

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
Северо-Кавказский филиал
ордена Трудового Красного Знамени федерального государственного бюджетного образовательного учреждения высшего образования
"Московский технический университет связи и информатики"



Методические указания
к лабораторным работам

МИКРОПРОЦЕССОРЫ

Периферийные устройства

Направление подготовки:

09.03.01 Информатика и вычислительная техника

11.03.02 Инфокоммуникационные технологии и системы связи

Ростов-на-Дону
2019

УДК 681.3.06 (076)

ББК 32.07

Чикалов А.Н. Микропроцессоры. Периферийные устройства. Методические указания к лабораторным работам. Ростов-на-Дону: Северо-Кавказский филиал МТУСИ, 2019.- 38 с.

В пособии изложены методические рекомендации, содержательные материалы и контрольные задания для проведения лабораторных работ 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 2019 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 .	16
3. Программирование интервального таймера	27

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. <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.

Задание 1.1 Структурная схема адаптера

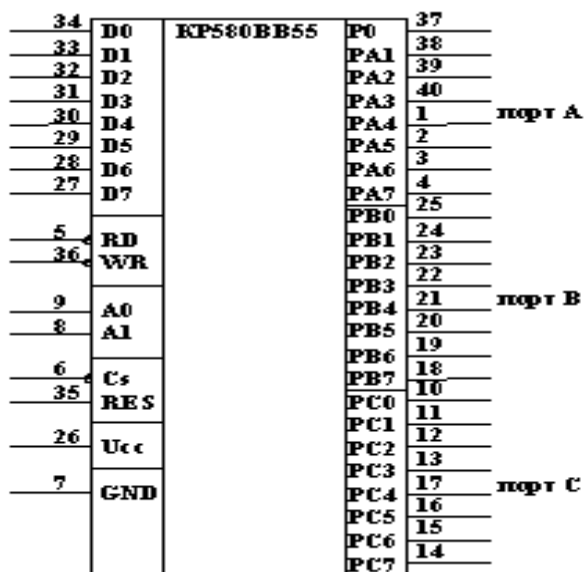


Рис.1.1. УГО ППА KP580BB55

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).

Условное графическое обозначение адаптера приведено на рис.1.1. 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).

Таблица 1.1

Назначение выводов

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
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 регистры управляющего слова и состояния (в зависимости от сигналов на входах RD и WR, см. табл. 2)
PA0...PA7	Канал А
PB0...PB7	Канал В
PC0...PC7	Канал С

ППА структурно разбит на две группы (А и В). Группа А включает в себя канал А и старшую тетраду канала С. Группа В состоит из канала В и младшей тетрады канала С.

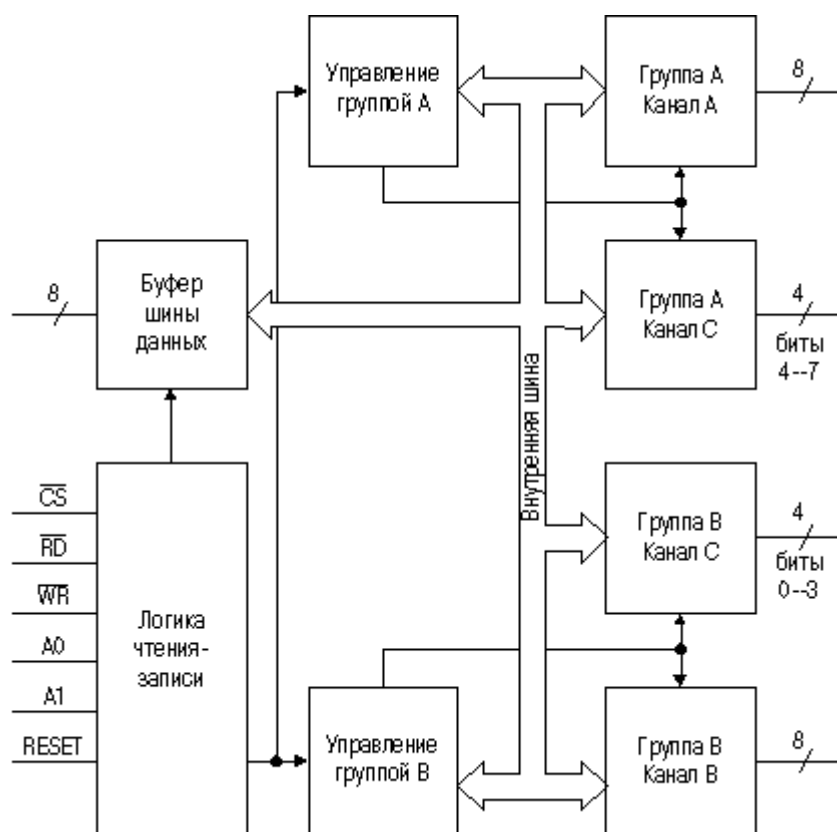


Рис.1.2. Структурная схема ППА KP580BB55

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, 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 ВВ 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, простой статический ввод/вывод) могут работать все три порта. Направление для каждого порта определяется индивидуально. Порт С можно разделить на две независимых половины. Выходы имеют память, входы работают без буферной памяти.

В этом режиме информация, поступающая на входы каналов, записывается в буферные регистры этих каналов без стробирования, 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 ввода/вывода) может работать порт А и порт В. Для управления передачей или приемом по каналу А используется младшая тетрада канала С, канал В управляется старшей тетрадой канала С.

Используя режим 1, можно реализовать программно-управляемый обмен между МП и УВВ.

Для организации этой процедуры порт РС служит регистром состояний внешнего устройства, порты РА и РВ для передачи данных. Опрос и модификация разрядов порта С осуществляется программно. Порт С 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). По низкому уровню на этом входе входные данные записываются в буферную память;

IBF (INPUT BUFFER FULL). Высокий уровень на этом выходе показывает, что данные загружены в буферную память. Задний фронт сигнала STB устанавливает этот выход в уровень лог. 1, передний фронт сигнала на входе RD сбрасывает его в лог. 0;

INTR (INTERRUPT). Высокий уровень на этом выходе может быть использован для прерывания выполнения основной программы процессора, чтобы обслужить устройство ввода. Сигнал INTR устанавливается в лог. 1 по переднему фронту сигнала /STB (если установлен сигнал INTE разрешения прерывания) и сбрасывается по заднему фронту сигнала на входе /RD микросхемы. Этот процесс обеспечивает ввод данных в микропроцессор со стороны устройства ввода простым стробированием своих данных;

INTE A. Разрешение прерывания для канала A. Управляется установкой/сбросом бита 4 канала C.

INTE B. Разрешение прерывания для канала B. Управляется установкой/сбросом бита 2 канала C.

Взаимодействие описанных сигналов показано на рис.1.3.

