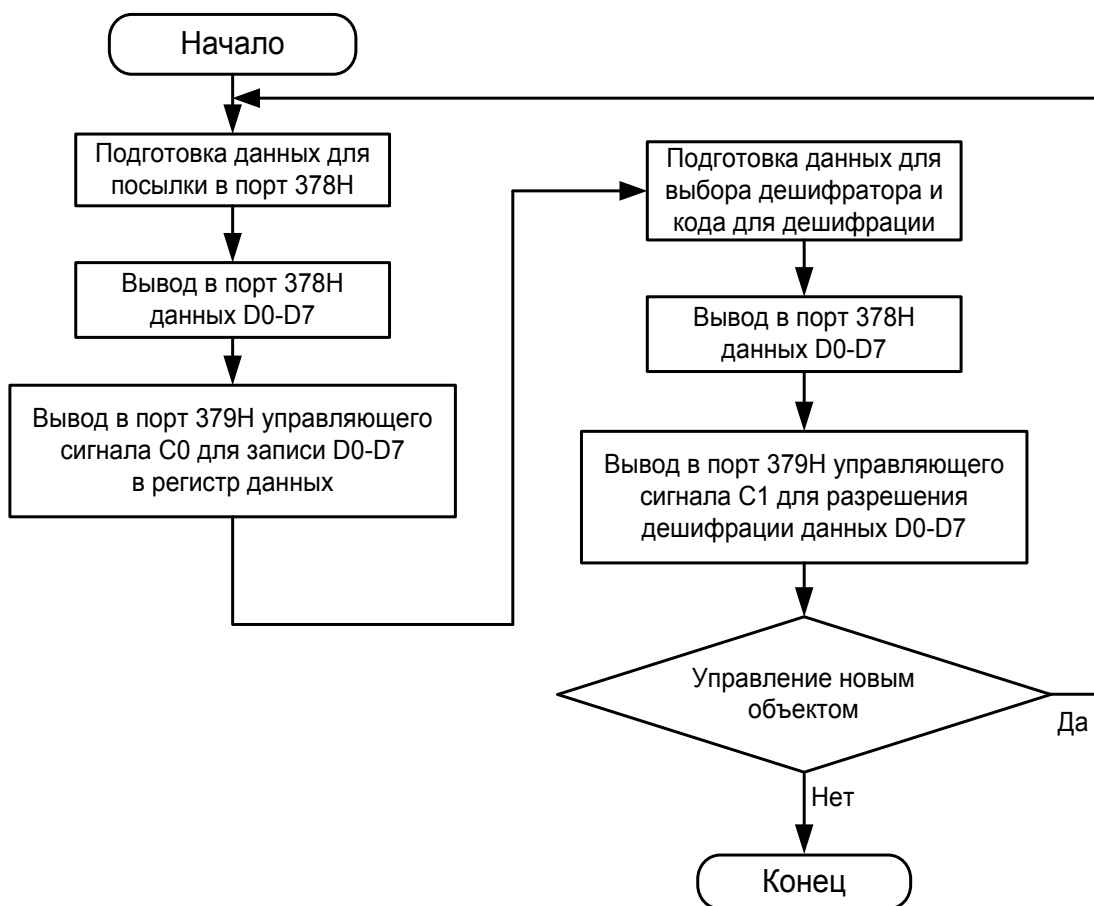


Рис.7.1. Алгоритм реализации управления объектами

Алгоритм реализации контроля состояния объектов и временная диаграмма работы устройства сопряжения при контроле представлены на рис.7.2.

Поскольку инструкции ассемблера In и Out в защищенном режиме недоступны, то можно воспользоваться встроенными процедурами работы с внешними устройствами как с файлами. Однако эти функции существуют только для последовательных портов. Следовательно, для NT необходимо создавать УСО для последовательного порта COM или шины USB.

В противном случае, либо создавать драйвер, либо применить библиотеки OCX (ActiveX компоненты), т.е. драйверы сторонних производителей, которые бы выполняли функции аналогичные инструкциям In и Out. Для разработчиков программного обеспечения в среде Delphi такой альтернативой стали динамические библиотеки IOPORT. Так, например, драйвер VicX предоставляет доступ к портам ввода-вывода на платформе Windows NT, при этом, не создавая конфликтов с другими приложениями. Драйвер содержит необходимый минимум функций для работы с портами: открытие порта, запись данных в порт, чтение данных из порта, закрытие порта и информацию о количестве LPT – портов в системе.

