

Утвержден
3.055.430РЭ-ЛУ

ПРОЦЕССОР КОНСОЛЬНЫЙ СМ 2700.2805

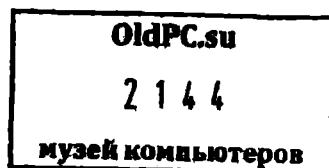
Руководство по эксплуатации

Часть 1

3.055.430РЭ

Магнитная лента

Листов 97



1987

Перв. примен.
3.055.430

Литера

Инв. N подл.	Подп. и дата	взам. инв. N	Инв. N дубл.	Подпись и дата

3.055.430PЭ

СОСТАВ ДОКУМЕНТА

Документ состоит из двух частей:

Часть первая - "Руководство по эксплуатации", 3.055.430PЭ, в которой изложен текстовый материал описывающий принципы работы консольного процессора.

Часть вторая - "Руководство по эксплуатации", 3.055.430PЭ1, в которую вошло приложения 4 и 5 (рисунки, поясняющие работу консольного процессора).

Ссылки на рисунки даны без указания обозначения и наименования часть 2.

СОДЕРЖАНИЕ

Лист

Часть 1

1.	ВВЕДЕНИЕ.....	6
2.	НАЗНАЧЕНИЕ.....	8
3.	ТЕХНИЧЕСКИЕ ДАННЫЕ.....	9
4.	УКАЗАНИЕ МЕР БЕЗОПАСНОСТИ.....	11
5.	УСТРОЙСТВО И РАБОТА КОНСОЛЬНОГО ПРОЦЕССОРА.....	12
5.1.	Микропроцессор.....	13
5.2.	Универсальные синхронно-асинхронные приемопередатчики (УСАПП-ы).....	16
5.2.1.	Основные операции.....	17
5.2.2.	Импульсы синхронизации.....	18
5.2.3.	Передача данных терминала и ленты.....	19
5.3.	Таймер.....	24
5.3.1.	Описание сигналов интерфейса.....	24
5.3.2.	Функциональное описание.....	27
5.3.3.	Регистры порта сигналов управления.....	29
5.3.4.	Регистры команд.....	29
5.3.5.	Регистр указателя данных.....	32
5.3.6.	Регистр состояния.....	34
5.3.7.	Регистры порта данных. Логические группы счетчиков.....	34
5.3.8.	Регистры режима работы счетчиков.....	35
5.3.9.	Регистры сигнализации и компараторы.....	35
5.3.10.	Доступ к регистрам.....	36
5.3.11.	Управление главным режимом работы.....	36
5.3.12.	Описание режимов работы счетчиков.....	38
5.3.13.	Варианты управления режимом работы счетчика.....	38

5.055.430P3

5.3.14.	Описание команд.....	40
5.3.15.	Внутренний таймер центральной части.....	41
5.3.16.	Связь псевдорегистров центральной части с консолью.....	42
5.3.17.	Часы времени года (группы логических счетчиков 4 и 5).....	45
5.3.18.	Таймер азарии питания (группа 3 логических счетчиков).....	47
5.4.	Операции чтения и записи.....	48
5.4.1.	Логика, управляющая чтением/записью.....	50
5.4.2.	Операции чтения ПЗУ.....	51
5.4.3.	Работа ОЗУ.....	52
5.4.4.	Операции ввода/вывода.....	54
5.4.5.	Чтение и запись регистров УСАПП-ов.....	55
5.4.6.	Чтение и запись других устройств.....	56
5.5.	Связь между консольным процессором и процессором.....	58
5.5.1.	Общение в консольном режиме.....	58
5.5.2.	Общение в программном режиме.....	61
5.6.	Микропрограммная память.....	62
5.6.1.	Базовая микропрограммная память.....	63
5.6.2.	Пользовательская микропрограммная память.....	64
5.6.3.	Регистр микрокоманд (CSR).....	65
5.6.4.	Основной микроцикл.....	65
5.6.5.	цикл регенерации базовой микропрограммной памяти.....	67
5.6.6.	Операция записи микропрограммной памяти.....	68
5.6.7.	Паритет микропрограммной памяти и микросинхронизация.....	70
6.	ПОДГОТОВКА К РАБОТЕ.....	72
6.1.	Порядок установки.....	72
6.2.	Первоначальное включение.....	72

7.	ИЗМЕРЕНИЕ ПАРАМЕТРОВ, РЕГУЛИРОВАНИЕ И НАСТРОЙКА.....	73
8.	ТЕХНИЧЕСКОЕ ОБСЛУЖИВАНИЕ.....	74
9.	ВОЗМОЖНЫЕ НЕИСПРАВНОСТИ, СПОСОБЫ ИХ ОБНАРУЖЕНИЯ И УСТРАНЕНИЯ.....	76
10.	ТРАНСПОРТИРОВАНИЕ И ХРАНЕНИЕ.....	78
	ПРИЛОЖЕНИЕ 1. РАСПРЕДЕЛЕНИЕ СИМВОЛОВ ПО КОНТАКТАМ РАЗЪЕМА ОШ.....	79
	ПРИЛОЖЕНИЕ 2. СИСТЕМА КОМАНД МИКРОПРОЦЕССОРА.....	84
	ПРИЛОЖЕНИЕ 3. КОМАНДЫ RIM И SIM.....	94
	Часть 2	
	ПРИЛОЖЕНИЕ 4.....	4
	ПРИЛОЖЕНИЕ 5.....	45

3.055.430PЭ

1. ВВЕДЕНИЕ

Данное руководство по эксплуатации 3.055.430PЭ предназначено для изучения консольного процессора СМ 2700.2805 (WCS), принципов его работы и обеспечения правильной эксплуатации.