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 A и B образуют два канала, а порт C используется для накопления и обработки сигналов управления обменом. При вводе в режим 1 устройство, требующее обмена, формирует на соответствующем разряде порта C (PC4 для порта A или PC2 для порта B) инверсный управляющий сигнал STB (строб байта) и загружает в нужный порт 8-ми разрядный код, который запоминается в этом порту. В ответ ППА формирует сигнал IBF (входной буфер полон), извещающий УВВ о невозможности приема следующего байта. С появлением сигнала IBF внешнее устройство снимает сигнал STB.

Если управляющим словом BIT SET/RESET установлен в 1 соответствующий триггер INTE (INTEA или INTEB), то данному УВВ разрешена работа в режиме прерывания. Тогда по заднему фронту стробирующего сигнала STB ППА генерирует и передает в МП сигнал запроса прерывания INT. При удовлетворении запроса, в ППА поступает сигнал RD, по переднему фронту которого снимается сигнал INT, а по заднему фронту устанавливается в 0 разряд IBF, что свидетельствует о прочтении процессором данного байта и возможности загрузки в порт ППА очередного байта со стороны УВВ.

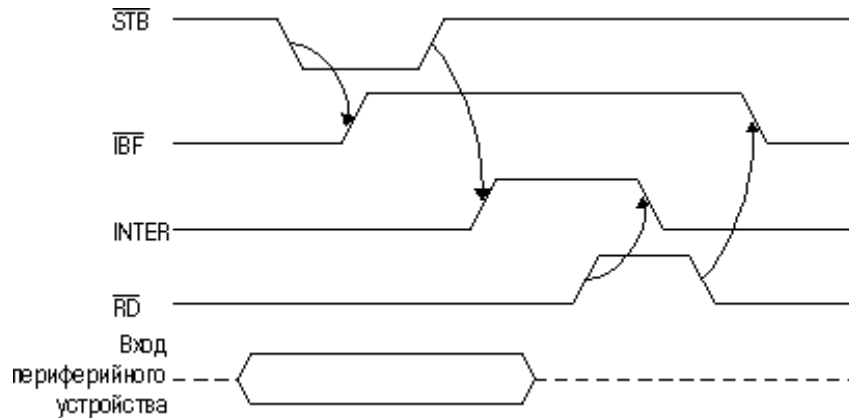


Рис.1.3. Динамика сигналов управления вводом режим 1

Описание сигналов для управления операцией вывода (рис.1.4):

OBF (OUTPUT BUFFER FULL). Сигнал высокого уровня на этом выходе показывает устройству вывода, что микропроцессор записал данные в канал. Сигнал устанавливается в лог. 1 по переднему фронту на входе WR микросхемы и сбрасывается по заднему фронту сигнала ACK;

ACK (ACKNOWLEDGEMENT). Низкий уровень на этом выходе показывает микропроцессору, что периферийное устройство приняло данные;

INTR (INTERRUPT). Высокий уровень на этом выходе может быть использован для прерывания выполнения основной программы процессора в момент принятия данных периферийным устройством. Сигнал INTR устанавливается в лог. 1 по переднему фронту сигнала ACK (если одновременно сигналы OBF и INTE установлены в лог. 1) и сбрасывается по заднему фронту сигнала на входе /WR микросхемы;

INTE A. Разрешение прерывания для канала A. Управляется установкой/сбросом бита 6 канала C;

INTE B. Разрешение прерывания для канала B. Управляется установкой/сбросом бита 2 канала C.

Вывод в режиме 1 тоже связан с процессом квитирования. При выводе в режиме 1 цикл работы начинается с записи байта в данный порт (A или B). С завершением сигнала записи WR формируется сигнал OBF (выходной буфер полон), который извещает УВВ о том, что МП выдал байт на выход. Когда устройство ввода-вывода готово прочитать байт, оно генерирует единственный внешний управляющий сигнал ACK. По этому сигналу открываются выходные буферы соответствующих портов A или B, данные передаются в УВВ. Появление сигнала ACK снимает сигнал OBF.

По заднему фронту сигнала ACK сигнал INT устанавливается в 1, что может быть использовано системой прерывания МП в качестве извещения о готовности УВВ к приему новой информации. Теперь цикл вывода можно

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

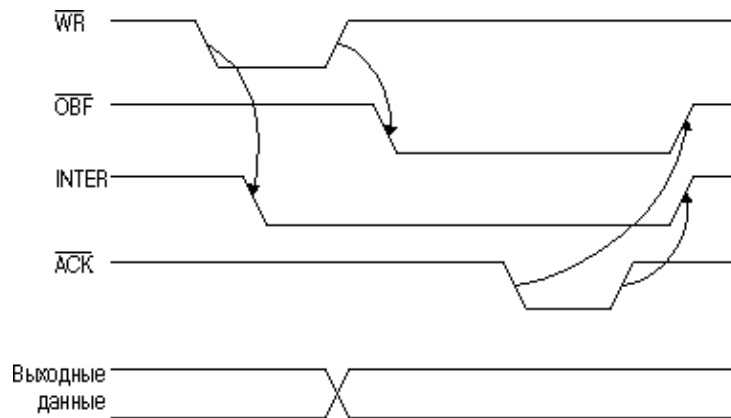


Рис.1.4. Взаимодействие сигналов управления выводом

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

Режимы на ввод или вывод задаются для каждого порта независимо. Входы и выходы обладают буферной памятью (рис.1.5).Организация обмена и программирование ППА в этом режиме аналогична режиму 1.

Описание управляющих сигналов для двунаправленного ввода/вывода:
INTR (INTERRUPT). Высокий уровень на этом выходе может быть использован для прерывания выполнения основной программы процессора.

Режим вывода:

OBF (OUTPUT BUFFER FULL). Сигнал высокого уровня на этом выходе показывает устройству вывода, что микропроцессор записал данные в канал А;

ACK (ACKNOWLEDGEMENT). Низкий уровень на этом выходе выводит выходной буфер канала А из третьего состояния для вывода данных в периферийное устройство. В остальных случаях выходной буфер остается в третьем состоянии;

INTE 1 (INTE плюс OBF). Разрешение прерывания для канала А при работе на вывод. Управляется установкой/сбросом бита 6 канала С.

Режим ввода:

STB (STROBE). Низкий уровень на этом входе загружает данные из периферийного устройства в буферную память канала А;

IBF (INPUT BUFFER FULL). Высокий уровень на этом выходе показывает, что данные загружены в буферную память;

INTE 2 (INTE плюс IBF). Разрешение прерывания для канала А при работе на вывод. Управляется установкой/сбросом бита 4 канала С.