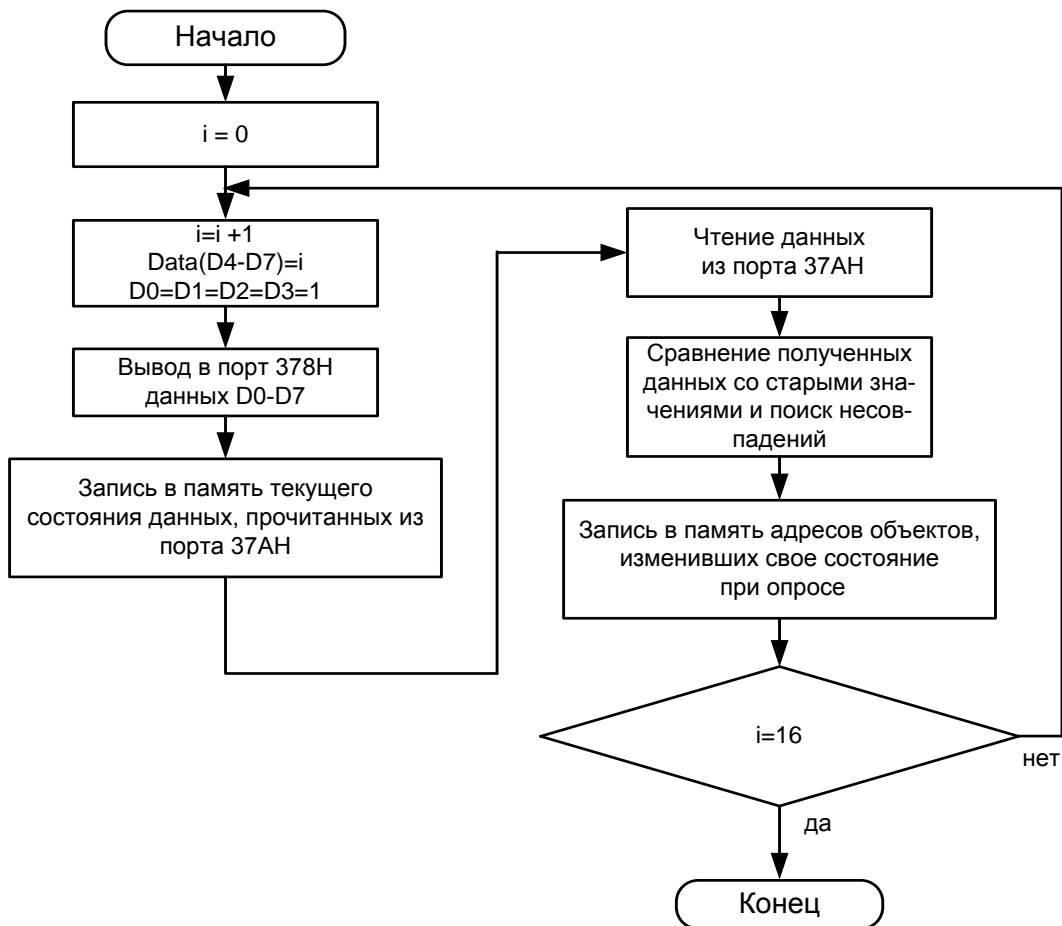


Рис.7.2. Алгоритм реализации контроля состояния объектов

Описание функций VICX

OpenPrt(Addr, size, PortCount); //Открыть порт

где Addr - Базовый адрес порта; size - размер порта(1 - BYTE, 2 - WORD, 3 - DWORD); PortCount - количество рядом расположенных портов

Функция OpenPrt возвращает дескриптор открытого порта или 0 в случае ошибки

R_P (Port, Index); //Чтение из порта

где Port - Дескриптор открытого порта, полученный вызовом функции OpenPrt (...); Index - индекс порта, из которого производится чтение(адрес порта в этом случае равен Addr+Index. Обязательное условие: Index < PortCount.)

Функция R_P возвращает данные, считанные из порта, или «-1» - в случае ошибки

W_P (Port, Data, Index); // Запись в порт

где Port - Дескриптор открытого порта, полученный вызовом функции OpenPrt (...); Data - Данные, записываемые в порт; Index - индекс порта, куда производится запись (адрес порта в этом случае равен Addr+Index. Обязательное условие : Index < PortCount.)

