

Министерство цифрового развития, связи и массовых коммуникаций
Российской Федерации
Северо-Кавказский филиал
ордена Трудового Красного Знамени федерального государственного бюджетного образовательного учреждения высшего образования
"Московский технический университет связи и информатики"



Методические указания
к лабораторным и практическим занятиям

МИКРОПРОЦЕССОРЫ

Периферийные устройства

Ростов-на-Дону
2022

УДК 681.3.06 (076)

ББК 32.07

Чикалов А.Н. Микропроцессоры. Периферийные устройства. Методические указания к лабораторным и практическим занятиям. Ростов-на-Дону: Северо-Кавказский филиал МТУСИ, 2022.- 44 с.

В пособии изложены методические рекомендации, содержательные материалы и контрольные задания для проведения лабораторных и практических занятий по изучению принципов построения периферийных устройств микропроцессорных систем, исследованию режимов работы периферийных микросхем, принципов подключения и программирования. Пособие содержит необходимые справочные материалы.

Методические указания предназначены для студентов, обучающихся по направлениям подготовки 09.03.01 Информатика и вычислительная техника и 11.03.02 Инфокоммуникационные технологии и системы связи, профилей Многоканальные телекоммуникационные системы, Сети связи и системы коммутации, Защищенные системы и сети связи, Системы радиосвязи и радиодоступа, Вычислительные машины, комплексы, системы и сети, Программное обеспечение и интеллектуальные системы.

Пособие предназначено для использования при изучении дисциплин Микропроцессорные системы, Вычислительная техника и информационные технологии, а также может быть использовано преподавателями и студентами при изучении родственных дисциплин и в процессе самостоятельной работы.

Учебное пособие обсуждено и одобрено на заседании кафедры ИВТ
Протокол от 26 августа 2021 г. № 1

Рецензент Зав. кафедрой ИВТ д.т.н. профессор Соколов С.В.

СОДЕРЖАНИЕ

1. Анализ режимов программирования параллельного интерфейса . . .	4
2. Анализ режимов программирования последовательного интерфейса .	17
3. Программирование интервального таймера	32

1. АНАЛИЗ РЕЖИМОВ ПРОГРАММИРОВАНИЯ ПАРАЛЛЕЛЬНОГО ИНТЕРФЕЙСА

Цель

1. Изучить принципы использования параллельного обмена в МПС;
2. Совершенствовать навыки анализа, обобщения и систематизации полученных результатов, навыки составления и оформления отчетных материалов, навыки точного и лаконичного представления докладов на вопросы технического характера.

Учебные вопросы

- 1.1. Структурная схема адаптера;
- 1.2. Программная модель адаптера и режимы работы;
- 1.3. Программирование режимов работы адаптера.

Литература для подготовки к занятию

1. Каган Б.М., Сташин В.В. Основы проектирования микропроцессорных устройств автоматики. – М.: Энергоатомиздат, 1987. – 304 с.
2. Алексенко А.Г., Галицин А.А., Иванников А.Д. Проектирование радиоэлектронной аппаратуры на микропроцессорах. – М.: Радио и связь, 1984. – 272 с.
3. Угрюмов Е.П. Цифровая схемотехника: Учебное пособие для вузов. - СПб.: БХВ-Петербург, 2007. - 800 с. (стр.404-419).
4. <http://www.computer-museum.ru/technlgy/i8255.htm>

Содержание отчета

1. Название работы;
2. Для каждого задания: название задания и материал в объеме, указанном в задании.

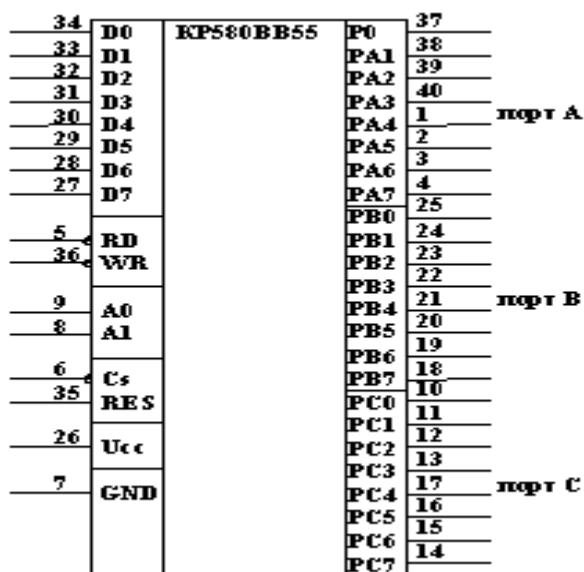


Рис.1.1. УГО ППА KP580BB55

Задание 1.1 Структурная схема адаптера

Условное графическое обозначение адаптера приведено на рис.1.1.

Обмен данными в параллельной форме в микропроцессорной системе может быть организован различными способами. Это может быть только вывод слова данных на внешнее устройство, только ввод слова данных с внешнего устройства или двухсторонний обмен. Любой из способов может быть организован

как асинхронный или синхронный обмен. Значительно упрощает подобную задачу использование программируемого периферийного адаптера (ППА) KP580BB55 (аналог Intel 8255).

Схема может выполнять буферизацию (при асинхронном режиме обмена данными); преобразование формата данных; сопряжение микро-ЭВМ с ВУ в режиме прерывания.

Микросхема ППА содержит три восьмиразрядных канала (порты PA, PB, PC), буфер шины данных (BD), необходимый для согласования шины МПС и входов ППА, схему управления режимом и две схемы управления портами группы А и В, программно объединенных в один регистр управляющего слова (РУС). По шине данных происходит не только обмен данными, но и засылка из МП в ППА управляющих слов, генерируемых программным обеспечением процессора, и передача в МП информации о состоянии периферийного оборудования. Низкий уровень сигнала на управляющем входе CS разрешает активизировать информационную связь между ППА и МП (рис.1.2). При отсутствии активного сигнала на входе CS выходы всех трех каналов переводятся в высокоимпедансное состояние.

В таблице 1.1 приведено назначение входов и выходов схемы.

Таблица 1.1

Назначение выводов

D0...D7	Двунаправленная шина данных. Предназначена для передачи и приема данных процессором, а также передачи управляющего слова для ППА
CS	Chip Select. Выбор микросхемы. Низкий уровень инициирует обмен между процессором и ППА
RD	Read. Чтение. Низкий уровень на этом входе позволяет ППА передать данные процессору
WR	Write. Запись. Низкий уровень на этом входе позволяет ППА принять данные или управляющее слово
RESET	Reset. Сброс. Высокий уровень на этом входе сбрасывает все внутренние регистры, включая регистры управления, переключает все каналы (А, В, С) в режим 0, одновременно переводя их в режим ввода (только при RD = 0 и WR = 0)
A0, A1	Адресные входы. Выбор канала. Позволяют выбрать один из каналов (А, В, С) или регистр управляющего слова (РУС)
PA0...PA7	Канал А
PB0...PB7	Канал В
PC0...PC7	Канал С

ППА структурно разбит на две группы: А и В. Группа А может работать в режимах 0, 1, 2 и включает в себя канал А и старшую тетраду канала

С. Группа В может работать в режиме 0, 1 и состоит из канала В и младшей тетрады канала С.

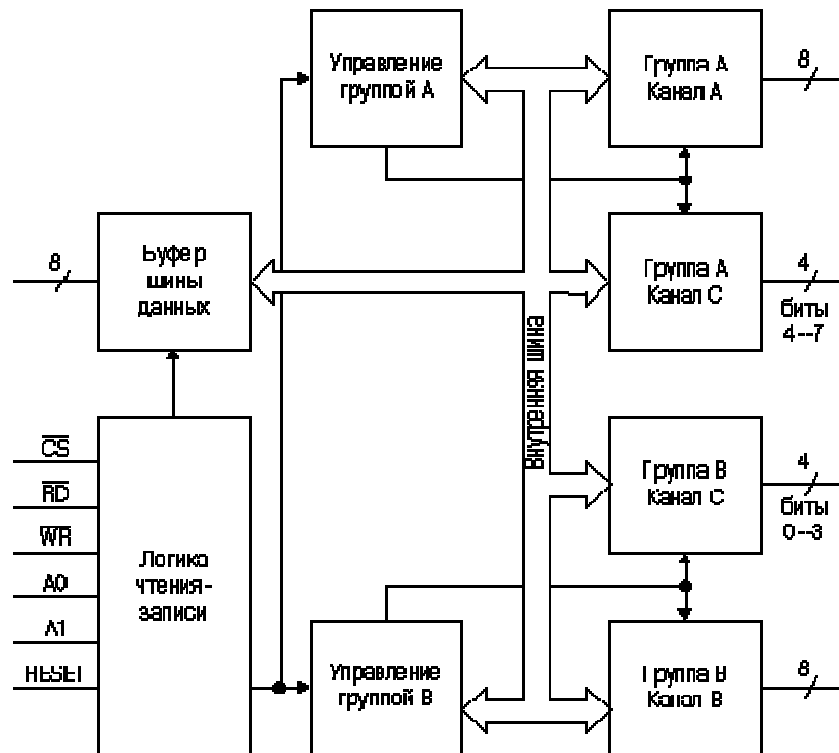


Рис.1.2. Структурная схема ППА KP580BB55

В отчете представить:

1. Структурную схему адаптера;
2. Назначение входов и выходов устройства.

Задание 1.2 Программная модель адаптера и режимы работы

Программно ППА представляет собой четыре 8-разрядных порта, которые адресуются адресными входами A0 и A1. Один порт является регистром управляющего слова (или состояния), а другие образуют каналы ввода/вывода А, В и С. Доступ к каналам осуществляется в соответствии с табл.1.2.

Таблица 1.2

Доступ к каналам ввода/вывода и регистру управляющего слова

A1	A0	RD	WR	CS	Направление передачи
0	0	0	1	0	Шина данных <- канал А
0	1	0	1	0	Шина данных <- канал В
1	0	0	1	0	Шина данных <- канал С
1	1	0	1	0	Данные не определены
0	0	1	0	0	Шина данных -> канал А

0	1	1	0	0	Шина данных -> канал В
1	0	1	0	0	Шина данных -> канал С
1	1	1	0	0	Шина данных -> регистр управления
X	X	X	X	1	Шина данных в Z-состоянии

Первоначальное включение или подача сигнала высокого уровня на вход сброс RESET микросхемы переключает все каналы (А, В, С) в режим 0, одновременно переводя их в режим ввода. Сброс выполняется только при условии $RD = 0$ и $WR = 0$.

Режимы работы ППА

Восьмиразрядные порты РА, РВ и РС адаптера могут быть использованы различным способом в зависимости от характеристик конкретного устройства ввода/вывода или периферийного оборудования.

Настройка внутренней организации ППА для конкретного периферийного оборудования, осуществляется при записи управляющего слова ВЫБОР РЕЖИМА (MODE SELECTION). Каналы ППА могут работать в одном из трех основных режимов:

В режиме 0 (основной, нестробируемый режим ввода-вывода информации, прямой однонаправленный статический ввод/вывод) могут работать все три порта. Направление для каждого порта определяется индивидуально. Порт РС можно разделить на две независимых половины. Выходы имеют память, входы работают без буферной памяти. Настройка каждого канала осуществляется индивидуально.

В этом режиме информация, поступающая на входы каналов, не запоминается и сигналы должны присутствовать при чтении данных процессором. При поступлении сигнала чтения данные передаются на шину данных. При выводе в режиме 0 информация, поступающая с шины данных, запоминается в буферном регистре соответствующего канала и сразу же передается на его выходы.

Обычно режим 0 используется для ввода-вывода медленно изменяющихся или постоянных значений данных. В режиме 0 параметры работы РА и РВ задаются независимо друг от друга.

В режиме 1 (стробируемого однонаправленного ввода-вывода информации, тактируемого программно-управляемого ввода/вывода, асинхронного по отношению к процессору) может работать порт РА и порт РВ. Для управления передачей или приемом по каналу А используется старшая часть канала С (С7-С3), канал В управляется младшей частью канала С (С2-С0).

Для организации этой процедуры порт РС служит регистром состояний внешнего устройства, порты РА и РВ для передачи данных. Опрос и модификация разрядов порта РС осуществляется программно. Порт РС

считывается процессором, а затем анализируется состояние его отдельных бит. На основании анализа применяется решение о выводе или вводе информации.

Описание сигналов для управления операцией **ввода**:

STB (STROBE) - строб приема от УВВ. Входной сигнал для порта. По низкому уровню на этом входе входные данные записываются в порт соответствующего канала (порт в этом случае играет роль буфера данных);

IBF (INPUT BUFFER FULL - входной буфер полон) - подтверждение приема в порт. Выходной сигнал порта. Появление высокого уровня на этом выходе показывает устройству ввода, что данные загружены в порт, еще не прочитаны, и следующий байт принимать невозможно;

INTER (INTERRUPT) - запрос на прерывание. Высокий уровень на этом выходе может быть использован для прерывания выполнения основной программы процессора (если программно в процессоре установлен режим разрешения прерывания), чтобы обслужить устройство ввода: прочитать из порта данные.