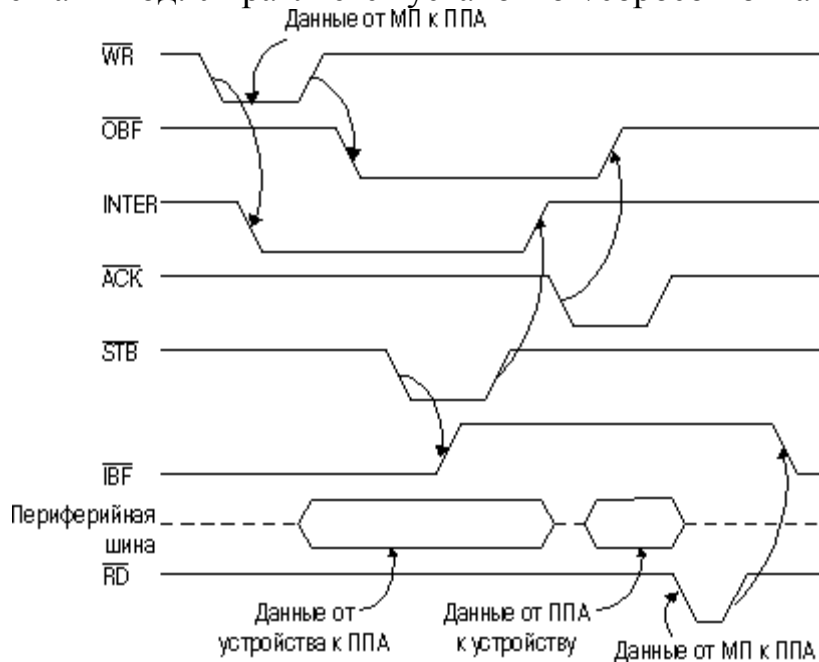


Рис.1.5. Работа ППА в режиме 2

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

На основе ППА можно разрабатывать схемы управления различными устройствами, например, клавиатурой или индикатором. Простейшая система ввода и отображения информации (рис.1.6) содержит клавиатуру на 16 клавиш и 5 семисегментных индикаторов.

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	STB	INTR	STB	IBF	INTR	
Канал А				Канал В				Режим 1 Вывод
OBF	ACK	I/O	I/O	INTR	ACK	OBF	INTR	
Канал А								Режим 2
OBF	ACK	IBF	STB	INTR	I/O	I/O	I/O	

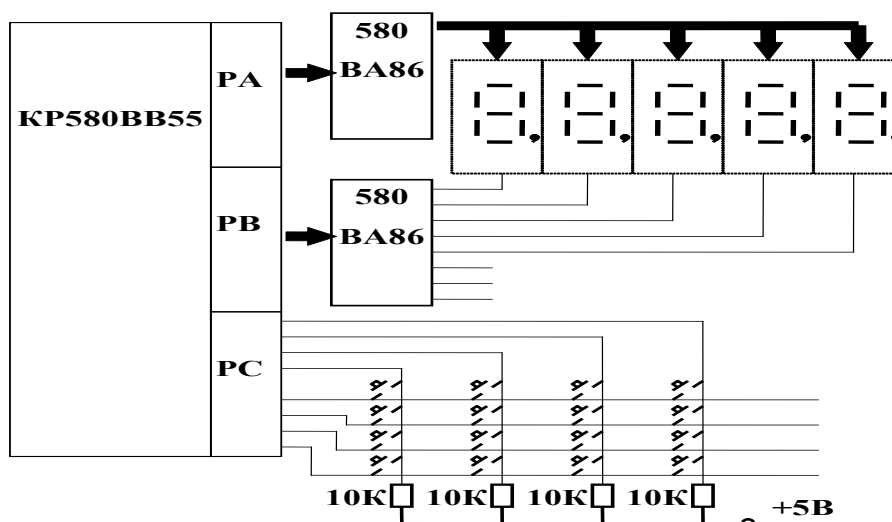


Рис.1.6. Схема ввода и отображения на основе ППА

Режим работы и направление передачи для каждого канала задается с помощью записи управляющего слова в регистр управления в соответствии с табл.1.4 или 1.5. 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	D6	D5	D4	D3	D2	D1	D0
Флаг управления 1 = выбор режима	Группа А				Группа В		
	Выбор режима: 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: 0 = режим 0 1 = режим 1	Канал В: 1 = ввод 0 = вывод	Канала С РС3-РС0: 1 = ввод 0 = вывод

Таблица 1.5

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

Каждый из восьми битов канала С может быть независимо от других установлен или сброшен с помощью специальной команды вывода, посылаемой в регистр управления. Это свойство канала С облегчает программирование в случае, когда биты канала С используются для управления передачей данных по каналам А или В. Формат команды приведен в табл.1.6.

Работа микросхемы в режиме 1 или 2 характеризуется наличием управляющих сигналов, которые могут вызывать прерывание работы процессора КР580ВМ80. Любой из разрядов порта С может быть выбран разработчиком МПС в качестве триггера запроса прерывания INT или триггера фиксации разрешения прерывания INTE. Программный доступ к ним дает возможность программисту разрабатывать различные процедуры обработки прерываний, наиболее эффективно приспособив их к структуре МПС.

Таблица 1.6

Формат управляющего слова для изменения битов канала С

D7	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	

Задание 1.3. Программирование режимов работы адаптера.

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 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. <http://literaturki.net/elektronika/mikroprocessory--mikrokmntrollery/250-18-posledovatelnyi-adapter-vv51a>

Содержание отчета

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

Задание 2.1 Структурная схема адаптера

Для связи с удаленными источниками информации, например с датчиками параметров, установленными на технологическом оборудовании в производственном помещении, нерационально проводить 8-разрядные магистрали для параллельной передачи. 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. Микросхема программируемого связного адаптера КР580ВВ51 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.2.1.