ClosePrt (Port); //Заккрыть порт

Port - Дескриптор открытого порта, полученный вызовом функции OpenPrt (...)

CountLPT (*port1, *port2, *port3); // Информация о LPT-портах, где *port1 (2,3) - Указатели на переменные, по которым записываются адреса портов.

Функция CountLPT возвращает количество LPT-портов в системе.

Пример вызова функций драйвера VicX:

```
Var Port:integer;  
    z:word;  
begin  
    Port = OpenPrt (0x378 , 1);  
    W_P(Port , 0xff);  
    z = R_P(Port);  
    ClosePrt(Port);  
End;
```

8. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Изучить общие принципы организации ввода-вывода в вычислительных системах. Ознакомиться с архитектурой системного интерфейса компьютера, дать сравнительную характеристику основным стандартным портам системной шины.

2. Изучить интерфейсы периферийного оборудования. Оценить преимущества и недостатки применения различных портов ввода-вывода.

3. Изучить общую функциональную схему устройства управления и контроля объектов. Проанализировать возможные варианты структур - в зависимости от требований поставленной задачи, типов интерфейсов и элементной базы.

4. По п.9 выбрать вариант исходных данных для разработки устройства связи с внешними объектами. На основании этих данных принять решение о структуре разрабатываемого УСО.

5. По имеющимся материалам и справочной литературе произвести подбор элементной базы, дать обоснование применению выбранных элементов и описание принципов и порядка работы основных из них.

6. Разработать основные схемные узлы УСО: каналные приемопередатчики; селектор адреса (если есть необходимость); схемы блока управления объектам; схемы контроля; схемы управления.

7. При использовании аналого-цифровых или цифро-аналоговых преобразователей произвести расчет требуемой разрядности АЦП (ЦАП) исходя из диапазона напряжений, диапазона преобразований и допустимой абсолютной погрешности. На основе полученных результатов подобрать АЦП (ЦАП) промышленного образца, отвеча-

ющий поставленным задачам. При необходимости разработать схему усилителей и согласования уровней для преобразователей.

8. Разработать принципиальные схемы подключения объектов управления с расчетом элементов схем транзисторных ключей или усилителей. На основании расчетов выполнить подбор транзисторов и предоставить их основные характеристики.

9. Построить принципиальную схему подключения объектов контроля, учитывая тот факт, что дискретные объекты представляют собой пару свободных контактов (замкнут-разомкнут), а аналоговые – пары гальванически несвязанных выходов источников измеряемых напряжений.

10. Разработать общую принципиальную схему УСО, представить спецификацию элементов и построить временные диаграммы работы устройства.

11. Составить логическую диаграмму последовательности действий при операциях чтения-записи в порт компьютера. В диаграмме должны быть указаны информационные биты (байты) в циклах записи, чтения и управления в двоичном и (или) шестнадцатеричном виде.

12*. Разработать алгоритм и программное обеспечения циклов ввода-вывода, передачи информации к объектам управления, сбора и хранения данных от объектов контроля с применением средств низкоуровневого программирования или через библиотеки виртуальных драйверов. (В зависимости от заданной преподавателем операционной системы).

13*. Посредством применения среды разработки программ создать проект, визуально отображающий состояние объектов контроля и позволяющий в интерактивном режиме передавать команды к объектам управления.

14. Работу представить в виде отчета, содержащего пп.4 -13 порядка выполнения работы.

Примечание * - Только для специализации МПИУС.

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Укажите, какие параметры используются при оценке производительности вычислительной системы?

2. В чем суть понятия «интерфейс ввода-вывода»?

3. Какие организации занимаются стандартизацией шин ввода-вывода?

4. Оцените скорость обмена информацией стандартных устройств ввода-вывода ПК.

5. Перечислите основные контроллеры портов ввода-вывода.

6. Перечислите и дайте сравнительную характеристику основных

стандартов системного интерфейса.

7. Какие функции выполняют каналные приемо-передатчики в устройствах связи с объектами?

8. В каких схемах УСО необходимо разрабатывать селектор адреса? А в каких такой необходимости нет?

9. В каких случаях применяются аналого-цифровые и цифро-аналоговые преобразователи?

10. Приведите возможные схемы аналоговых коммутаторов.

11. Перечислите основные достоинства и недостатки устройства связи с объектами на основе системного интерфейса ПК.

12. Перечислите основные достоинства и недостатки устройства связи с объектами на основе периферийных интерфейсов ПК.