В этом режиме ППА может быть использован для реализации системного (асинхронного) обмена с применением квитирования. Квитирование связано с обменом информационными и управляющими сигналами-"квитанциями". При этом порты РА и РВ образуют два канала, а порт РС используется для накопления и обработки сигналов управления обменом.

При вводе в режиме 1 устройство, требующее обмена, формирует на соответствующем разряде порта РС (РС4 для порта РА или РС2 для порта РВ) инверсный управляющий сигнал STB (строб байта), по которому загружается в выбранный порт 8-ми разрядный код данных и там запоминается. В ответ ППА формирует на УВВ единичный сигнал подтверждения заполнения порта IBF, извещающий УВВ о невозможности приема следующего байта. С появлением сигнала подтверждения приема IBF внешнее устройство должно снять сигнал строба STB.

Со снятием строба STB формируется и сигнал запроса на прерывание INTER, который по системной шине инициирует обработчик для чтения данных из порта в процессор по инверсному сигналу RD. С появлением сигнала чтения RD (активный 0) снимается запрос на прерывание, а после окончания сигнала чтения RD снимается и сигнал подтверждения приема IBF, что свидетельствует о прочтении процессором данного байта, и дает возможность загружать очередной байт в порт ППА со стороны УВВ. Канал приема данных переходит в Z-состояние.

Этот процесс обеспечивает ввод данных в микропроцессор со стороны устройства ввода простым стробированием своих данных.

Взаимодействие описанных сигналов показано на рис.1.3.

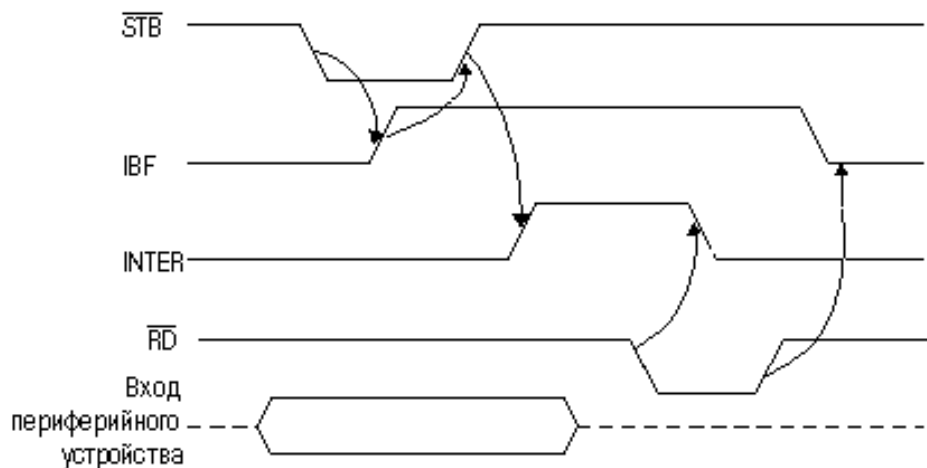


Рис.1.3. Динамика сигналов управления вводом режим 1

Работа с формированием сигнала прерывания возможна при наличии разрешения выработки портом такого сигнала - установке бита C4 для канала PA и C2 для канала PB в единичное состояние. Это приводит к установке внутреннего триггера разрешения прерывания. Установка/сброс этих разрядов (как, впрочем, и остальных разрядов порта PC) осуществляется программным способом путем формирования специальной команды работы с битами порта PC.

Описание сигналов для управления операцией **вывода**:

OBF (OUTPUT BUFFER FULL - выходной буфер полон) - строб записи. Выходной сигнал порта. Сигнал низкого уровня на этом выходе показывает устройству вывода, что микропроцессор записал данные в канал;

ACK (ACKNOWLEDGEMENT - подтверждение) - подтверждение приема данных внешним устройством. Входной сигнал для порта. Низкий уровень на этом выводе показывает ППА, что периферийное устройство приняло данные. Происходит снятие Z-состояния порта и выдача данных на UBB;

INTER (INTERRUPT). Высокий уровень на этом выходе может быть использован для прерывания выполнения основной программы процессора в момент принятия данных внешним устройством. Сигнал говорит о возможности выполнения следующего цикла вывода.

Вывод в режиме 1 тоже связан с процессом квитирования. При выводе в режиме 1 цикл работы начинается с записи байта из процессора в порт (PA или PB). С завершением сигнала записи WR (активный уровень 0) формируется строб записи OBF, который извещает UBB о том, что МП выдал байт в порт. Когда устройство ввода-вывода готово прочитать байт, оно генерирует в порт активный нулевой сигнал подтверждения ACK. По этому

сигналу открываются выходные буферы (снимается Z-состояние) соответствующих портов РА или РВ, данные передаются в УВВ. Появление сигнала подтверждения приема АСК снимает строб записи ОВФ и формирует запрос прерывания INTER для вывода очередного байта.

Взаимодействие описанных сигналов показано на рис.1.4.

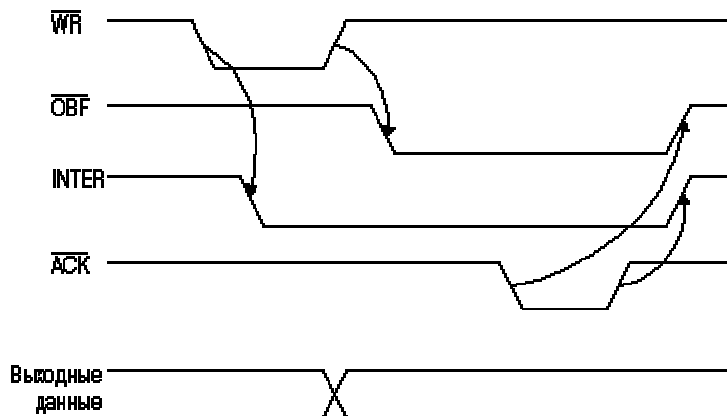


Рис.1.4. Взаимодействие сигналов управления выводом

Вывод с формированием сигнала прерывания возможен при наличии разрешения выработки такого сигнала - установке бита С6 для канала РА и С2 для канала РВ в единичное состояние. Их установка производится, как и для режима ввода, программно при работе с битами порта РС.

Режим 1 ППА удобно использовать в развитых МП системах, обслуживающих, например, большое количество датчиков и устройств ВВ.

В режиме 2 (режим двунаправленного тактируемого ввода/вывода) работает только порт РА как двунаправленный канал. Управление обменом данными, а также направление передачи задается с помощью 5 управляющих сигналов порта РС аналогичным режиму 1 способом. Возможно также генерирование сигналов прерываний и запрета разрешения. Назначение сигналов управления аналогично ранее описанным для ввода и вывода.

Режимы на ввод или вывод задаются для каждого порта независимо. Входы и выходы обладают буферной памятью. Организация обмена и программирование ППА в этом режиме аналогична режиму 1. Временные диаграммы управления для режима 2 показаны на рис.1.5.

Разрешение формирования сигналов прерывания INTER осуществляется индивидуальной программной установкой в единичное состояние разрядов РС4 для ввода и РС6 для вывода байта.

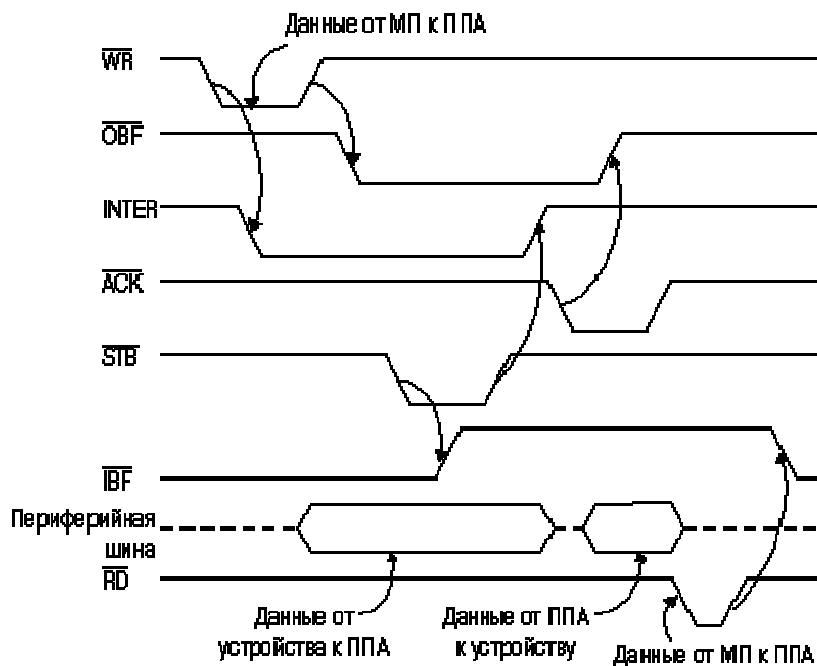


Рис.1.5. Работа ППА в режиме 2

Использование ППА в режиме 2 удобно при организации обмена информацией между двумя микро-ЭВМ или центральным вычислителем и контроллером сложного устройства ввода/вывода.

На основе ППА можно разрабатывать схемы управления различными устройствами, например, клавиатурой или индикатором.

Управляющие сигналы для различных режимов сведены в табл.1.3.

Таблица 1.3

Управляющие сигналы для различных режимов работы

C7	C6	C5	C4	C3	C2	C1	C0	
Канал А					Канал В			Режим 1 Ввод
I/O	I/O	IBF _A	STB _A	INTER _A	STB _B	IBF _B	INTER _B	
Канал А					Канал В			Режим 1 Вывод
OBF _A	ACK _A	I/O	I/O	INTER _A	ACK _B	OBF _B	INTER _B	
Канал А								Режим 2
OBF _A	ACK _A	IBF _A	STB _A	INTER _A	I/O	I/O	I/O	

Темным фоном выделены разряды, устанавливающие внутренний триггер разрешения/запрещения формирования сигнала прерывания, в конкретном режиме для приема и выдачи данных.

Свободные от управляющих сигналов линии канала РС (в таблице они обозначены I/O) могут быть использованы как линии ввода или вывода данных.

Режим работы и направление передачи для каждого канала задается с помощью записи управляющего слова в регистр управления РУС.

Если бит 7 данных содержит 1, то данные передаются в регистр управления и задают режим работы порта в соответствии с табл.1.4.

Если же бит 7=0, то данные являются командой установки/сброса отдельных бит в регистре РС в соответствии с табл.1.5. Это свойство канала РС облегчает программирование в случае, когда биты канала РС используется для управления передачей данных по каналам РА или РВ.

Возможно комбинирование режимов работы для разных каналов. К примеру, канал В может быть запрограммирован на вывод в режим 0 для управления различными выключателями и реле, в то время как канал А может быть запрограммирован в режим 1 для ввода с клавиатуры или перфоленты.

Таблица 1.4

Формат управляющего слова для выбора режима работы

D7=1	D6	D5	D4	D3	D2	D1	D0
Флаг управления 1 = выбор режима	Группа А				Группа В		
	Выбор режима гр.А: 00 = режим 0 01 = режим 1 1x = режим 2	Канал А: 1 = ввод 0 = вывод	Канал С РС4-РС7: 1 = ввод 0 = вывод	Выбор режима гр.В: 0 = режим 0 1 = режим 1	Канал В: 1 = ввод 0 = вывод	Канал С РС3-РС0: 1 = ввод 0 = вывод	

Таблица 1.5

Формат команды сброса/установки для изменения битов канала С

D7=0	D6	D5	D4	D3	D2	D1	D0
Флаг управления 0 = оперирование битами (установка/сброс)	Произвольное состояние			Выбор бита: 0 0 0 = 0 бит 0 0 1 = 1 бит 0 1 0 = 2 бит 0 1 1 = 3 бит 1 0 0 = 4 бит 1 0 1 = 5 бит 1 1 0 = 6 бит 1 1 1 = 7 бит			1 = установить бит 0 = снять бит

Поскольку для режима 2 используются разряды PC7-PC3 порта PC по заранее заданному вполне конкретному предназначению, то управляющее слово для этого режима целесообразно представить как частный случай от общего формата, который определен в табл.1.4. Такой упрощенный формат задан в табл.1.6.

Таблица 1.6

Формат управляющего слова для двунаправленного режима

D7	D6	D5	D4	D3	D2	D1	D0
Флаг управления 1 = выбор режима	Выбор режима: 1 = режим 2	Произвольное состояние			Выбор режима для канала В: 0 = режим 0 1 = режим 1	Канал В: 1 = ввод 0 = вывод	Биты 0..2 канала С: 1 = ввод 0 = вывод

Задание 1.3. Программирование режимов работы адаптера.

Программирование режимов работы порта и обмен данными с ним при раздельной организации памяти осуществляется реализацией команд обмена с внешними устройствами IN и OUT по указанному адресу порта. В соответствие с программной моделью порта таких адресов используется 4.

При этом следует учесть, что пересылка осуществляется из регистров AL или AX. Адресом порта может быть либо числовая константа, если адрес изменяется в пределах 0-255, либо адрес, расположенный в регистре DX.