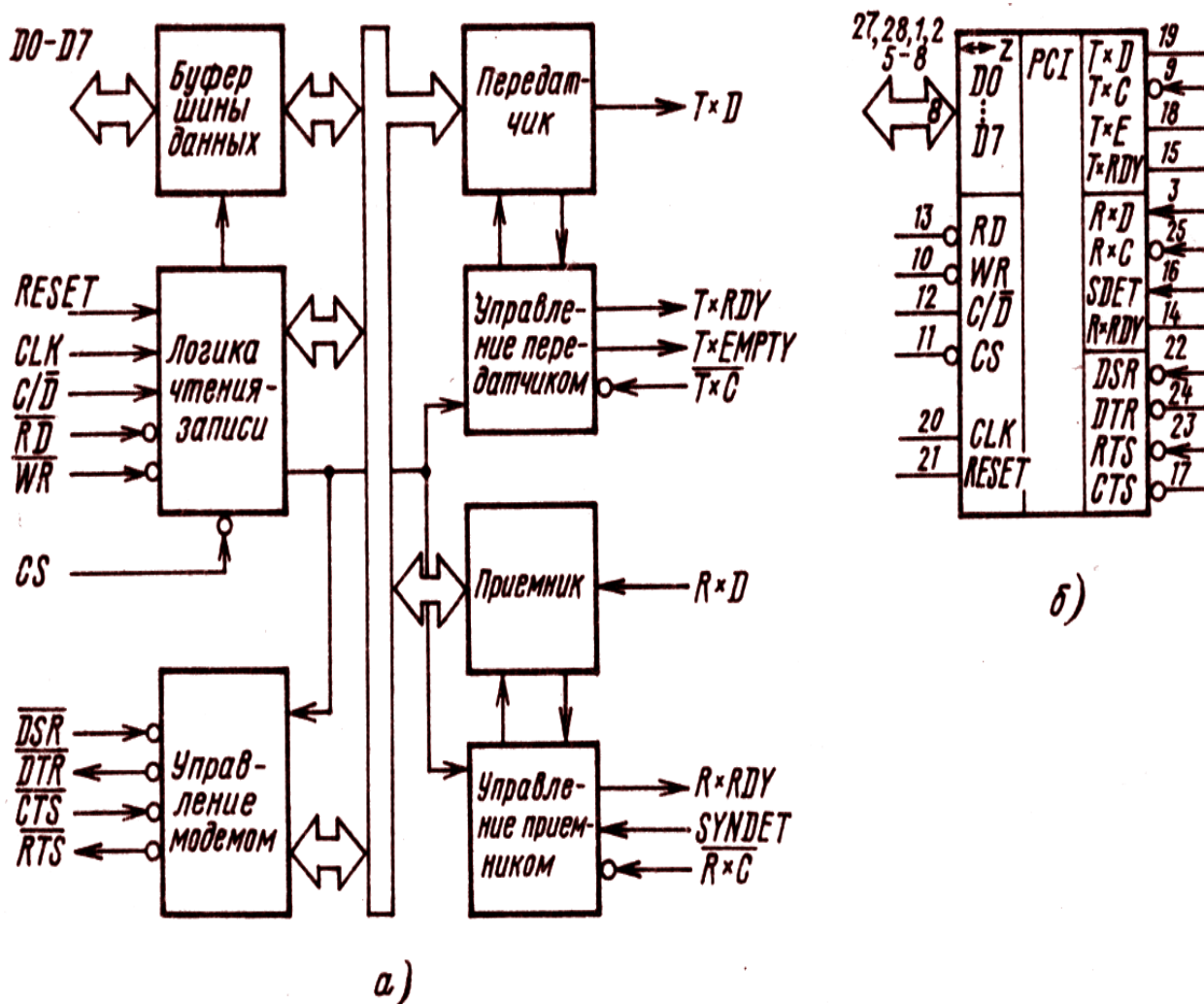


Рис.2.1. Структурная схема (а) и УГО периферийного связного адаптера (ПСА)

На рис.2.2 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;

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

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

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

C/D (управление/данные)— сигнал идентификации передачи данных или управляющих слов;

CS (выбор кристалла)— нулевой активный сигнал, разрешающий связь между адаптером с шиной данных.

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

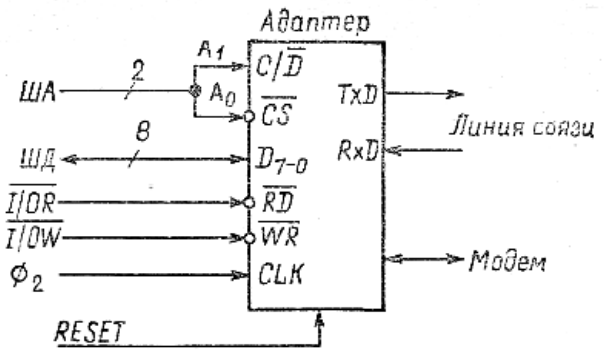


Рис.2.2. Подключение ПСА к системной шине

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 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. К этому узлу относятся 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 час-

тоте сигнала TxС, а в асинхронном режиме программируется как 1, 1/16 или 1/64 частоты сигнала TxС;

TxEmpy (пустой передатчик) — единичный активный выходной сигнал, обозначающий отсутствие в адаптере символа для передачи. Его можно использовать для идентификации в полудуплексной связи окончания передачи и коммутации линии на прием. В синхронном режиме Н-уровень TxЕ указывает, что символ вовремя не загружен в адаптер и в качестве «заполнителей» автоматически передаются символы синхронизации. Сигнал TxЕ сбрасывается при загрузке символа в адаптер;

TxRDY (готовность передатчика)—Н-активный выходной сигнал, определяющий готовность передатчика к восприятию символа. 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 «собранный» 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 синхронномрежиме скорость приема соответствует частоте сигнала RxС, а в асинхронном частота RxC кратна скорости приема. Данные вводятся в адаптер по фронту RxC. Обычно передача и прием осуществляются с одинаковой скоростью, поэтому сигналы TxС 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. Его можно подключить на вход прерывания процессора или проверить значение при считывании состояния адаптера (бит 1). Когда процессор вводит символ из адаптера, сигнал RxRDY сбрасывается;

SYNDET (обнаружение синхронизации)—Н-активный сигнал синхронного режима, который может быть запрограммирован как выходной или входной. Если он запрограммирован как выходной, то при обнаружении символа SYN на выходе SYNDET формируется высокий уровень в момент времени, соответствующий середине последнего бита (в случае работы с двумя символами синхронизации это относится к SYN2). При считывании состояния адаптера сигнал сбрасывается. Когда сигнал SYNDET определен как входной, подача на него высокого уровня фиксирует момент начала приема символа. Адаптер обслуживает четыре линии управления модемом, сигналы которых при необходимости могут использоваться и для других функций.

Выходными сигналами можно управлять с помощью слов-приказов, входные сигналы считываются в слове состояния;

DSR (готовность модема)—входной сигнал от модема, означающий его готовность к работе (бит 7 в слове состояния);

DTR (готовность терминала)—выходной сигнал (бит 1 слова приказа);

RTS (запрос передачи) — выходной сигнал (бит 5 слова приказа);

CTS (передача) - входной сигнал, разрешающий адаптеру передавать данные, обычно генерируется модемом в ответ на RTS.

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.

В отчете представить:

1. Структурную схему ПСА;;
2. Схему алгоритма обмена;

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, т. е. передаваемый символ, снабженный 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 генера-

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

Формат управляющего слова для синхронного режима представлен на рис.2.3.