13. Сколько портов выделено для параллельного периферийного интерфейса, укажите их адреса, разрядность и назначение.

14. Какие режимы и скорость передачи обеспечивает последовательный интерфейс ПК?

15. Как зависит дальность и скорость передачи по COM порту от типа соединительного кабеля?

16. Перечислите основные стандарты последовательных интерфейсов.

17. Какими достоинствами обладает универсальная последовательная шина USB?

18. Что понимается под USB хабом?

19. Что представляют собой USB функции?

20. Какую роль выполняет USB хост?

21. Какие принципиальные отличия имеют шины ISA-8 и ISA-16?

22. Что такое канал прямого доступа к памяти?

23. Укажите основные различия между статическими и динамическими, одноктактными и двухтактными регистрами.

24. В каких случаях целесообразно применение параллельного периферийного интерфейса на микросхеме K580BB55?

25. Для какой цели устанавливаются полупроводниковые элементы при построении матрицы контроля двухпозиционных объектов?

26. Как рассчитывается транзисторный ключ в схеме управления объектом?

27. В каких случаях необходимо применение мультиплексоров в схеме контроля?

28. Как определяется коэффициент разветвления при расчете числа нагружающих входов для микросхем различных серий?

29. Перечислите основные характеристики аналого-цифровых и цифро-аналоговых преобразователей.

30. Какие параметры влияют на точность цифро-аналогового преобразователя?

31. Как рассчитывается разрядность АЦП, исходя из диапазона

преобразования и допустимой абсолютной погрешности?

32. Какие существуют архитектуры аналого-цифровых преобразователей?

33. Перечислите основные элементы, обеспечивающие гальваническое разделение сигналов.

34. Предложите вариант схемного решения устройства ввода с коммутируемым дифференциальным входом.

35. Для чего создается драйвер устройства?

36. Каким образом можно организовать прямой ввод-вывод в защищенном режиме?

37. В каких случаях можно обойти блокировку портов ввода-вывода при использовании низкоуровневого программирования на Ассемблере?

ВАРИАНТЫ ИНДИВИДУАЛЬНЫХ ЗАДАНИЙ

Варианты разрабатываемой системы управления и контроля выбираются по таблице, приведенной в конце (с.71). Номер варианта определяется по списочному номеру студента в группе с учетом номера группы. Так, для студентов 245 группы номера вариантов начинаются с первого по 24-й, а для студентов 246 группы первый списочный номер соответствует 25-му варианту. Для студентов-заочников номер варианта вычисляется по двум последним цифрам шифра зачетной книжки.

Например, если шифр студента КТ00-АТС-043, то его номер варианта будет, соответственно, 43, а в случае, если последние две цифры составляют число более 50, то из него вычитается 50 (т.е. для шифра КТ00-АТС-366 номер варианта будет 16). При использовании в разрабатываемом УСО портов системного интерфейса базовый адрес порта ввода-вывода определяется выражением: $N_{\text{внн}} * 8 + 300H$. (Например: при номере варианта 16, базовый адрес порта будет определен как:

$$16 * 8 + 300H = 128 (80H) + 768 (300H) = 380H).$$

В таблице содержатся следующие поля:

$N_{\text{внн}}$ - номер варианта;

$N_{\text{упр}}$ - количество дискретных или аналоговых объектов управления;

$N_{\text{контр}}$ - количество двухпозиционных объектов контроля (кнопки или пары контактов) или объектов измерения (аналоговых);

$I_{\text{упр}}$ - максимальный ток, потребляемый дискретным объектом управления;

$U_{\text{упр}}$ - напряжение включения объекта управления (напряжение срабатывания реле, питания ламп или других источников нагрузки);

$I_{аналог}$ - максимальный потребляемый ток аналогового источника нагрузки;

$U_{диапаз}$ - диапазон напряжений для аналоговых объектов управления;

Δ - допустимая абсолютная погрешность по управлению;

$U_{диап.контр.}$ - диапазон измеряемых напряжений для аналоговых объектов контроля;

Δ_k - допустимая абсолютная погрешность по контролю;

Порт – необходимость применения интерфейса с указанным в этом поле портом;

Примеч. – дополнительные условия по использованию определенной элементной базы при создании УСО.

Пояснения

1. Если поле, указывающее $I_{упр}$ или $U_{упр}$ не заполнено, то это означает, что в разрабатываемой системе цифровых объектов нет.