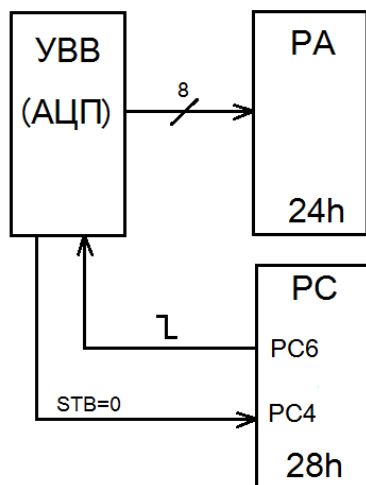
Например: переслать код символа "А" по адресу A0B7h. Т.к. адрес превышает 255, то программа будет выглядеть:

```
MOV     DX,A0B7h
MOV     AL,'A'
OUT     DX,AL.
```

Для анализа взаимодействия с внешним устройством можно рассмотреть некую типовую схему, представленную на рис.1.6. Допустим, внешнее устройство, например, аналого-цифровой преобразователь (АЦП), запускается по сигналу заднего фронта и по готовности выдает нулевой активный сигнал. ИМС порта должна прочесть сформированный код. Адреса портов РА, РВ, РС и РУС, допустим, 24h, 26h, 28h и 2Ah.

В этом случае канал порта, например, РА, должен быть настроен на ввод в режиме 1, тогда активный сигнал окончания преобразования внешней схемы должен восприниматься на выводе PC4 (вход STB для канала А в этом

режиме). По этому сигналу порт зафиксирует код и его можно будет прочитать процессором. Формирование заднего фронта запуска внешней



схемы можно поручить одному из свободных выводов порта РС, который не участвует в управлении режимом 1, например, РС6.

Тогда для задания требуемых режимов (режим 1 порта РА, режим 0 порта РВ, разряд РС6 настроить на вывод) необходимо в соответствии с табл.1.4 сформировать команды:

```
MOV    AL,10110000b
OUT    2Ah,AL.
```

Рис.1.6. Соединение АЦП с портом

Следующим шагом должно быть задание состояния вывода РС6 сначала

1, затем 0 для формирования заднего фронта запуска внешнего устройства. Поэтому режим управления этим разрядом порта РС задается командой сброса/установки по табл.1.5 в адрес РУС микросхемы:

```
MOV    AL,00001101b
OUT    2Ah,AL
MOV    AL,00001100b
OUT    2Ah,AL.
```

Ввод байта от АЦА осуществляется только после получения сигнала окончания преобразования STB=0 на входе РС4, по которому порт его запомнит. Подтверждением прима кода портом является сигнал IBF=1, который должен в режиме 1 появиться на выводе РС5. Только после этого возможно чтения этого байта процессором. Поэтому для согласования временных параметров этих преобразований предполагается цикл ожидания появления 1 подтверждения прима байта из АЦП на выводе РС5:

```
M1: IN    AL,28h
      TEST AL,00100000b
      JZ    M1
      IN    AL,24h.
```

Эту же процедуру можно организовать с использованием сигнала прерывания от канала РА, который снимается в этом режиме с вывода РС3, однако при этом необходимо предварительно выполнить установку триггера разрешения прерывания (разряд РС4, см табл.1.3) с помощью команды сброса/установки по табл.1.5 в адрес РУС микросхемы.

Если необходимо управлять цифро-аналоговым преобразователем (ЦАП), то достаточно использовать свободный порт РВ в режиме 0. Попадание кода в порт РВ сразу вызовет выдачу данных на выход и получение аналогового сигнала.

Задание 1.3.1. Написать программу опроса клавиатуры 8x8 с определением кода нажатой клавиши. Использовать режим 0.

Основная матрица клавиатуры размером 8x8 подключена к каналам А и В, причем канал А должен работать на вывод, а канал В на ввод информации. К выводам С5, С6, С7 канала С подключены дополнительные клавиши управления регистрами. Вывод С4 используется для ввода данных с магнитной ленты. Следовательно, старшая тетрада канала С должна быть запрограммирована на ввод. В младшей тетраде канала С используются только два бита: С0 – вывод данных на магнитную ленту и С3 – индикация раскладки клавиатуры "Рус/Лат". Следовательно, младшая тетрада канала С должна быть запрограммирована на вывод.

Для работы клавиатуры через разряды С4–С7 выполняется сканирование клавиатуры, т.е. выдается код бегущего нуля. При нажатой клавише на одном из контактов С0–С3 появляется код нуля, который обнаруживается при чтении младшей части порта С (опрос клавиатуры).

Данные для программирования можно представить следующим образом:

Адрес		Данные							
A1	A0	D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	0	0	0	1	0	0	1
Регистр управления		Режим программирования	Канал А режим 0		Канал А вывод	Старшая тетрада канала С ввод	Канал В режим 0	Канал В ввод	Младшая тетрада канала С вывод

В отчете представить:

1. Структуру управляющих слов для программирования ППА;
2. Схему алгоритма опроса клавиатуры и обработки данных для получения кода клавиши;
3. Схему подключения клавиатуры к ППА;
4. Текст программы на ассемблере.

Задание 1.3.2. Написать программу выдачи 20 байт через эмулятор порта LPT.

В отчете представить:

1. Диаграмму протокола LPT;
2. Схему алгоритма выдачи данных;
3. Схему управляющих выводов от ППА.

Задание 1.3.3. Написать программу опроса клавиатуры 8x8 с определением кода нажатой клавиши по прерыванию. Использовать режим 0.

В отчете представить данные, аналогичные заданию 1.3.1.

Задание 1.3.4. Написать программу вывода на 4-разрядный индикатор. Использовать режим 0. В рассматриваемой системе индикатор реализован по динамической схеме. Информационный код с порта А зажигает соответствующие сегменты индикатора, а управляющий код с порта В активизирует нужный индикатор, выдавая уровень нуля.

В отчете представить данные, аналогичные заданию 1.3.1.

Вопросы для самопроверки

1. Сформулируйте назначение параллельного интерфейса.
2. Какова программная модель параллельного интерфейса?
3. Каково назначение типовых регистров параллельного интерфейса?
4. Нарисуйте схему подключения микросхемы параллельного интерфейса к шинам МП.
5. Какие существуют режимы работы параллельного интерфейса?
6. Нарисуйте схему подключения внешних устройств к параллельному интерфейсу.
7. Разработайте программу инициализации параллельного интерфейса для основного режима.
8. Разработайте программу инициализации параллельного интерфейса для стробируемого однонаправленного режима.
9. Разработайте программу инициализации параллельного интерфейса для стробируемого двунаправленного режима.

2. АНАЛИЗ РЕЖИМОВ ПРОГРАММИРОВАНИЯ ПОСЛЕДОВАТЕЛЬНОГО ИНТЕРФЕЙСА

Цель

1. Изучить принципы использования последовательного обмена в МПС;
2. Совершенствовать навыки анализа, обобщения и систематизации полученных результатов, навыки составления и оформления отчетных материалов, навыки точного и лаконичного представления докладов на вопросы технического характера.

Учебные вопросы

1. Анализ режимов работы интерфейса;
2. Программная модель и режимы работы интерфейса;
3. Программирование режимов работы интерфейса.

Литература для подготовки к занятию

1. Каган Б.М., Сташин В.В. Основы проектирования микропроцессорных устройств автоматики. – М.: Энергоатомиздат, 1987. – 304 с.
2. Балашов В.Л. и др. Микро- и мини-ЭВМ: Учебное пособие для вузов. - Л.: Энергоатомиздат, 1984, - 376 с.
3. Угрюмов Е.П. Цифровая схемотехника: Учебное пособие для вузов. - СПб.: БХВ-Петербург, 2007. - 800 с. (стр.419-439).
4. <http://literaturki.net/elektronika/mikroprocessory--mikrokrntrollery/250-18-posledovatelnyi-adapter-vv51a>

Содержание отчета

1. Название работы;
2. Для каждого задания: название задания и материал в объеме, указанном в задании.

При больших расстояниях трудно обеспечить помехозащищенность передачи данных параллельным кодом. Кроме того, большое количество линий резко увеличивает стоимость канала. Поэтому передачу обеспечивают последовательным кодом. Этот способ обеспечивает меньшую стоимость, высокую помехозащищенность за счет высоких уровней сигнала и допускает большую протяженность линий передачи. Однако производительность получается ниже, и имеются сложности в интерпретации передаваемых данных.

При передаче необходимо решить вопросы определения начала передачи, моменты времени фиксации очередного бита данных, убедиться в достоверности передачи и определить момент окончания передачи. Применяются протоколы последовательного обмена асинхронный и синхронный.

При **асинхронном последовательном обмене** посылка состоит из стартового бита, фиксирующего начало передачи, самих передаваемых полезных данных (символа, длиной от 5 до 8 бит, начиная с младшего разряда), возможного бита четности (или нечетности), а также стоповых бит, количество которых может быть 1, 1,5 или 2. Такая посылка называется кадром. Ее структура показана на рис.2.1. Каждый символ передается отдельной посылкой.

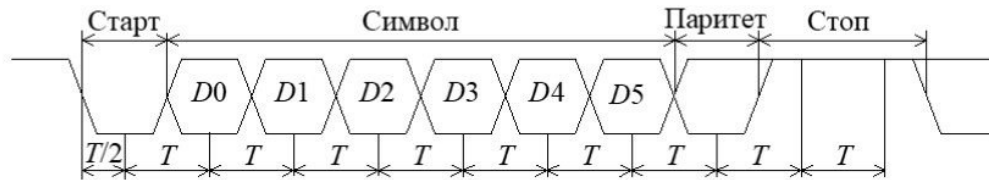


Рис.2.1. Структура кадра при асинхронном обмене

При асинхронном обмене символы передаются по мере их готовности. При отсутствии передачи в линии присутствует высокое напряжение (единица). С появлением нуля начинает работать генератор приемника, который инициирует фиксацию напряжения в линии через половину длительности бита T . Если нуль не подтвердился, то ситуация воспринимается как помеха и оборудование возвращается в исходное состояние. Если нуль зафиксирован, то считается что это старт-бит и происходит последующее чтение информационных битов. После прочтения стоп-бита приемник заканчивает опрос и ждет следующего старт-бита.

После старт-бита все остальные фиксации происходят через время T , чтобы считывание происходило в середине интервала каждого бита, где искажения амплитуды минимальны.

Моменты выдачи и чтения задаются собственными генераторами передатчика и приемника, частоты которых должны быть минимум в 16 раз выше частоты передачи. Поэтому первый отсчет осуществляется через 8 тактов генератора, а последующие - через 16 тактов. Скорости передачи также стандартизированы. Преобразование последовательного кода в параллельный и наоборот происходит на регистрах сдвига. В устройствах имеются отдельно регистр сдвига и регистр хранения кода, из которого код читается микропроцессором.

Для успешной передачи частоты передачи и приема должны быть одинаковы, а интерпретация составляющих формата кадра должны быть едиными для обеих сторон.

Такой формат кадра позволяет фиксировать три типа ошибок передачи:

- ошибку паритета - при обнаружении бита четности, не соответствующего фактическому приему при передаче;
- ошибку кадра (ошибку формата) - при отсутствии стоп-бит в структуре кадра;
- ошибку пропуска (переполнения) - в случае, если процессор не успел прочитать код из регистра хранения при заполненном регистре сдвига к мо-

менту начала приема следующего кода. Такой код (уже третий по счету) будет утрачен. Процессор должен считывать код быстрее, чем заполняется регистр сдвига.

Асинхронная передача имеет большое количество служебных бит (они имеют место для передачи каждого кадра), что снижает эффективную скорость передачи. Однако протокол хорошо отработан и имеется недорогое интерфейсное оборудование.

При **синхронной передаче** структура посылки внешне очень похожа на структуру кадра при асинхронной передаче. Вариант такой посылки показан на рис.2.2. Но каждый прямоугольник на этом рисунке задает не 1 бит, а целый символ (5-8 бит). Длина всех символов одинакова.



Рис.2.2. Структура посылки при синхронном обмене

Приемник при синхронной передаче находится в режиме активного ожидания и постоянно сравнивает принятый символ с символом синхронизации. Например, в системе ASCII таким символом является код 16H=0001.0110. Символов синхронизации может быть 1 или 2 для большей помехозащищенности. Любой символ после символов синхронизации, не являющимся синхросимволом, считается информационным. Окончание передачи фиксируется либо по количеству информационных символов, либо по поступлению символа синхронизации.

Контроль передачи осуществляется по биту четности после каждого символа или вычислением контрольной суммы по модулю для всего сообщения. В последнем случае контрольный символ передается в конце посылки. При отсутствии передачи передатчик формирует синхросимволы до начала следующей информационной посылки.

Синхронная передача обеспечивает более высокую скорость, допускает надежный встроенный механизм обнаружения ошибок, однако требует более сложного и дорогого интерфейсного оборудования.

Задание 2.1 Структурная схема адаптера

Для связи с удаленными источниками информации проще использовать однопроводную линию связи с последовательной передачей по ней информационных битов. На практике нередко используют обычные телефонные

линии связи. Но даже когда применяются специальные линии, то очевидно, что по протяженной линии невозможна передача информации с тактовой частотой микропроцессора (2 МГц).