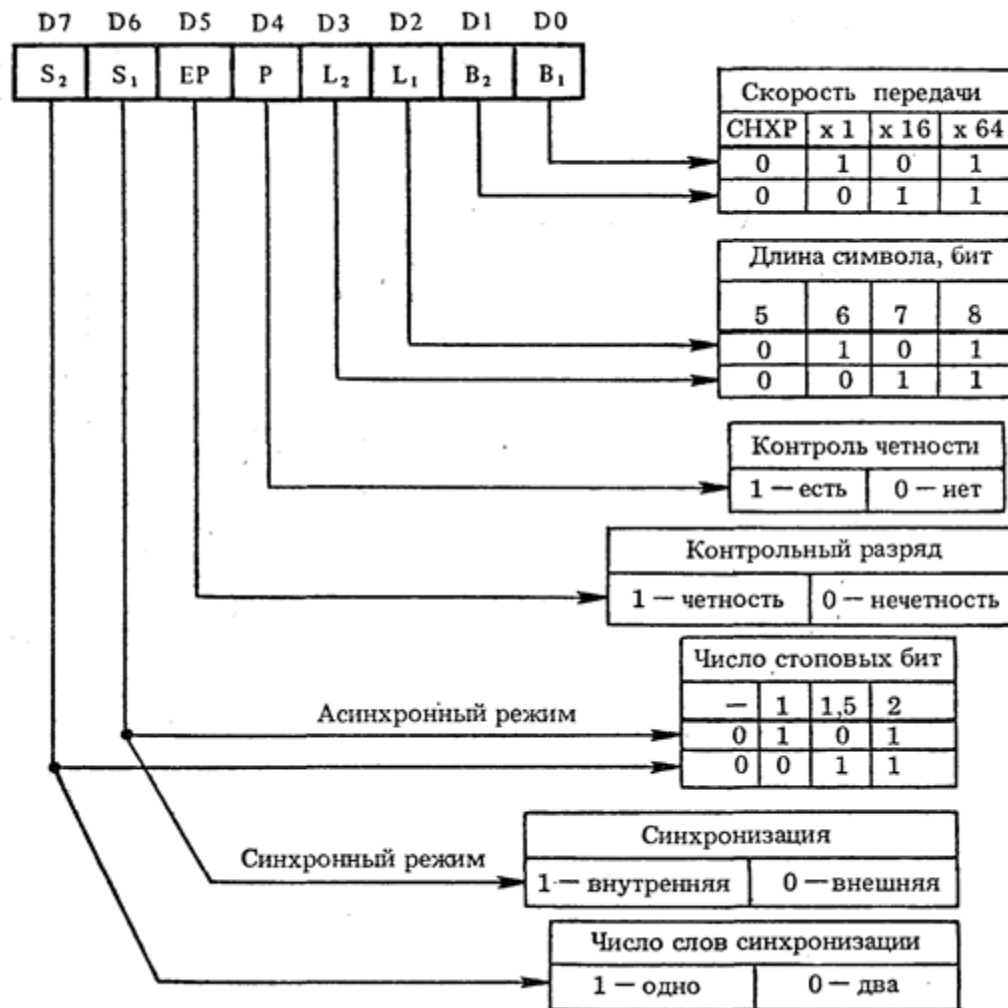


Рис.2.3. Формат управляющего слова ПСА

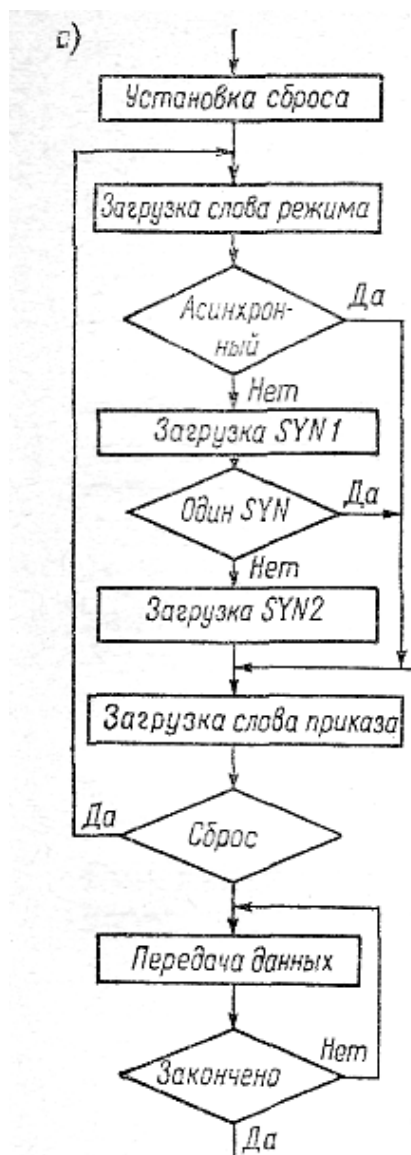
В отчете представить:

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, 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. 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.

Управляющие слова, определяющие функциональную конфигурацию адаптера, должны загружаться сразу после операции сброса. Схема алгоритма программирования приведена на рис.2.4.



Управляющие слова имеют два формата: **слово режима**(рис.2.5) и **слово приказа**. 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 SYN, если был определен синхронный режим. Символ SYN2 пропускается, если адаптер программируется для работы с одним символом синхронизации. Когда же в слове режима определен асинхронный обмен, пропускаются оба символа SYN. Последним в адаптер загружается слово приказа, определяющее его конкретные действия в соответствии со словом режима.

Для правильной работы адаптера управляющие слова должны следовать в определенном порядке. Слово режима записывается в адаптер сразу после сброса прибора, инициируемого сигналом RESET или специальным битом в слове приказа.

После того как слово режима запрограммирует общие функции адаптера и будут загружены один или два символа синхронизации синхронного режима, адаптер готов к обмену данными.

Рис.2.4. Схема алгоритма программирования ПСА

Загружаемое после этого **слово приказа** задает действительную операцию адаптера. В слове приказа задаются разрешение передачи или приема, сброс ошибок, управление модемом и некоторая другая информация. Отдельные биты приказа имеют следующий смысл:

D0(TxEN — разрешение передачи); единичное значение разрешает передачу данных, а нулевое — запрещает;

D1 (DTR); если этот бит содержит единицу, на выходе DTR адаптера формируется низкий уровень;

D2(RxE — разрешение приема) действует как маска сигнала готовности приемника RxRDY (1 - прием разрешен, 0 - прием запрещен). Когда устанавливается бит RxE, приемник может содержать один или два символа;

D3(SBRK - разрыв) при единичном значении на линии TxD формирует низкий уровень, а нулевое значение соответствует нормальной работе;

D4(ER - сброс ошибок) при единичном значении форсирует сброс ошибок паритета, переполнения и нарушения кадра;

D5 (RTS); если этот бит содержит единицу, на выходе RTS адаптера формируется низкий уровень;

D6(IR - внутренний сброс) при единичном значении инициирует внутренний сброс адаптера, после которого должно загружаться слово режима;

D7(EH - режим поиска); в синхронном режиме единичное значение этого бита вводит режим поиска символа синхронизации.