2. Если поля с характеристиками аналоговых требований по управлению или контролю заполнены, то из этого следует, что в полях $N_{упр}$ или $N_{контр}$ указано количество аналоговых объектов. Поле Порт содержит требование по созданию УСО на системном или периферийном интерфейсе.

3. Если в поле не указано ничего, то студент, исходя из заданных условий информационной емкости УСО, выбирает тип интерфейса самостоятельно.

Таблица

Варианты индивидуальных заданий

$N_{\text{впн}}$	$N_{\text{упр}}$	$N_{\text{контр}}$	$I_{\text{упр}}$	$U_{\text{упр}}$	$I_{\text{аналог}}$	$U_{\text{диапаз}}$	Δ	$U_{\text{диап.контр.}}$	Δ_{κ}	Порт	Примеч.
1	80	250	35mA	24V							
2	160	30	100mA	24V						ISA	580BB55
3	30	60	1A	5V						ISA	
4	50	50	1A	5V							580BB55
5	100	50	100mA	5V							
6	128	16	100mA	48V						LPT	
7	256	8	100mA	60V						LPT	
8	1	100			1A	-10..10V	10mV				
9	5	50			100mA	-10..10V	10mV				
10	10	10			1A	0..30V	0,1V				
11	8	20	1A	60V				-10..10V	10mV	ISA	
12	80	10	2mA	5V				0..30V	0,1V	LPT	
13	10	10			100mA	0..30V	0,1V				
14	8	200	1A	400V							
15	160	400	2mA	5V							580BB55
16	16	1	2mA	2mA				0..400V	1V		
17	10	100			100mA	0..30V	0,1V				
18	2	4			1A	-10..10V	1mV	-10..10V	1mV	ISA	
19	80	20	100mA	60V				-10..10V	10mV	LPT	
20	8	10	2mA	5V				0..30V	0,1V	ISA	
21	10	200			500mA	0..5V	1mV				
22	8	2			500mA	-10..10V	1mV	-10..10V	1mV	LPT	
23	128	1	35mA	24V				0..30V	0,1V		
24	48	10	90mA	24V							
25	10	300			100mA	0..12V	0,1V				
26	8	1	400mA	24V				-5..5V	10mV		580BB55

Окончание таблицы 6

$N_{\text{пп}}$	$N_{\text{упр}}$	$N_{\text{контр}}$	$I_{\text{упр}}$	$U_{\text{упр}}$	$I_{\text{аналог}}$	$U_{\text{диапаз}}$	Δ	$U_{\text{диап.контр.}}$	$\Delta_{\text{к}}$	Порт	Примеч.
27	100	10	500mA	24V							580BB55
28	200	10	500mA	24V							
29	300	40	100mA	60V							
30	400	400	100mA	60V							
31	500	100	2mA	5V							
32	1000	1	2mA	5V				0..30V	1mV		
33	28	200	1A	400V							
34	16	400	2mA	5V							
35	160	1	2mA	2mA				0..400V	2V		
36	10	100			100mA	0..30V	0,1V				
37	20	40			1A	-10..10V	10mV	-10..10V	10mV		
38	8	200	100mA	60V				-10..10V	2mV		
39	180	10	200mA	5V				0..30V	0,1V		580BB55
40	2	400			1A	-5..5V	1mV				
41	2	4			1A	-5..5V	1mV	-10..10V	10mV		
42	2	48			1A	-5..5V	1mV	-10..10V	5mV		
43	8	256	500mA	24V						ISA	
44	16	128	500mA	24V						LPT	
45	32	64	1A	5V						ISA	
46	48	48	1A	5V						LPT	
47	64	32	1A	5V							
48	128	16	1A	5V						ISA	
49	256	8	1A	5V						ISA	
50	1000	1000	2mA	5V							
51	200	200	2mA	5V						COM	
52	400	400	2mA	5V						USB	

ЗАКЛЮЧЕНИЕ

Современные устройства автоматики и телемеханики опираются на передовые технологии управления, контроля и диагностики и предъявляют повышенные требования к инженерным кадрам в области знания архитектур микропроцессорных систем, схемотехники и программно-аппаратной организации как вычислительной техники, так и методов ее взаимодействия с исполнительными устройствами.

Выполнение практической работы по разработке устройства связи с внешними объектами требует понимания внутренней архитектуры ввода-вывода компьютера, знания элементной базы микроэлектронной аппаратуры и основ программирования функций ввода-вывода. Предложенный материал должен выработать системный подход в выборе структуры устройства, разрабатываемого для определенных целей управления. Этого требует современный уровень производства.