Для построения интерфейса микропроцессорных систем с использованием последовательной передачи данных используются специализированные интерфейсные БИС, которые получили название программируемых связных адаптеров (ПСА). Они обеспечивают преобразование формы представления информации, согласование протоколов обмена данными внутри системы и вне ее, согласование скоростей передачи и тактовой частоты процессора.

Микросхема программируемого связного адаптера KP580BB51 может осуществлять выдачу байта данных, представленного параллельным кодом, в последовательном формате с соответствующими битами обрамления: стартовым, стоповым и контрольными. Она же может осуществлять прием информации, представленной последовательным кодом, и ее преобразование в стандартную параллельную форму. Структурная схема ПСА и его УГО представлены на рис.2.3.

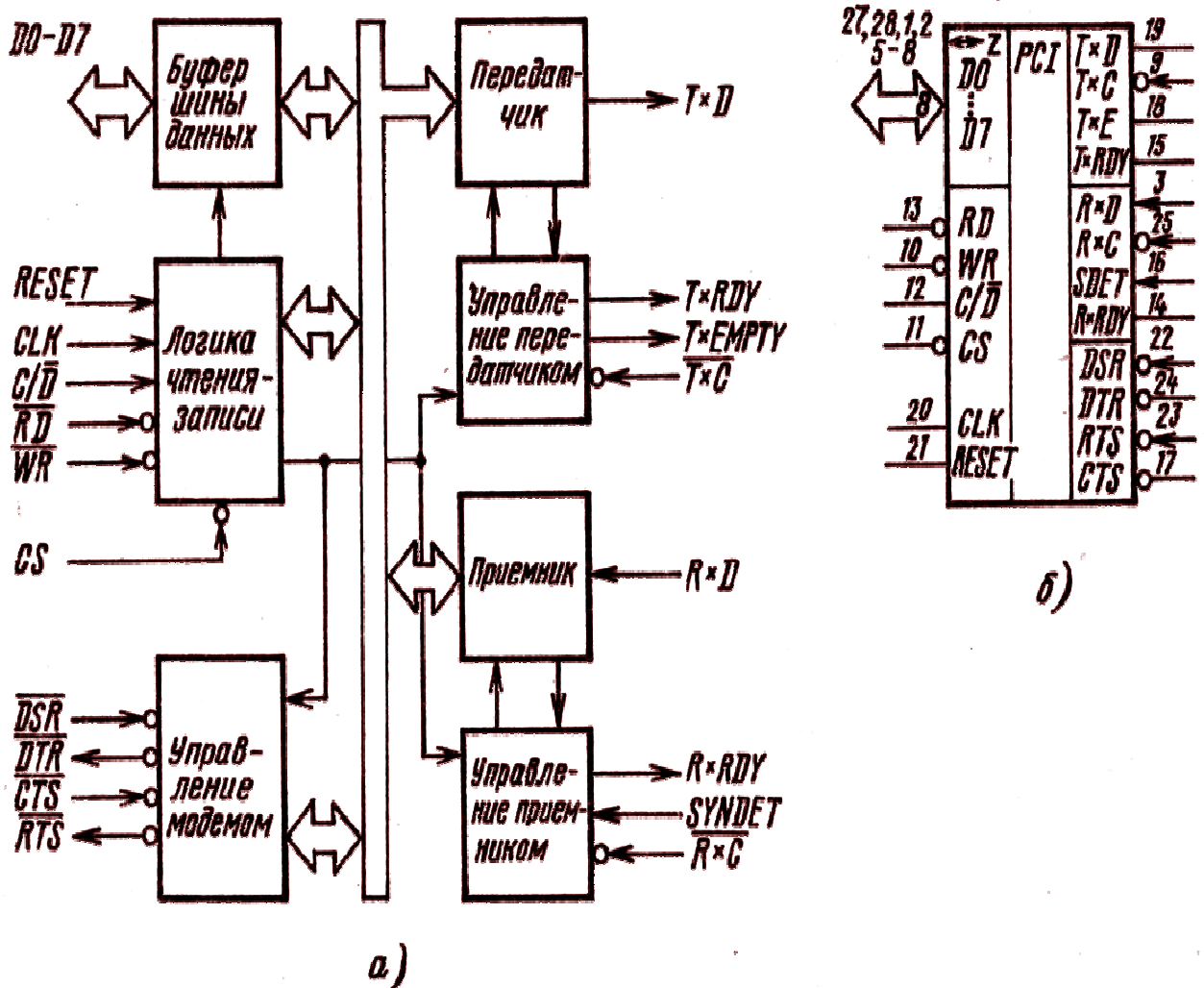


Рис.2.3. Структурная схема (а) и УГО периферийного связного адаптера (ПСА)

По принятой классификации такая микросхема в английской транскрипции соответствует USART - Universal Synchronous/Asynchronous Receiver/Transmitter - универсальный синхронно/асинхронный приемопередатчик.

На рис.2.4 с учетом использования только четных адресов показано подключение ПСА к МП.

В адаптер подаются шесть входных сигналов:

RESET (сброс) — высокий активный сигнал сброса с минимальной длительностью 6 периодов синхронизации. После действия этого сигнала адаптер переводится в «холостой» режим и остается в нем до загрузки управляющих слов. В этот режим его можно перевести также программно;

CLK (синхронизация), подключается ко второй фазе системного генератора синхронизации. Частота сигналов CLK должна быть минимум в 30 раз больше максимальной скорости обмена данными;

RD (считывание) — нулевой активный сигнал, инициирующий передачу данных или состояния из адаптера на шину данных МП;

WR (запись) — нулевой активный сигнал загрузки информации с шины данных МП в адресуемый регистр адаптера;

C/D (Control/Data - управление/данные) — сигнал идентификации передачи данных или управляющих слов: 1 - передается управляющее слово, 0 - передаются данные;

CS (выбор кристалла) — нулевой активный сигнал, разрешающий связь между адаптером с шиной данных. Формируется дешифратором адреса по сигналам адресной шины процессора.

Функции ПСА и комбинации сигналов управления показаны в табл.2.1.

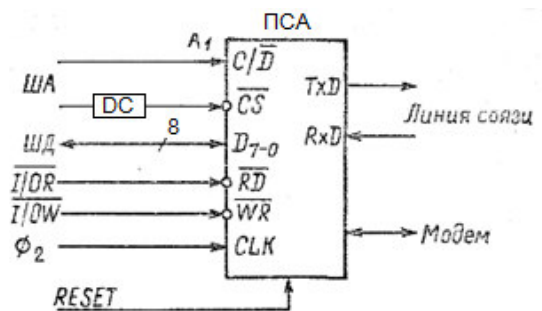


Рис.2.4. Подключение ПСА к шинам

Таблица 2.1

Функции ПСА и сигналы управления

CS	C/D	RD	WR	Функции
0	0	0	1	Считывание байта из ПСА
0	1	0	1	Считывание слова состояния из ПСА
0	0	1	0	Запись байта в ПСА
0	1	1	0	Запись управляющего слова в ПСА
0	x	1	1	Отключение ПСА от шины данных
1	x	x	x	

Состояние $RD = 0$ и $WR = 0$ является запрещенным и дает непредсказуемый результат. Схема управления адаптера содержит собственные цепи синхронизации, поэтому сигналы RD и WR могут быть несинхронными со входом CLK .

Узел передатчика со схемой управления выполняет все функции, связанные с передачей последовательных данных: воспринимает параллельные коды символов от процессора, автоматически вводит необходимые служебные биты и символы синхронизации и выдает последовательный поток на выход TxD . С этим узлом взаимодействуют следующие внешние сигналы:

TxD (выход передатчика) — выходная линия, по которой действуют сигналы передаваемых данных;

TxC (синхронизация передатчика) — входной сигнал, управляющий скоростью передачи данных. Спад TxC "выдвигает" последовательные биты на выход TxD . В синхронном режиме скорость передачи соответствует частоте сигнала TxC , а в асинхронном режиме программируется как 1, $1/16$ или $1/64$ частоты сигнала TxC ;

TxE или $TxEmpty$ (пустой передатчик) — единичный активный выходной сигнал, обозначающий отсутствие в адаптере в буферном регистре символа для передачи. Его можно использовать для идентификации в полудуплексной связи окончания передачи и коммутации линии на прием. В синхронном режиме H -уровень TxE указывает, что символ вовремя не загружен в адаптер и в качестве "заполнителей" автоматически передаются символы синхронизации. Его можно проверить при считывании слова состояния адаптера (бит 2). Сигнал TxE сбрасывается при загрузке символа в адаптер;

$TxRDY$ (готовность передатчика) — H -активный выходной сигнал, определяющий готовность передатчика к восприятию символа от МП. Используется для прерывания процессора или проверяется при считывании состояния адаптера (бит 0). При загрузке в передатчик нового символа сигнал $TxRDY$ сбрасывается.

Узел приемника с автономной схемой управления воспринимает последовательные данные со входа RxD , преобразует их в параллельный формат, контролирует и исключает служебные биты и символы синхронизации, а затем передает готовый символ в процессор. К приемнику относятся следующие внешние сигналы:

RxD (вход приемника) — входная линия, по которой передаются сигналы принимаемых последовательных данных;

RxC (синхронизация приемника) — входной сигнал, определяющий скорость приема символов. В синхронном режиме скорость приема соответствует частоте сигнала RxC , а в асинхронном частота RxC кратна скорости приема (ниже с коэффициентом 1, $1/16$, $1/64$, что обеспечивается делителями частоты). Данные вводятся в адаптер по фронту RxC . Обычно передача и прием осуществляются с одинаковой скоростью, поэтому сигналы TxC и RxC

должны иметь одинаковую частоту. Для этого они объединяются и подключаются к одному генератору синхронизации;

RxRDY (готовность приемника) — выходной сигнал, Н-уровень которого свидетельствует о наличии в адаптере принятого символа и готовности его для чтения. Его можно подключить на вход прерывания процессора или проверить значение при считывании слова состояния адаптера (бит 1). Когда процессор вводит символ из адаптера, сигнал RxRDY сбрасывается;

SYNDET (Detect/Break Detect - обнаружение синхронизации) — Н-активный сигнал синхронного режима, который может быть запрограммирован как выходной или входной. Если он запрограммирован как выходной, то при обнаружении символа синхронизации (SYN) на выходе SYNDET формируется высокий уровень в момент времени, соответствующий середине последнего бита (в случае работы с двумя символами синхронизации это относится ко второму сигналу синхронизации SYN2). При считывании состояния адаптера сигнал сбрасывается. Когда сигнал SYNDET определен как входной при внешней синхронизации, подача на него высокого уровня фиксирует момент начала приема символа.

В асинхронном режиме вывод формирует высокий потенциал в случае обнаружения нулей на месте стоп-битов в двух последовательных посылках. Его называют сигналом обнаружения разрыва, и он может быть выявлен по состоянию бита 6 слова состояния адаптера. Сигнал на выходе обнуляется при сбросе или при появлении стоп-битов при приеме очередных символов.

Адаптер обслуживает четыре линии управления модемом, сигналы которых при необходимости могут использоваться и для других функций других устройств (терминалов). Выходными сигналами можно управлять с помощью слов-приказов, а входные сигналы считывать в слове состояния адаптера:

DSR (Data Set Ready - готовность данных модема) — входной нулевой сигнал от терминала, означающий наличие данных у терминала и его готовность к передаче в адаптер (бит 7 в слове состояния адаптера);

DTR (Data Terminal Ready - готовность терминала) — выходной нулевой сигнал (активизируется битом 1 слова приказа адаптера). Является реакцией адаптера на запрос DSR от терминала при наличии разрешения обмена с приемником модема RxEN (бит 2 слова приказа ПСА). По этому сигналу модем передает данные на вход адаптера;

RTS (Request to Send - запрос передачи) — выходной нулевой сигнал (активизируется битом 5 слова приказа адаптера). Является запросом готовности приемника модема принять данные для передачи. Формируется при наличии разрешения обмена с передатчиком модема TxEN (бит 0 слова приказа адаптера);

CTS (Clear to Send - готовность для передачи) - входной нулевой сигнал готовности приемника терминала, разрешающий адаптеру передавать данные для передачи. Обычно генерируется модемом в ответ на RTS при наличии разрешения работы на передачу TxEN (бит 0 слова приказа адаптера).

Программируемый связной адаптер KP580BB51 обеспечивает полный дуплексный режим связи, детектирование стартовых посылок, может использовать различные способы представления стоповых бит, детектирование ошибок четности и ошибок в формате. Он может использоваться для работы как в цепях асинхронной последовательной связи, так и в цепях синхронного обмена. При традиционных методах создания интерфейса цифровых систем подобный универсализм потребовал бы очень большого числа управляющих сигналов, что невозможно реализовать из-за ограниченного числа внешних выводов БИС процессора.

В отчете представить:

1. Структурную схему ПСА;
2. Схему алгоритма обмена;
3. Кратко назначение внешних связей адаптера.

Задание 2.2. Программная модель и режимы работы интерфейса

ПСА в асинхронном режиме обмена.

Для организации последовательных асинхронных передач сопрягаемые системы соединяются через ПСА, установленные на передающей и приемной стороне. Передающая и принимающая подсистемы имеют тактовые генераторы, работающие с одинаковой частотой. Каждый символ передаваемого сообщения заключается в «рамку», которая образуется стартовым и стоповым битами. Приемник автоматически синхронизируется стартовым битом так, чтобы можно было производить отбор данных в моменты, соответствующие середине каждого бита. В результате, если возникнет небольшой дрейф частоты тактовых генераторов, это не приведет к срыву передачи.