Рис.2.5. Формат слова режима ПСА

При организации последовательного интерфейса возникает необходимость проанализировать состояние адаптера. Состояние адаптера можно считать в любой момент времени посредством команды ввода IN, обеспечиваю-

щей формирование сигнала $C/D = 1$. Кроме уже рассмотренных сигналов DSR(D7), SYNDET(D6), TxE(D2), RxRDY(D1) и TxRDY(D0) **слово состояния**(рис.2.6) содержит следующие три бита, или флажка, ошибок, формируемых в процессе приема данных:

D5 (FE - ошибка кадра), устанавливается в асинхронном режиме, если в конце любого символа не обнаружен столовый бит;

D4(OE — ошибка переполнения) устанавливается в любом режиме, когда процессор вовремя не считал символ (этот символ теряется);

D3 (PE - ошибка паритета) устанавливается при обнаружении в принятом символе нарушения паритета.

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

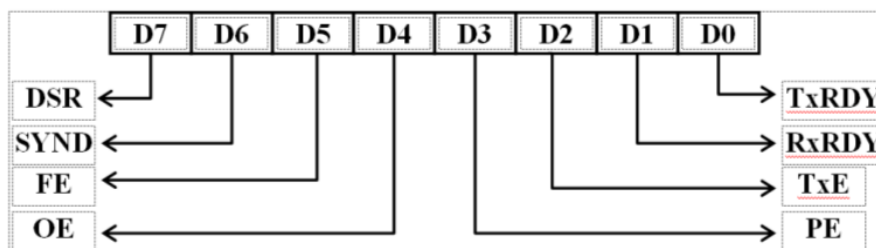


Рис.2.6. Слово состояния ПСА

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

В отчете представить:

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

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

В отчете представить:

1. Формат управляющего слова для задания режима;
2. Схему алгоритма обмена;
3. 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 (трехстабильный 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.

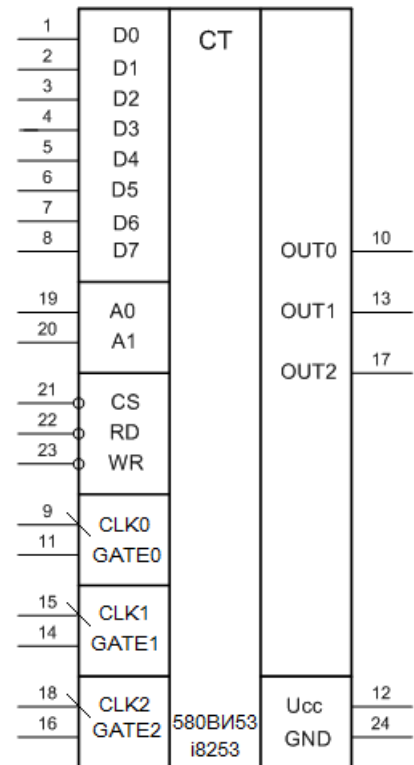


Рис.3.1. УГО KP580BI53

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).

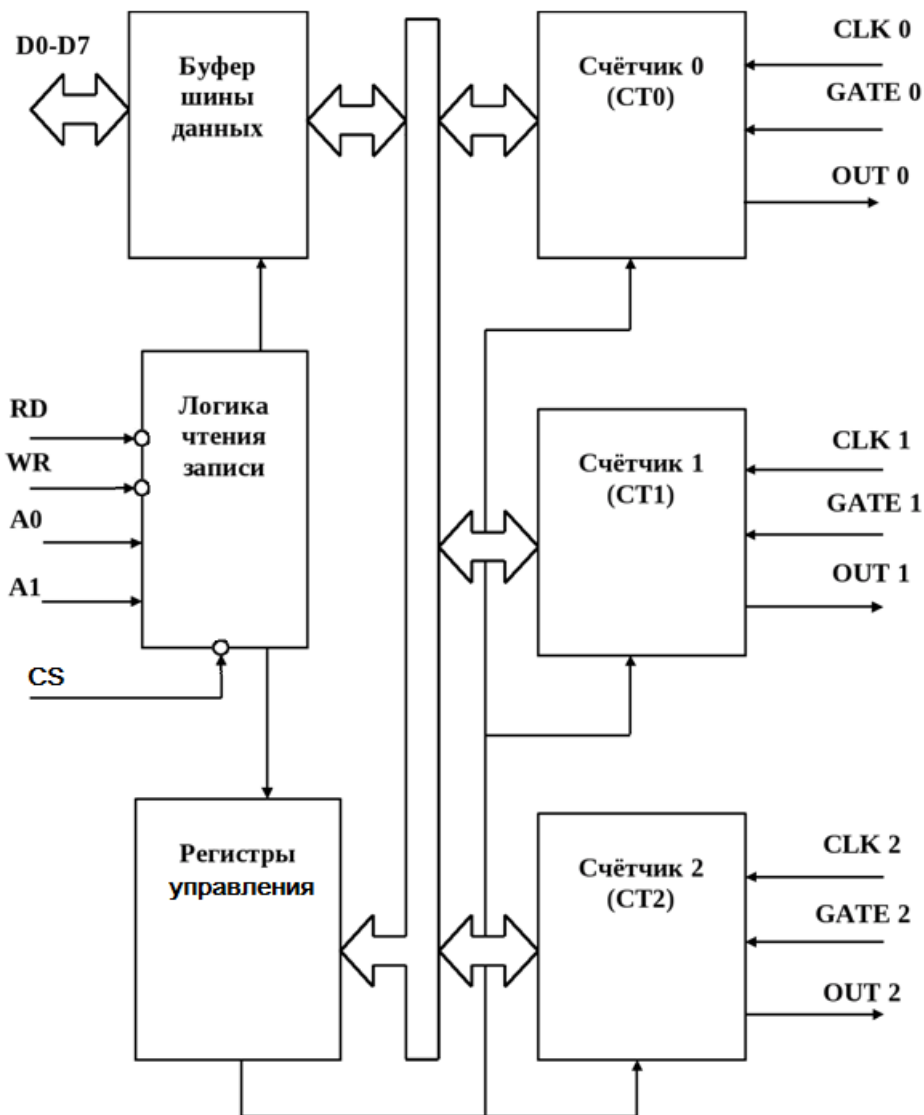
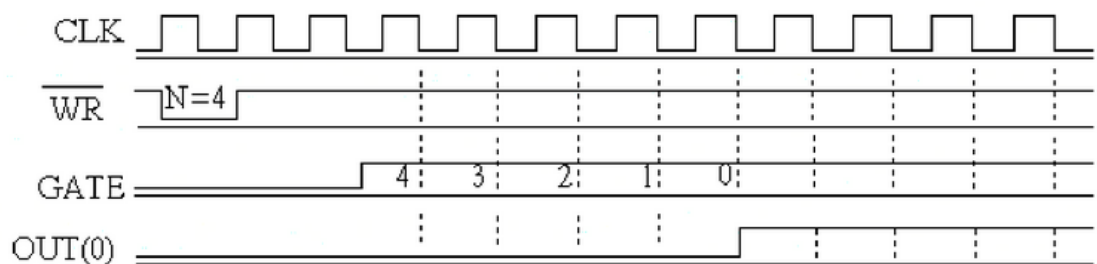