Настоящее пособие поможет сформировать творческий подход в области схемотехники, системотехники, при изучении и разработке микропроцессорных информационно-управляющих систем, диагностических производственных комплексов, т.е. тех устройств, которые призваны осуществлять сбор и обработку информации, формировать управляющие сигналы, выполнять функции контроля и регулирования.

РЕКОМЕНДУЕМЫЙ БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Абрайтис, Б.Б. Микропроцессоры и микропроцессорные комплекты интегральных микросхем. Справочник. В 2-х т. Т. 1 / Б.Б. Абрайтис, Н.Н. Аверьянов, А.И. Белоус и др. – М.: Радио и связь, 1988. - 368 с.
2. Якубовсий, С.В. Цифровые и аналоговые интегральные микросхемы. Справочник / С.В. Якубовский, Л.И. Ниссельсон, В.И. Кулешова и др. – М.: Радио и связь, 1989. – 496 с.
3. Гук, М. Интерфейсы ПК. Справочник / М. Гук – СПб: Питер Ком, 1999. – 416 с.
4. Гук, М. Аппаратные средства локальных сетей. Энциклопедия / М. Гук - СПб: Изд. «Питер», 2000. – 576 с.
5. Гук, М. Аппаратные средства IBM PC. Энциклопедия /М. Гук - СПб: Изд. «Питер», 2000. – 816 с.
6. Усатенко, С.Т. Выполнение электрических схем по ЕСКД. Справочник / С.Т. Усатенко – М.: Издательство стандартов, 1989. - 325 с.
7. Шило, В.Л. Популярныe цифровые микросхемы. Справочник, 2-е изд., испр. / В.Л. Шило - Челябинск, 1989. - 352 с.
8. Глазенко, Т.А. Электротехника и основы электроники: Учеб. пособие для неэлектротехн. спец. вузов / Т.А. Глазенко, В.А. Пряшников – М.: Высш. шк., 1996. – 207 с.
9. Пей, Ан. Сопряжение ПК с внешними устройствами / Ан. Пей; Пер. с англ. – М.: ДМК Пресс, 2001. – 320 с.
10. Лысенко, А.А. Преобразователи интерфейса USB на микросхемах FT8U232AM, FT8U245AM / А.А. Лысенко, Р.Н. Назмутдинов // Радио. – 2002. - №6. – С. 36-39.
11. Коломбет, Е.А. Микроэлектронные средства обработки аналоговых сигналов / Е.А. Коломбет. – М.: Радио и связь, 1991. – 376 с.
12. Нефедов, А.В. Зарубежные интегральные микросхемы для промышленной электронной аппаратуры. Справочник / А.В. Нефедов, А.М. Савченко, Ю.Ф. Феоктистов – М.: Энергоатомиздат, 1989. – 288 с.
13. Хоровиц, П. Искусство схемотехники. В 2-х т. Т. 1 / П. Хоровиц, У. Хилл Изд. 3-е, стереотип. – М.: Мир, 1986. – 598 с.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
1. АРХИТЕКТУРА СИСТЕМ ВВОДА-ВЫВОДА	5
2. ОРГАНИЗАЦИЯ СТАНДАРТНЫХ ИНТЕРФЕЙСОВ ВВОДА-ВЫВОДА	12
2.1. Параллельный интерфейс LPT	12
2.2. Последовательный интерфейс COM	15
2.3. Последовательный интерфейс USB	18
2.4. Структура и взаимодействие системы USB	19
3. ПРИМЕНЕНИЕ ШИН СИСТЕМНОГО ИНТЕРФЕЙСА ПК	26
4. СОЗДАНИЕ УСО С ПАРАЛЛЕЛЬНЫМ ИНТЕРФЕЙСОМ	40
5. ПРИМЕНЕНИЕ АНАЛОГОВЫХ УСТРОЙСТВ	57
6. ЗАЩИТА ЭЛЕКТРИЧЕСКИХ ЦЕПЕЙ И ПОРТОВ ВВОДА-ВЫВОДА	61
7. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УСТРОЙСТВ	64
ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ	69
ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ	70
ВАРИАНТЫ ИНДИВИДУАЛЬНЫХ ЗАДАНИЙ	72
ЗАКЛЮЧЕНИЕ	76
РЕКОМЕНДУЕМЫЙ БИБЛИОГРАФИЧЕСКИЙ СПИСОК	77