На передающей стороне в режиме асинхронной передачи каждый символ, представленный последовательным кодом, выдвигается из буфера ПСА в линию связи поразрядно со скоростью, которая равняется $1/16$ или $1/64$ максимальной скорости передачи, определяемой частотой синхронизации. Естественно, что перед началом передачи в ПСА уже сформировано полное сообщение (передаваемый символ, снабжен стартовым и стоповым битами и, если требуется, битом контроля четности-нечетности).

На приемной стороне на вход нормально подан высокий уровень. Переход сигнала на этом входе в состояние 0 оповещает ПСА о том, что поступил стартовый бит. При этом запускается счетчик бит и внутренний генератор опорных сигналов, который выдает стробирующий сигнал приблизительно в середине каждого бита. Принимаемый символ последовательно, разряд за разрядом поступает в буфер ПСА. Если на том месте, где по указанию внутреннего счетчика должен находиться стоповый бит, оказывается низкий уровень, то устанавливается в 1 триггер ошибки в формате. Нормальный стоповый бит сигнализирует о конце символа, гасит внутренний счетчик

бит и генерирует сигнал, который сообщает МП о том, что символ готов к загрузке в аккумулятор.

Асинхронный принцип системного обмена нашел широкое распространение в микропроцессорных системах благодаря совместимости с линией связи, в частности с телетайпом. Однако область применения этого способа передач ограничена относительно низкой скоростью передачи данных — 9600 бит/с. Для работы с высокоскоростными устройствами более подходит способ последовательной синхронной передачи.

ПСА в синхронном режиме обмена.

В синхронном режиме данные передаются не по одному символу, а целыми массивами слов. Для синхронизации запуска приемника с приемом массива используется не единичный бит, а одно или два слова — символы синхронизации. Передатчик в синхронном режиме обмена работает постоянно и выдает в линию данные в последовательном коде. При отсутствии (даже временном) данных передатчик непрерывно выдает в линию символы синхронизации. Приемник перед началом работы находится в режиме активного ожидания. В это время он побитно принимает поступающую информацию и проверяет ее на совпадение с кодом синхронизации. Символом синхронизации осуществляется адресация того или иного приемника. Совпадение свидетельствует о начале информационного массива. Сигналом совпадения синхронизируется запуск приемника, который начинает прием данных и их преобразование в параллельный формат. Один передатчик может работать на несколько приемников. Новые символы синхронизации, вставляемые передатчиком в поток данных, позволяют повторно осуществлять процедуру синхронизации.

В режиме синхронного приема символы синхронизации могут быть как внутренними, так и внешними. Если программное обеспечение микропроцессора загружает в ПСА слово синхронизации, которое определяет внутренний режим, то приемник переходит в режим активного ожидания. В этом режиме приемник ожидает заполнения буфера. Как только буфер заполнится, его содержимое интерпретируется адаптером как символ синхронизации. Если ПСА запрограммирован на два символа синхронизации, то и следующий символ, загруженный в буфер, воспринимается как символ синхронизации. Затем приемник переходит из режима ожидания в уже описанный режим синхронного приема.

При внешней синхронизации прием данных начинается в момент поступления на вход CLK внешнего сигнала, отмечающего начало посылки. В этом случае канал связи должен иметь дополнительную физическую линию внешней синхронизации.

Контроль четности-нечетности и переполнения в синхронном режиме осуществляется так же, как в асинхронном.

С точки зрения скорости передачи данных для информационных массивов малой размерности (менее 100 бит) более выгодным является асинхронный режим, а для массивов большой размерности — синхронный. Пер-

вый обеспечивает максимальную скорость передачи данных до 9,6, а второй — до 56К бит/с.

Программная модель ПСА основывается всего на двух адресах, которые задаются сигналом на входе C/D. Однако с учетом сигналов чтения RD и записи WR обращение осуществляется к регистру входных данных, выходных данных, регистру команд и регистру слова состояния ПСА. Комбинации сигналов, участвующих в адресации были указаны ранее в табл.2.1.

Режим работы ПСА задается после аппаратного сброса по входу RE-SET или программного сброса по текущему управляющему слову (слову приказа). Формат управляющего слова режима (начального управляющего слова) представлен на рис.2.5.

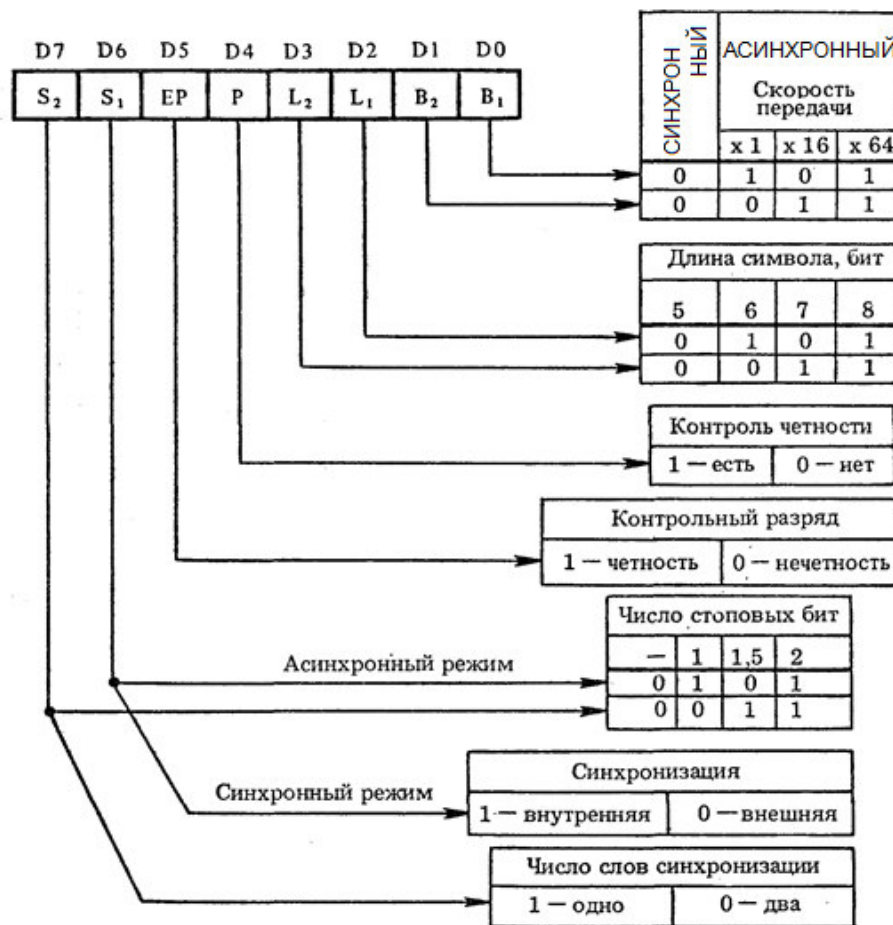


Рис.2.5. Формат управляющего слова режима ПСА

Следует обратить внимание, что интерпретация разрядов D6, D7 зависит от задаваемого режима работы, который определяется разрядами D1, D0.

В отчете представить:

1. Управляющие слова для задания асинхронного и синхронного режимов работы ПСА;

2. Сравнительный анализ этих режимов.

Задание 2.3. Программирование режимов работы интерфейса

Для программирования адаптера необходимо загрузить несколько управляющих слов в определенном порядке, определяющих скорость передачи, длину символа, число стоповых бит, режим работы и условия контроля (четный или нечетный паритет). Кроме того, для синхронного режима требуется определить внешнюю или внутреннюю синхронизацию, а также символ (или два символа) синхронизации. После программирования адаптер готов выполнять свои функции.

Управляющие слова имеют два формата: слово режима (см. рис.2.5) и слово приказа.

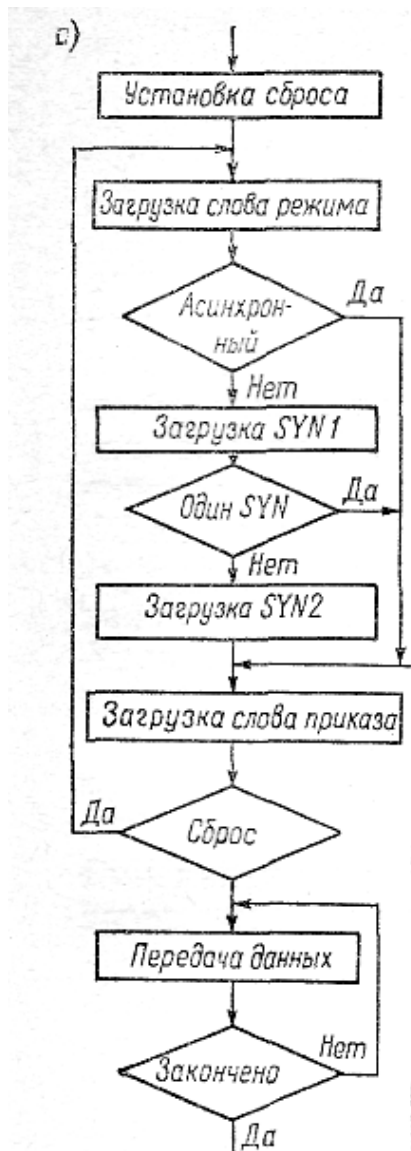


Рис.2.6. Схема алгоритма программирования ПСА

На входе TxRDY формируется высокий уровень, сигнализирующий о готовности передатчика принять символ для передачи. Приемник сдвигает последовательные биты с линии RxD и после приема всего символа генерирует сигнал RxRDY. Адаптер не начинает передачу до тех пор, пока не установлен бит разрешения передатчика TxEN в слове приказа. На выходе TxD передатчика до сброса формируется высокий уровень.

Следует подчеркнуть, что правильная загрузка нескольких регистров в ПСА без указания их персональных адресов обеспечивается жесткой последовательностью записи. Поэтому порядок записи слов очень важен.

Программирование начинается с записи управляющего **слова режима**, задаваемого после операции аппаратного или программного сброса, задаваемого специальным битом в слове приказа.

Схема алгоритма программирования приведена на рис.2.6. Слово режима задает общие рабочие характеристики адаптера и обязательно загружается первым, так как оно необходимо для коммутаций схем прибора.

После слова режима загружаются

один или два символа синхронизации SYN, если был определен синхронный режим с соответствующим количеством синхросимволов и внутренней синхронизацией в управляющем слове режима. Синхросимволы загружаются по адресу регистра режима. Синхросимволы сохраняются в специальных регистрах ПСА и используются для адресации конкретного адаптера. При задании асинхронного обмена синхросимволы не задаются.

Далее в адаптер загружается **слово приказа**, определяющее его конкретные действия в соответствии со словом режима. После этого обмениваются данными. Новое слово приказа можно загружать в ПСА в любое время в соответствии с логикой обмена.

Загружаемое **слово приказа** задает текущую операцию адаптера. В слове приказа задаются разрешение передачи или приема, сброс ошибок, управление модемом и некоторая другая информация. Смысл отдельных битов слова приказа показан на рис.2.7. Установка единицы в соответствующий бит обеспечивает реализацию конкретного функционала или формирование активного сигнала на соответствующем выводе ПСА.

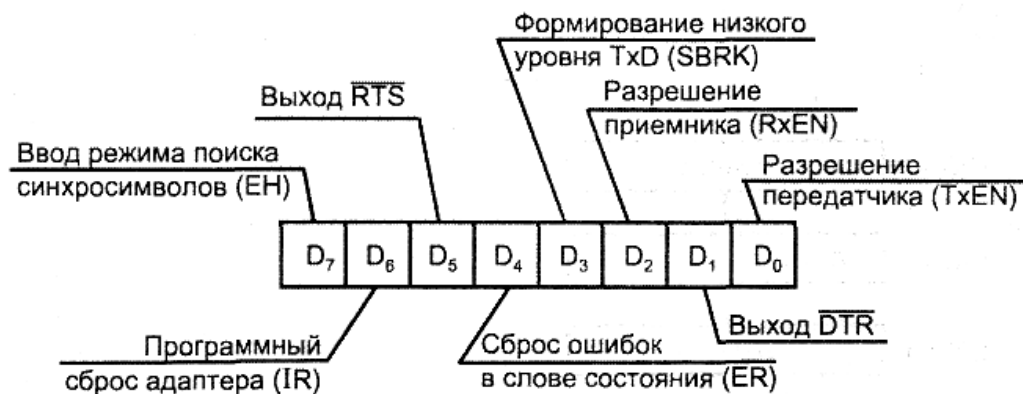


Рис.2.7. Формат слова приказа

При организации последовательного интерфейса возникает необходимость проанализировать состояние адаптера и текущего результата обмена данными. Состояние адаптера можно считывать в любой момент времени посредством выполнения команды ввода данных от внешнего устройства IN по адресу регистра командного слова (обеспечивающего формирование сигнала C/D=1).

Формат слова состояния адаптера представлен на рис.2.8.