Рис.3.2. Структурная схема таймера

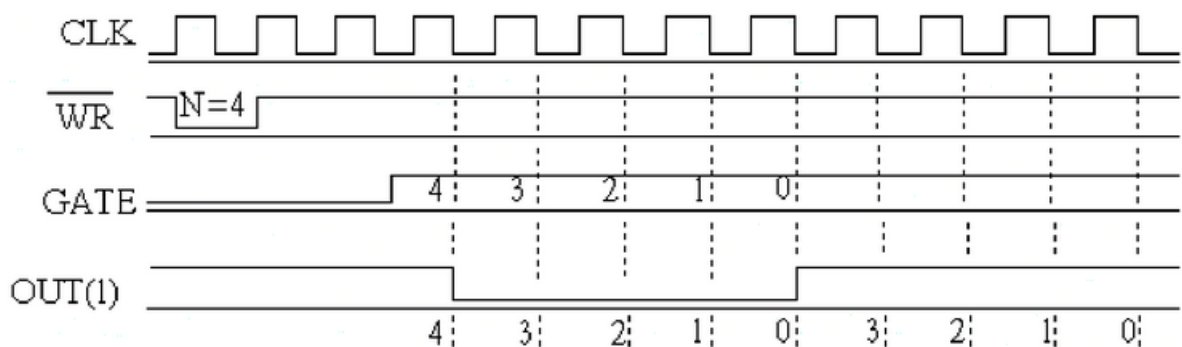
Каналы таймера могут работать в следующих режимах:

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.



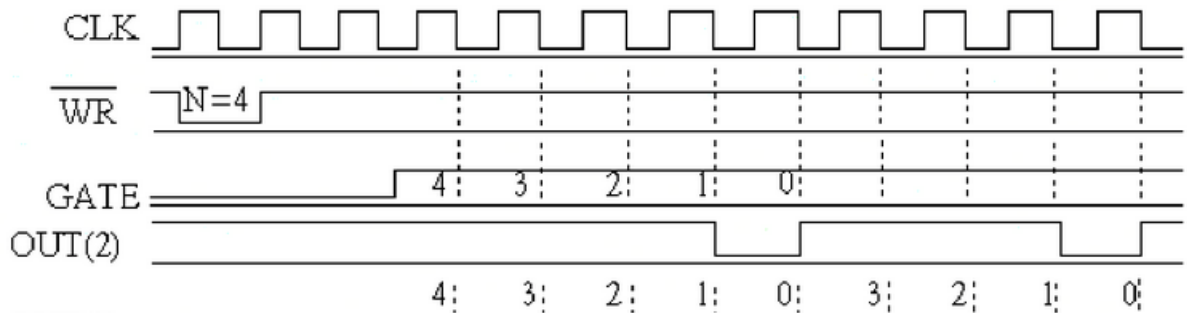
Режим 1(001) ждущий мультивибратор с программно-устанавливаемой длительностью сигнала N тактовых периодов. В этом режиме выход канала OUT устанавливается в уровень "0" после первого заднего фронта тактового сигнала, следующего за передним фронтом на управляющем входе GATE. Одновременно начинается счет, 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.

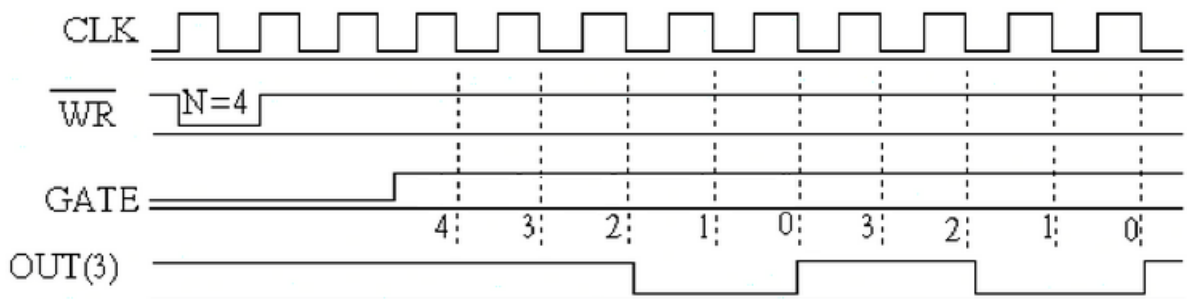


Режим 2(x10) генератор тактовых сигналов (делитель частоты). На выходе канала через число периодов тактовой частоты, записанное в счетчике канала, появляется уровень "0" длительностью в один период тактовой частоты. Начальная установка счетчика N определяет коэффициент деления.

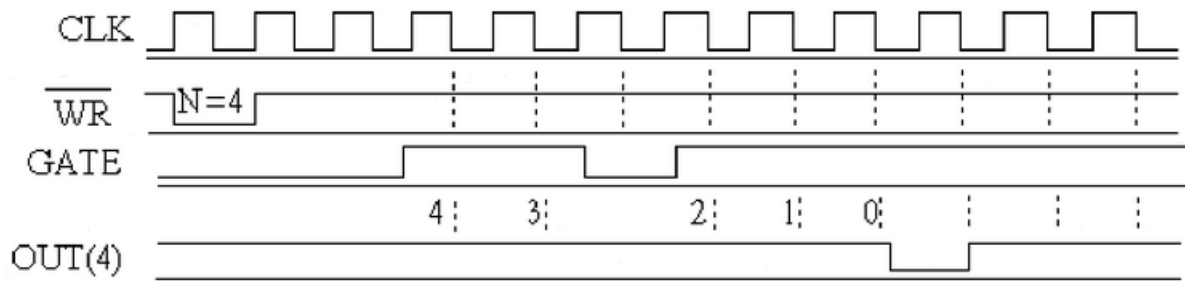
Перезагрузка счетчика не приводит к изменению длительности текущего периода, но влияет на длительность последующего. При подаче на управляющий вход GATE уровня «0» выход OUT устанавливается в состояние «1». Передний фронт GATE запускает счетчик из начального состояния и может служить для аппаратной синхронизации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