Все флажки ошибок сбрасываются, когда бит D₄ слова приказа установлен в единицу. Следует специально подчеркнуть, что возникновение любого ошибочного условия не останавливает работу адаптера.

Организация обмена с адаптером может предусматривать анализ слова состояния адаптера и обмен по готовности. Второй вариант предполагает использование сигналов прерывания, роль которых выполняют сигналы готовности адаптера TxRDY и RxRDY.

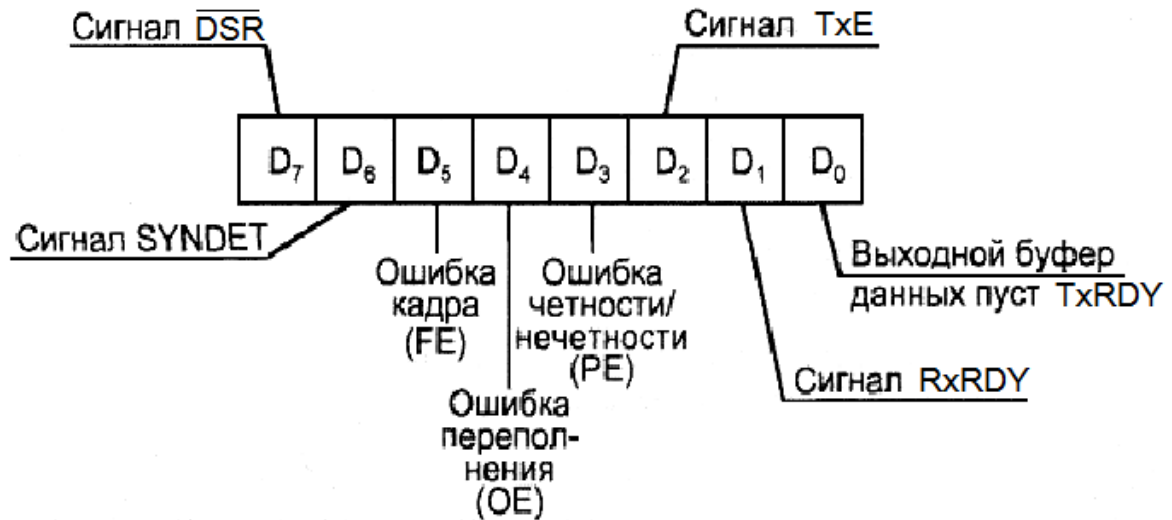


Рис.2.8. Формат слова состояния ПСА

Пример 1:

Запрограммировать асинхронный режим работы ПСА на частоте синхронизации в 64 раза выше скорости передачи. Передавать необходимо 1 байт с признаком нечетности и 2 стоп-битами. Адреса регистра управления и регистра данных соответственно 2Ch и 2Ah.

```
MOV    AL,11011111b    ;Асинхронный, 8 бит, нечетность, x64
OUT     2Ch,AL          ;передача слова режима
MOV     AL,00110001     ;Слово приказа: разрешена передача,
OUT     2Ch,AL          ;запрос готовности приемника
OUT     2Ah,55h         ;Выдача байта из чередующихся нулей и
                        ;единиц для передачи
```

Пример 2:

Перевести ПСА в синхронный режим и осуществить поиск двух символов синхронизации. Адресные данные ПСА и формат информационной посылки аналогичны примеру 1.

```
MOV     AL,00011100b    ;Синхронный, 8 бит, нечетность
OUT     2Ch,AL          ;передача слова режима
MOV     AL,00011000b    ;Слово-приказ: сброс ошибок, 0 на TxD
OUT     2Ch,AL
MOV     AL,10010100     ;Слово-приказ: поиск синхроимпульса,
OUT     51h,AL          ;сброс ошибок, разрешение приема
```

Пример 3:

Ввести из ПСА 40 символов в ОЗУ по адресу BUFER. Адресные данные ПСА и формат информационной посылки аналогичны примеру 1. Предусмотреть контроль ошибок приема.

	MOV	AL,00010101b	;Слово-приказ: сброс ошибок, разрешен
	OUT	2Ch,AL	;прием и передача
	MOV	DI,0	;Регистр индекса = 0
	MOV	CX,40	;Счетчик цикла =40
M1:	IN	AL,2Ch	;Ввод слова состояния
	TEST	AL,02h	;Контроль сигнала готовности RxRDY
	JZ	M1	;Ожидание готовности
	MOV	BUFER[DI],AL	;В память по адресу BUFER+DI
	INC	DI	;Инкремент индекса
	IN	AL,2Ch	;Ввод слова состояния
	TEST	AL,38h	;Контроль наличия ошибок
	IN	AL,2Ah	;Чтение очередного байта
	OUT	51h,AL	; сброс ошибок, разрешение приема
	JNZ	OSHIBKA	;Переход на анализ ошибок
	LOOP	M1	;Повтор цикла, если CX<>0
	JMP	EXIT	;Окончание процедуры

Чтение символа из порта выполняется по мере готовности (RxRDY=1). После чтения символа в процессор готовность сбрасывается и требуется ожидание нового RxRDY=1. Если читаемый символ в ПСА длиной менее 1 байта, то старшие разряды пересылаемого байта заполнятся нулями.

Задание 2.3.1. Написать программу выдачи 20 байт в синхронном режиме.

В отчете представить:

1. Формат управляющего слова режима для задания режима ПСА;
2. Формат слов приказа;
3. Схему алгоритма обмена;
4. Программу на языке ассемблера.

Задание 2.3.2. Написать программу приема 40 байт в асинхронном режиме.

В отчете представить:

1. Формат управляющего слова режима для задания режима ПСА;
2. Формат слов приказа;
3. Схему алгоритма обмена;
4. Программу на языке ассемблера.

Вопросы для самопроверки

1. Сформулируйте назначение последовательного интерфейса.
2. Какие коды называются последовательными и параллельными?
3. Какова программная модель последовательного интерфейса?

4. Каково назначение типовых регистров последовательного интерфейса?
5. Нарисуйте схему подключения микросхемы последовательного интерфейса к шинам МП.
6. Какие существуют режимы работы последовательного интерфейса?
7. Нарисуйте схему подключения внешних устройств к последовательному интерфейсу.
8. Разработайте программу инициализации последовательного интерфейса для работы в асинхронном режиме.
9. Разработайте программу инициализации последовательного интерфейса для работы в режиме внутренней синхронизации.
10. Разработайте программу инициализации последовательного интерфейса для работы в режиме внешней синхронизации.

3. ПРОГРАММИРОВАНИЕ ИНТЕРВАЛЬНОГО ТАЙМЕРА

Цель

1. Изучить принципы использования таймеров в МПС;
2. Совершенствовать навыки анализа, обобщения и систематизации полученных результатов, навыки составления и оформления отчетных материалов, навыки точного и лаконичного представления докладов на вопросы технического характера.

Учебные вопросы

1. Структурная схема таймера и режимы работы;
2. Программная модель таймера;
3. Программирование режимов работы таймера.

Литература для подготовки к занятию

1. Каган Б.М., Сташин В.В. Основы проектирования микропроцессорных устройств автоматики. – М.: Энергоатомиздат, 1987. – 304 с. (с.192-199).
2. Алексенко А.Г., Галицин А.А., Иванников А.Д. Проектирование радио-электронной аппаратуры на микропроцессорах. – М.: Радио и связь, 1984. – 272 с. (с.65-72).
3. <http://www.computer-museum.ru/technlgy/i8253.htm>
4. Джордейн Р. Справочник программиста персональных компьютеров типа IBM PC, XT, и AT. - М.: Финансы и статистика, 1992. - 554с.

Содержание отчета

1. Название работы;
2. Для каждого задания: название задания и материал в объеме, указанном в задании.

Задание 3.1. Структурная схема таймера и режимы работы

Таймер K580ВИ53 (аналог i8253) предназначен для задания временных интервалов МП систем, подсчета числа внешних событий и формирования сигналов с разными временными и частотными характеристиками. Он избавляет процессор от программного выполнения этих функций, тем самым высвобождая ценное процессорное время для выполнения основных задач.

Условное графическое обозначение таймера приведено на рис.3.1. Содержит три независимых канала на основе вычитающих счетчиков с максимальной частотой работы 2 МГц. Шины таймера имеют z-состояние (трех-стабильный буфер данных).

Основные элементы таймера представлены на рис.3.2. Каждый канал содержит регистры:

- состояния канала (8 разрядов);
- управляющего слова RSW (8 разрядов) - задает режим работы канала;

- буферный регистр (16 разрядов) для запоминания текущего содержимого счетчика без остановки процесса счета. После запоминания буферный регистр доступен программе для чтения;

- счетчик (16 разрядов) с предустановкой, работающий в режиме вычитания по заднему фронту сигнала CLK при условии, что на вход GATE установлен уровень логической 1;

- регистр констант пересчета (16 разрядов) для сохранения констант для повторной загрузки в счетчик, если это требуется в текущем режиме работы таймера.

В зависимости от режима работы таймера при достижении счетчиком нуля тем или иным образом изменяется выходной сигнал канала OUT.

Назначение выводов таймера показано в табл.3.1.

Возможны шесть режимов работы таймера. Они разделяются на три типа:

Режимы 0, 4 - однократное выполнение функций.

Режимы 1, 5 - работа с перезапуском.

Режимы 2, 3 - работа с автозагрузкой.

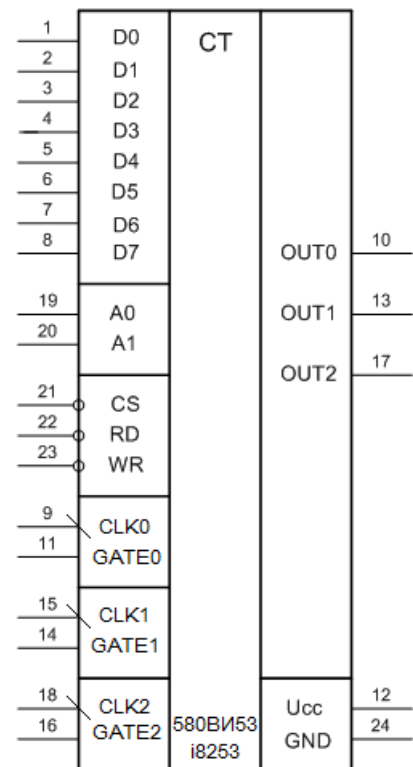


Рис.3.1. УГО KP580BI53

Таблица 3.1

Назначение выводов таймера

D0...D7	Двунаправленная шина данных. Предназначена для передачи и приема данных процессором, а также передачи управляющих команд и слова состояния
CS	Chip Select. Выбор микросхемы. Низкий уровень инициирует обмен между процессором и ППА
RD	Read. Чтение. Низкий уровень на этом входе позволяет ППА передать данные или слово состояния процессору
WR	Write. Запись. Низкий уровень на этом входе позволяет ППА принять данные или управляющую команду
A0, A1	Адресные входы. Позволяют выбрать один из каналов (0, 1, 2) или регистр управляющего слова и состояния
CLK	Входы тактовых импульсов каждого канала. Фронт срабатывания - задний
GATE	Вход разрешения счета. Особенности зависят от конкретного режима. Активный разрешающий сигнал - 1
OUT	Выход счетчика

В режиме однократного выполнения функций перед началом счета содержимое регистра констант пересчета переписывается в счетчик по сигналу CLK. В дальнейшем счетчик работает в режиме вычитания. Процесс счета можно приостановить сигналом GATE=0. Для повторения выполнения функции необходимо повторное программирование таймера.

При работе с перезапуском не требуется повторного программирования таймера для выполнения той же функции. По заднему фронту сигнала GATE значение константы вновь переписывается в счетчик при любом его текущем состоянии.

В режиме автозагрузки регистр констант автоматически переписывается в счетчик после завершения счета. Сигнал на выходе OUT появляется только при наличии на входе GATE уровня логической 1. Этот режим используется для создания программируемых импульсных генераторов и генераторов прямоугольных импульсов (меандра).

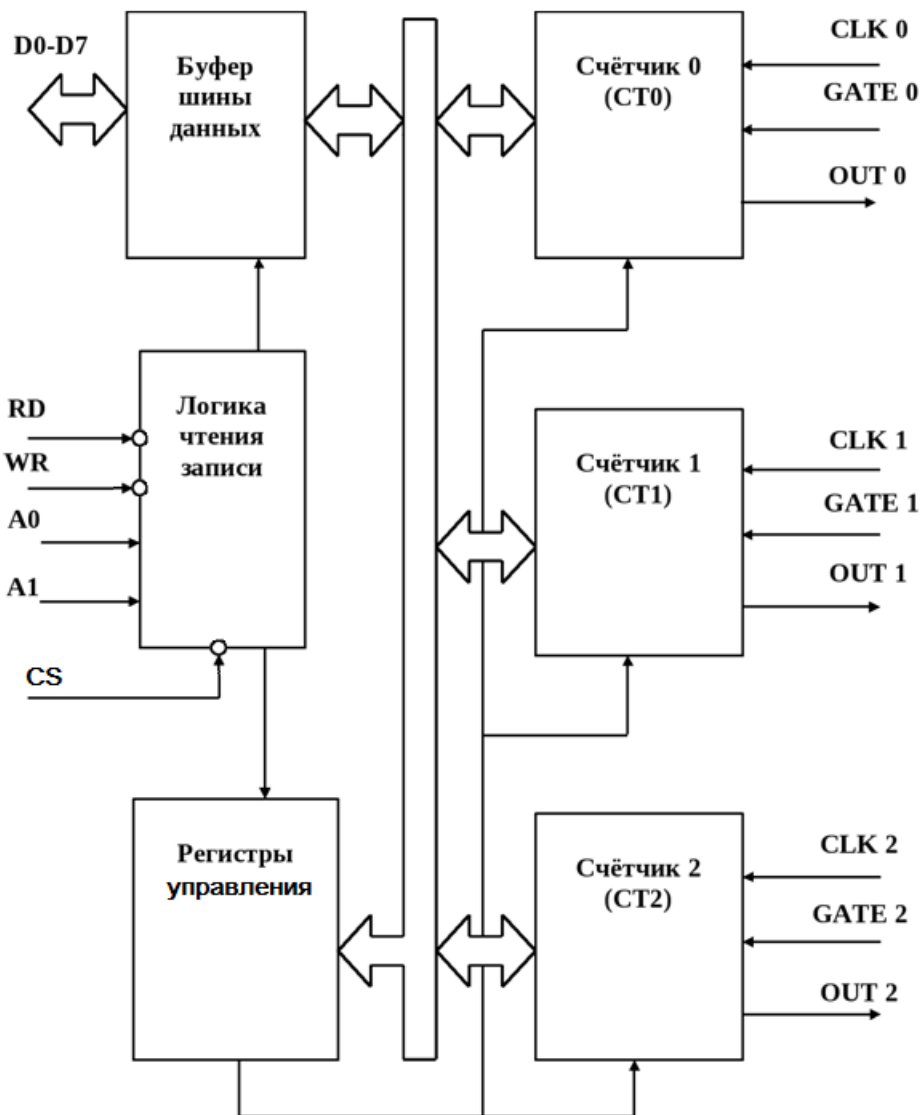
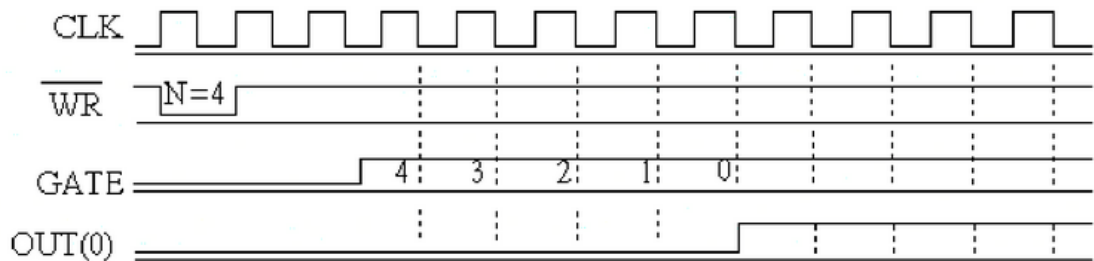


Рис.3.2. Структурная схема таймера

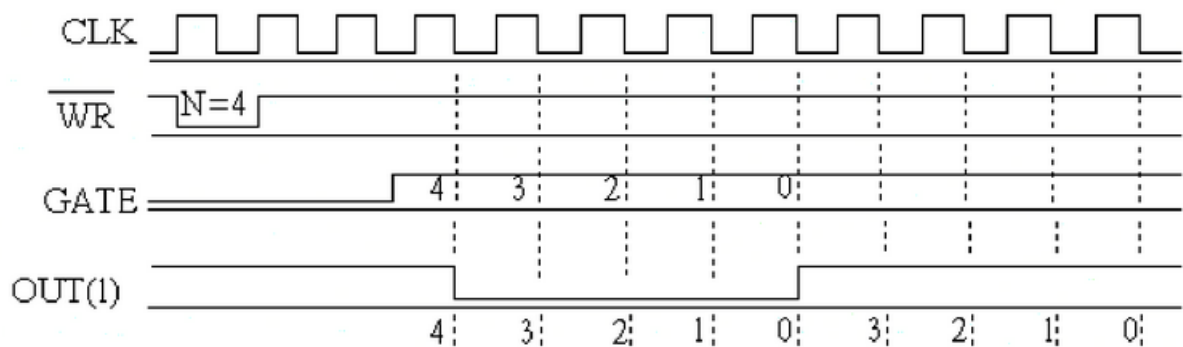
Каналы таймера могут работать в следующих режимах:

Режим 0 (000) прерывание по окончании счета (программируемая задержка). На выходе канала появляется уровень "0" сразу же после задания режима работы. После загрузки числа в счетчик канала выход остается в "0" и счетчик начинает считать, если на входе разрешения GATE установлен уровень "1". После достижения счетчиком нуля, на выходе устанавливается уровень "1" и остается до тех пор, пока канал не будет заново запрограммирован. Однако счетчик при этом не останавливается. Задержка отсчитывается от заднего фронта первого импульса CLK. Если во время счета сигнал GATE= 0, счет приостанавливается.



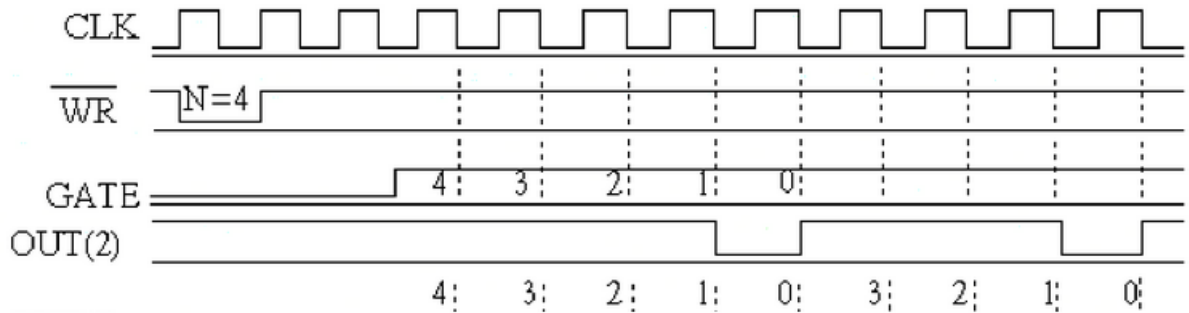
Режим 1 (001) ждущий мультивибратор с программно-устанавливаемой длительностью сигнала N тактовых периодов. В этом режиме выход канала OUT устанавливается в уровень "0" после первого заднего фронта тактового сигнала, следующего за передним фронтом на управляющем входе GATE. Одновременно начинается счет, а при достижении нуля счетчика на выходе устанавливается уровень "1". Мультивибратор автоматически перезапускается по каждому переднему фронту сигнала GATE.

Перезагрузка счетчика во время счёта не изменяет длительности текущего выходного импульса.

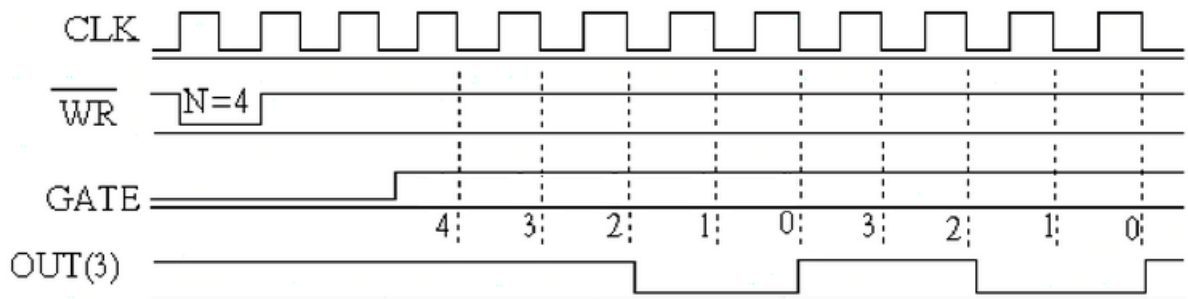


Режим 2 (x10) генератор тактовых сигналов (делитель частоты). На выходе канала через число периодов тактовой частоты, записанное в счетчике канала, появляется уровень "0" длительностью в один период тактовой частоты. Начальная установка счетчика N определяет коэффициент деления.

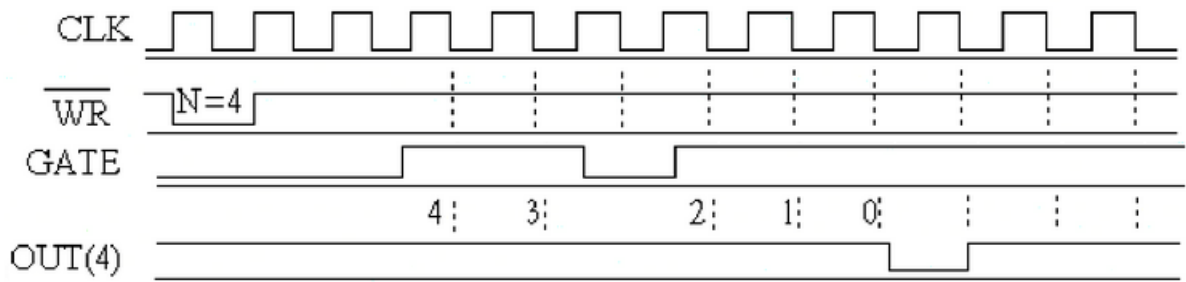
Перезагрузка счетчика не приводит к изменению длительности текущего периода, но влияет на длительность последующего. При подаче на управляющий вход GATE уровня «0» выход OUT устанавливается в состояние «1». Передний фронт GATE запускает счетчик из начального состояния и может служить для аппаратной синхронизации счетчика.



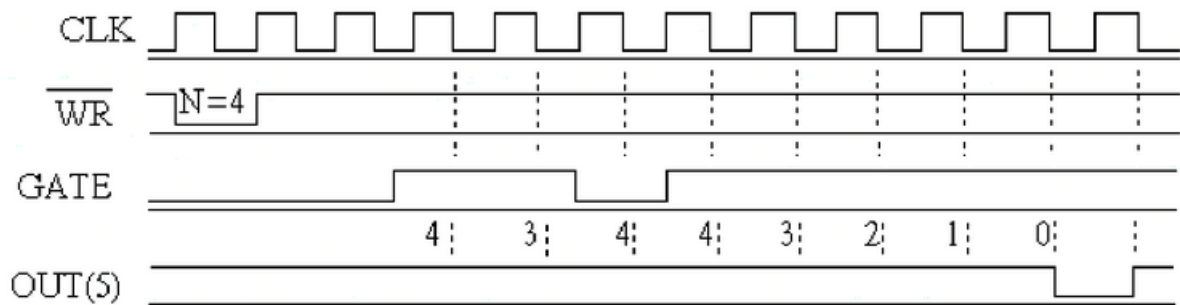
Режим 3 (x11) генератор прямоугольных импульсов с программно-управляемым периодом. На выходе канала будет уровень "1" в течение первой половины интервала времени, определяемого числом N в счетчике, и уровень "0" в течение второй половины. При нечетном N длительность сигнала "1" на один такт больше, чем для сигнала "0". При перезагрузке счетчика новое значение скажется на результате работы только при переходе OUT в другое состояние. В остальном режим подобен предыдущему.



Режим 4 (100) программно управляемая задержка строга (нулевого импульса). С момента загрузки кода предустановки в счетчик канала генерируется выходной сигнал "1" длительностью N периодов тактового сигнала CLK. После завершения счета устанавливается выходной сигнал "0" на один период CLK. Перезагрузка счетчика во время счёта приводит к перезапуску счетчика. Появление уровня "0" на входе GATE приостанавливает счёт.



Режим 5 (101) аппаратная задержка строка. Счётчик начинает работать только по переднему фронту сигнала на входе GATE. Новый фронт GATE перезапускает счёт с начального значения. В остальном режим подобен предыдущему. Загрузка в счетчик нового значения N в процессе счета не влияет на продолжительность текущего цикла, но новый цикл будет соответствовать новому значению N.



В отчете представить:

1. Структурную схему таймера;
2. Назначение входов и выходов устройства;
3. Данные о режимах работы таймера. Представить в виде таблицы: режим, название, краткое пояснение работы.

Задание 3.2. Программная модель таймера

Программно таймер представляет собой четыре порта, которые адресуются адресными входами A1 и A0. Обращение осуществляется по адресам:

A1	A0	Обращение
0	0	CT0. 16 разрядов
0	1	CT1. 16 разрядов
1	0	CT2. 16 разрядов
1	1	RCW Регистр управляющего слова. 8 разрядов

Каждый канал таймера программируется индивидуально в соответствии с управляющим словом. Структура управляющего слова представлена на рис.3.3.