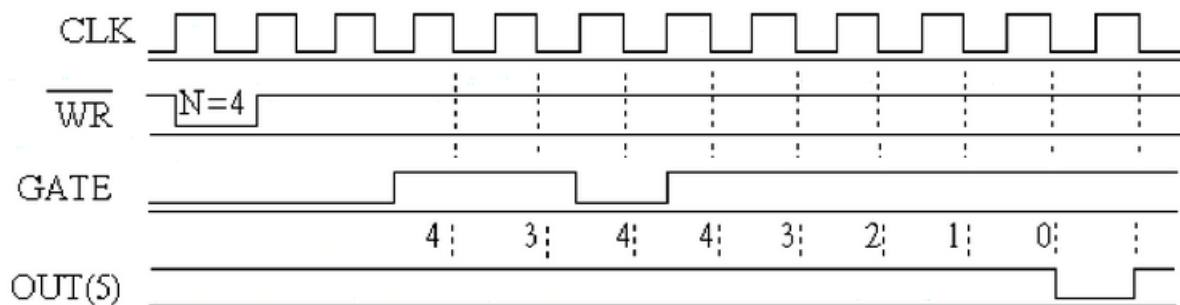
Режим 3(x11) генератор прямоугольных импульсов программно управляемым периодом. На выходе канала будет уровень "1" в течение первой половины интервала времени, определяемого числом N в счетчике, и уровень "0" в течение второй половины. При нечетном N длительность сигнала "1" на один такт больше, чем 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 строба (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 строга. 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 счётчик нового значения N в процессе счёта 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.

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
0	1	CT1
1	0	CT2
1	1	RCW Регистр управляющего слова

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

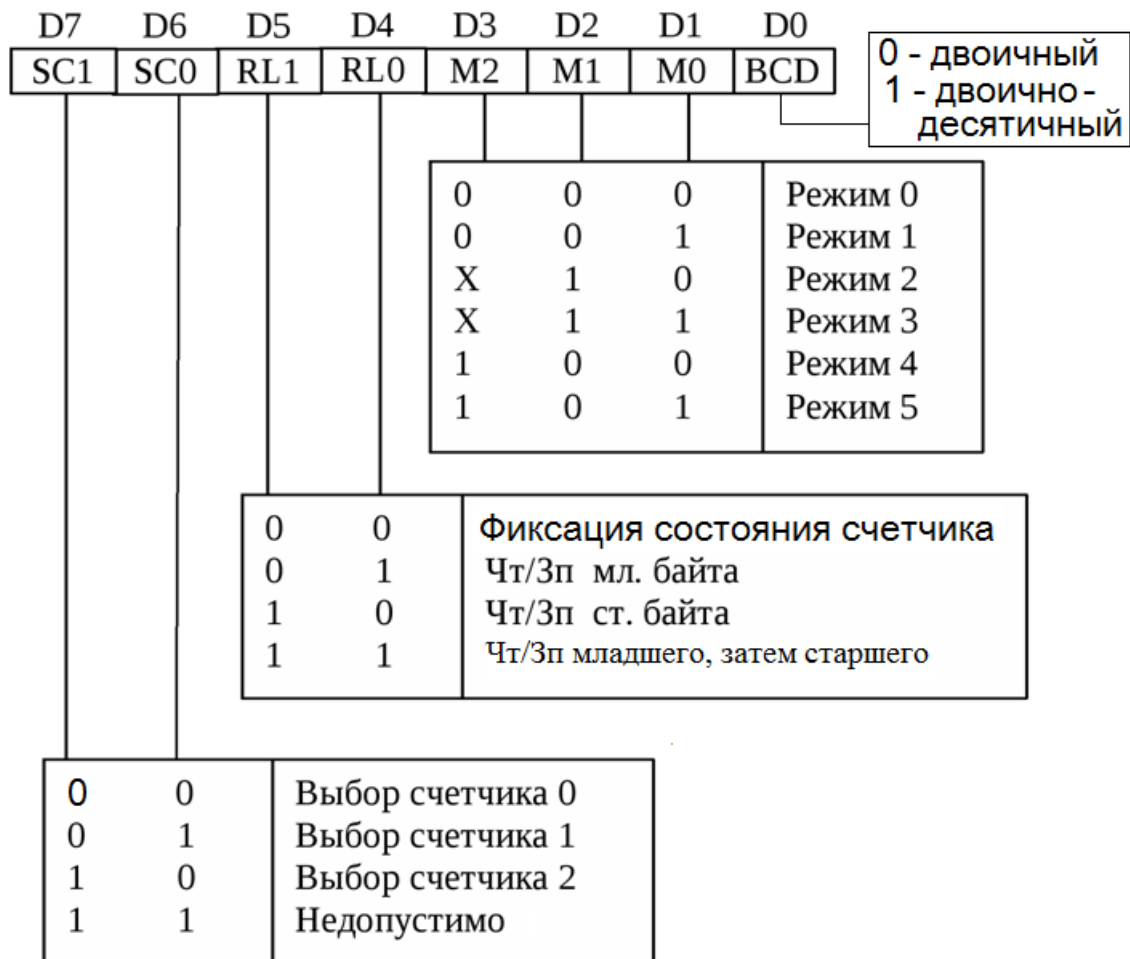


Рис.3.3. Структура управляющего слова таймера

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

Двоично-десятичный формат предполагает запись десятичного разряда числа в виде его четырехбитного двоичного кода. Например, десятичное число 311 будет записано в двоичном коде как 100110111, а в двоично-десятичном коде как 001100010001. 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.

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 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.

Данная команда не изменяет режим работы счетчика. При этом способе чтения нельзя предварительно по каждому каналу 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 счетчикСТ0в режим генератора импульсов для получения частоты $f_{\text{вых}} = 1\text{кГц}$. Адрес обращения к таймеру 40H. Частота входного сигнала CLK= 5МГц.

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 кодеBCD: $5000 = 0101\ 0000\ 0000\ 0000 = 5000\text{H}$.

Управляющее слово в соответствии с рис.3.3 для программирования счетчикаСТ0в режиме2, с двоичным способом кодирования:

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

Управляющее слово - 36H.

Тогда программа для таймера К580ВИ53 в кодах процессора i8086 будет иметь вид:

ADR53	EQU 40H	; директива присваивания имени значения
		; адреса таймера
MOV	AX,0000H	;установка регистра сегмента
MOV	DS,AX	;по условному адресу 0000H
MOV	AL, 36H	; Задание управляющего слова
OUT	ADR+3, 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 выводеOUT0импульсы 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.

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 40Н. Адрес сегмента 2800Н.

В отчете представить:

1. Аналитические расчеты параметров;
2. 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 70Н. Адрес сегмента 3500Н.

В отчете представить:

1. Аналитические расчеты параметров;
2. Текст программы.

Задание 3.3.3. Прочитать содержимое счетчика СТ0 и записать его в регистр ВХ

Прочитать содержимое счетчика СТ0 и записать его в регистр ВХ процессора. Адрес таймера 90Н. Адрес сегмента 650Н. Работа счетчика нарушаться не должна.

В отчете представить:

1. Аналитические расчеты параметров;
2. 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. Текст программы.

Вопросы для самопроверки

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