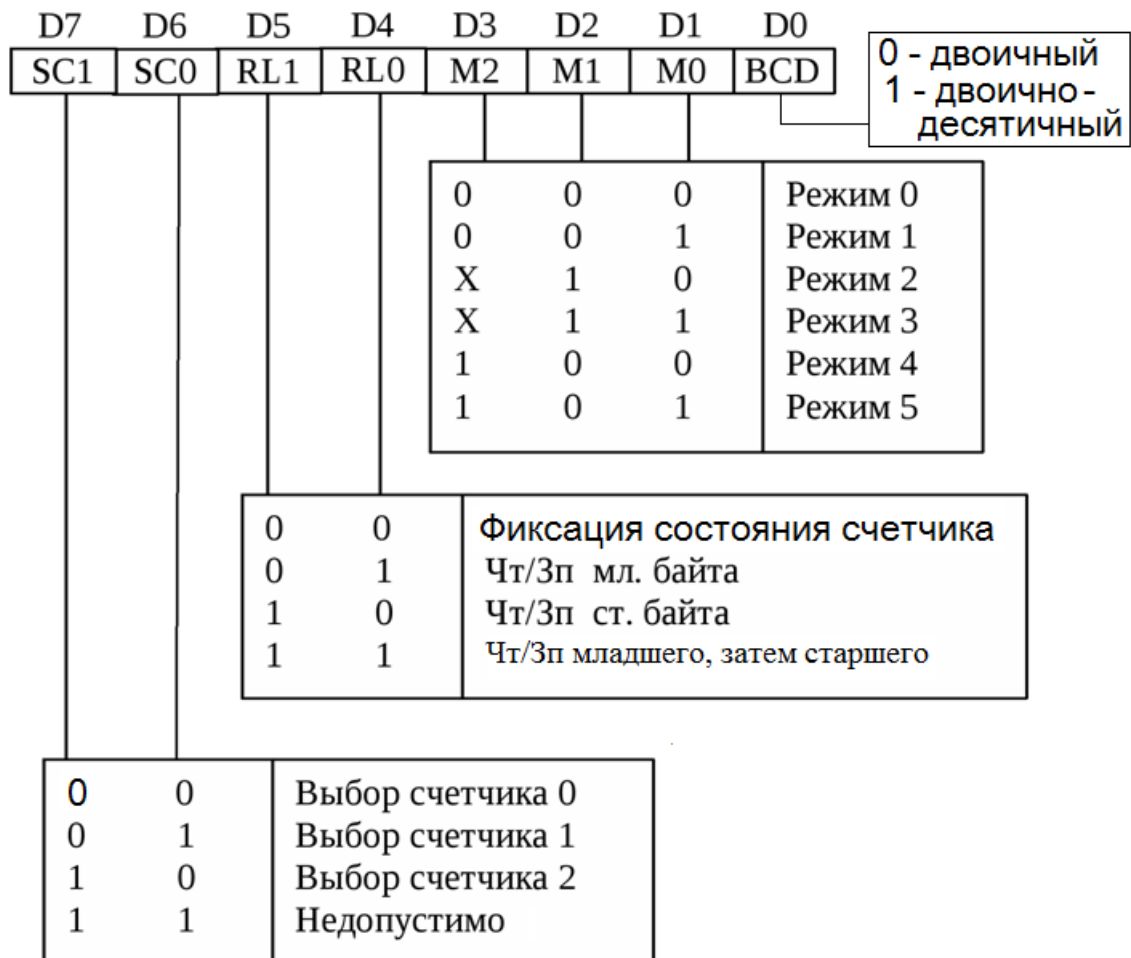


Рис.3.3. Структура управляющего слова таймера

Поле BCD (Binary-Coded Decimal) определяет способ кодирования константы предустановки счетчика.

Двоично-десятичный формат предполагает запись десятичного разряда числа в виде его четырёхбитного двоичного кода. Например, десятичное число 311 будет записано в двоичном коде как 100110111, а в двоично-десятичном коде как 0011 0001 0001. Это упрощает ввод чисел, однако уменьшает диапазон вводимых чисел при той же разрядности. При BCD кодировке диапазон допустимых чисел для 16 разрядов счетчика 0 - 9999, а в двоичном варианте 0 - 65535.

В группе RL1, RL0 код 00 задает фиксацию состояния счетчика на момент выдачи управляющего слова. Это позволяет считать состояние счетчик "на лету" в буферный регистр, не нарушая его работы. В дальнейшем его программно можно прочитать.

Режим обращения к таймеру зависит от комбинации управляющих сигналов на его входах. Следует учесть, что управляющее слово занимает 1 байт, а счетчик и его вспомогательные регистры имеют разрядность 2 байта. Обмен происходит по 8-разрядной шине данных. Поэтому бвухбайтные данные могут перемещаться за два обращения. Допустимые варианты показаны в табл.3.2.

Таблица 3.2

Функции управляющих и адресных сигналов

A1	A0	RD	WR	CS	Операция
0	0	1	0	0	Запись в счетчик СТ0
0	1	1	0	0	Запись в счетчик СТ1
1	0	1	0	0	Запись в счетчик СТ2
1	1	1	0	0	Запись управляющего слова
0	0	0	1	0	Чтение счетчика СТ0
0	1	0	1	0	Чтение счетчика СТ1
1	0	0	1	0	Чтение счетчика СТ2
1	1	0	1	0	Чтение управляющего слова
x	x	x	x	1	Шина данных в Z-состоянии (отключена)

В современных компьютерах задействованы все три канала таймера.

Канал 0 используется в системных часах. Этот канал работает в режиме 3 и используется как генератор импульсов с частотой примерно 18.2 Гц. Именно эти импульсы вызывают аппаратное прерывание таймера INT8h. Выход этого канала используется также для синхронизации некоторых дисковых операций.

Канал 1 используется для регенерации содержимого динамической памяти компьютера. Выход канала OUT используется для запроса к каналу прямого доступа DMA, который и выполняет обновление содержимого памяти. Нарушение его работы может привести к потере информации основной оперативной памяти компьютера.

Канал 2 подключен к громкоговорителю компьютера и может быть использован для генерации различных звуков или музыки, либо как генератор случайных чисел. Канал использует режим 3 таймера.

В отчете представить:

1. Структуру управляющего слова таймера;
2. Схему сопряжения БИС таймера с системным интерфейсом МПС (на логических элементах или на БИС дешифратора) по адресу 40H.

Задание 3.3. Программирование режимов работы таймера.

Для программирования (загрузки) канала таймера необходимо выполнить следующую последовательность действий:

- вывести в порт управляющего слова код управляющего слова;
- вывести код предустановки в порт канала, причем вначале выводится младший, а затем старший байты значения счетчика.

Сразу после этого канал таймера начнет выполнять требуемую функцию.

Загрузка в счетчик управляющего слова может выполняться в любой последовательности, т.к. каждый канал имеет свой буфер хранения. Каждый адресуемый счетчик в обязательном порядке должен быть загружен именно тем количеством байт, которое было задано в управляющем слове.

Следует помнить, что адрес порта образуется 8 разрядами адресной шины при раздельном способе организации адресного пространства.

Чтение содержимого счетчика при его работе может быть выполнено двумя способами:

- при помощи обычной операции чтения командами IN (или MOV для совмещенного адресного пространства). Но при этом для обеспечения стабилизации показаний счетчика его работа должна быть приостановлена подачей на управляющий вход уровня "0" (в режимах 0, 2, 3, 4) или с помощью внешней логической схемы, приостанавливающей подачу импульсов на тактовый вход канала, что неудобно;

- с помощью специального режима "чтения на лету". Такой способ позволяет процессору считывать содержимое счетчика, не прерывая процесса счета. Для этого по адресу A0=1, A1=1 осуществляется запись управляющего слова, определяющего в соответствии с рис.3.3 режим чтения на лету (D4=0, D5=0). Разряды D6 и D7 определяют номер канала, состояние разрядов D0...D3 произвольно. По этой команде в буферном регистре запоминается текущее значение счетчика, далее следует обычная операция чтения содержимого счетчика этого канала: сначала младшего байта, затем старшего. Допустим вариант чтения только младшего или только старшего байта в зависимости от управляющего слова в разрядах D4, D5.

Данная команда не изменяет режим работы счетчика. При этом способе чтения нельзя предварительно по каждому каналу сначала производить запись управляющих слов и лишь затем производить чтение счетчиков каналов. Особенность построения внутренней схемы таймера требует, чтобы операция чтения содержимого счетчика была выполнена до конца, т. е. если запрограммирована загрузка двух байтов, то нельзя, прочитав один младший байт, загружать счетчик новой командой.

Таким образом, для чтения текущего содержимого счетчика необходимо выполнить следующее:

- вывести в порт управляющего регистра код команды запоминания содержимого нужного счетчика (D5,D4=00);

- вывести в порт управляющего регистра код команды запроса на чтение/запись в регистры нужного канала (поле D5,D4=11);
- двумя последовательными командами ввода из порта нужного канала прочитать сначала младший и затем старший байты запомненного текущего состояния счетчика.

Пример.

Запрограммировать счетчик СТ0 в режим генератора импульсов для получения частоты $f_{\text{вых}} = 1\text{кГц}$. Адрес обращения к таймеру 40H. Частота входного сигнала $\text{CLK} = 5\text{МГц}$.

Для режима 2 (делителя) значения коэффициента деления:

$$N = f_{\text{CLK}} / f_{\text{вых}} = 5000000 / 1000 = 5000.$$

В двоичном представлении десятичное $5000 = 0001\ 0011\ 1000\ 1000 = 1388\text{H}$.

Коэффициент деления можно представить и в коде BCD: $5000 = 0101\ 0000\ 0000\ 0000 = 5000\text{H}$.

Управляющее слово в соответствии с рис.3.3 для программирования счетчика СТ0 в режиме 2, с двоичным способом кодирования:

D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	1	0	1	0	0

Управляющее слово - 34H.

Для процессора i8086 обращение осуществляется к портам устройств ввода-вывода только по четным адресам. Поэтому на схемах А0 микросхемы таймера подключается к адресной линии А1 процессора, а А1 микросхемы подключается к адресной линии А2.

Тогда программа для таймера К580ВИ53 в кодах процессора будет иметь вид:

ADR53	EQU 40H	; директива присваивания имени значения ; базового адреса таймера
MOV	AX,0000H	; установка номера регистра сегмента
MOV	DS,AX	; данных по условному адресу 00000H
MOV	AL, 34H	; Задание управляющего слова для СТ0
OUT	ADR+6, AL	
MOV	AL, 88	; Запись младшего байта
OUT	ADR53, AL	; кода предустановки
MOV	AL, 13H	; Запись старшего байта
OUT	ADR3, AL	; кода предустановки

После выполнения программы на выводе OUT0 импульсы с частотой 1 кГц будут до тех пор, пока не будет перепрограммирован таймер, или выключен источник тактовых сигналов 0-канала таймера, или снят сигнал разрешения GATE=1.

Задание 3.3.1. Генератор для контроллера последовательного порта

Запрограммировать канал №1 таймера в качестве генератора тактовой частоты для контроллера последовательного порта передачи данных. Частота передачи данных 4800 бит/с. Входная частота канала 2 МГц. Последовательный контроллер работает с внутренним делителем на 16. Предельно допустимая погрешность установки частоты 5%. Использовать режим таймера - 3. Адрес таймера 40H. Адрес сегмента 28000H.

В отчете представить:

1. Аналитические расчеты параметров;
2. Структуру управляющего слова;
3. Текст программы.

Задание 3.3.2. Запрограммировать мультивибратор с задержкой равной номеру зачетной книжки

Запрограммировать канал №2 таймера в качестве мультивибратора с задержкой равной номеру зачетной книжки. Входная частота канала 2 МГц. Адрес таймера 70H. Адрес сегмента 35000H.

В отчете представить:

1. Аналитические расчеты параметров;
2. Структуру управляющего слова;
3. Текст программы.

Задание 3.3.3. Прочитать содержимое счетчика CT0 и записать его в регистр ВХ

Прочитать содержимое счетчика CT0 и записать его в регистр ВХ процессора. Адрес таймера 90H. Адрес сегмента 650H. Работа счетчика нарушаться не должна.

В отчете представить:

1. Аналитические расчеты параметров;
2. Структуру управляющего слова;
3. Текст программы.

Задание 3.3.4. Запрограммировать выдачу нулевого строба с заданной задержкой

Запрограммировать канал 0 таймера на выдачу нулевого строба с задержкой (№ зачетной книжки -200) мкс. Входная частота канала 2 МГц. Адрес таймера 80Н. Адрес сегмента 400Н.

В отчете представить:

1. Аналитические расчеты параметров;
2. Структуру управляющего слова;
3. Текст программы.

Вопросы для самопроверки

1. Сформулируйте назначение таймера.
2. Какова программная модель таймера?
3. Какова структура каждого канала таймера?
4. Каково назначение типовых регистров таймера?
5. Нарисуйте схему подключения микросхемы таймера к шинам МП.
6. Какие существуют режимы работы таймера?
7. Нарисуйте схему подключения внешних устройств к таймеру.
8. Разработайте программу инициализации таймера для режима 0.
9. Разработайте программу инициализации таймера режима 1.
10. Разработайте программу инициализации таймера режима 2.
11. Разработайте программу инициализации таймера режима 3.
12. Разработайте программу инициализации таймера режима 4.
13. Разработайте программу инициализации таймера режима 5.
14. Нарисуйте схему подключения таймера подсчета внешних сигнала.

ЛОВ.