При изучении, эксплуатации и техническом обслуживании необходимо пользоваться следующими документами:

3.700.013PЭ - "Комплекс вычислительный СМ 1700. Руководство по эксплуатации."

3.059.006PЭ - "Вычислительная машина СМ 2700. Руководство по эксплуатации."

00076-01 34-01 - "Диагностическое программное обеспечение ЭК СМ 1700."

00076-01 13-01 - "Диагностическое программное обеспечение ЭК СМ 1700."

Сокращения, применяемые в настоящем PЭ:

С2; С2-ИС

- последовательный интерфейс стык С2-ИС для терминала и для кассетного накопителя на магнитной ленте;

БЭ

- блок элементов;

ОШ

- интерфейс "Общая шина";

ПЗУ

- постоянное запоминающее устройство;

ОЗУ

- оперативное запоминающее устройство;

УСАПП

- универсальные синхронно-асинхронные приемопередатчики;

3.055.430PЭ

- БИС - большая интегральная схема;
- CSR - регистр микрокоманд;
- в/в - ввод/вывод;
- CWR - регистр записи консоли;
- CRR - регистр чтения консоли;
- DAP - процессор;
- WCS - консольный процессор.

3.055.430PЭ

2. НАЗНАЧЕНИЕ

Консольный процессор предназначен для обеспечения основного интерфейса наладки и обслуживания ВК СМ 1700.

Агрегатный модуль предназначен для круглогодичной эксплуатации при воздействии климатических факторов согласно категории 3Б по ГОСТ 20397-82.

В процессе эксплуатации агрегатный модуль выдерживает воздействие следующих климатических и механических факторов:

температура окружающего воздуха - от +5 до +60 градусов С;

относительная влажность при температуре 30 градусов С - от 40 до 90 %;

атмосферное давление - от 84 до 107кПа;

вибрация с частотой до 25 Hz, с амплитудой - не более 0,1мм.

Нормальные условия эксплуатации агрегатного модуля:

температура окружающего воздуха - от +15 до +25 градусов С;

относительная влажность воздуха - от 45 до 75%;

атмосферное давление - от 84 до 107кПа.

3. ТЕХНИЧЕСКИЕ ДАННЫЕ

Элементная база - интегральные схемы и дискретные компоненты.

Электропитание - от источника питания постоянного напряжения СМ 1700.0304 / 2.087.065 (+5 +/- 0,05%, +15 +/- 0,05%, -15 +/- 0,05%)V.

Конструктивно агрегатный модуль WCS представляет шестислойную печатную плату типа Е3 с размещенными на ней цифровыми радиоэлектронными элементами. Плотность проводящего рисунка 2 класса по ГОСТ 23751-79.

Ток потребления - не более 7А (от +5V) и не более 0,8А (от 15V).

Габаритные размеры БЭ - не более 427x250x18 мм. Типы разъемов:

СНП59-64/94x118-23-1-8;

ОНП-КГ-56-24/50,5x7,4-8/53;

ОНП-КГ-56-40/70,5x7,4-3/53.

Масса БЭ - не более 1,0 кг.

Сигналы обмена с ОШ - в соответствии с ГОСТ 25 795-78.

Сигналы обмена с терминалом - в соответствии с ГОСТ 18 145-81.

Сигналы обмена с кассетным магнитофоном - в соответствии с ГОСТ 18 145-81.

Физическая реализация интерфейса - С2-ИС.

Максимальное удаление терминала и магнитофона соответствует ГОСТ 18 145-81.

Максимальная скорость обмена с терминалом - 9600 BOD.

Максимальная скорость обмена с кассетным магнитофоном - 38400 BOD.

Нагрузка на ОШ - не более 1 стандартной единицы нагрузки.

3.055.430PЭ

Разводка сигналов на контактах разъемов БЭ - в соответствии с приложением 1.

Консольный процессор имеет следующие технические характеристики:

длительность цикла - не более 200нс;

разрядность данных - 32;

разрядность адреса - 16;

объем программной памяти - 20К байт;

объем микропрограммной памяти - 20К слов по 24 разряда;

коэффициент технического использования - 0,98.

Система команд консольного процессора согласно приложениям 2 и 3.

4. УКАЗАНИЕ МЕР БЕЗОПАСНОСТИ

Эксплуатация консольного процессора должна выполняться инженерно-техническим персоналом, обслуживающим семейства 32-разрядных ЭВМ и имеющим удостоверение на право ее эксплуатации.

Обслуживающий инженерно-технический персонал должен регулярно проходить инструктаж и выполнять все требования электробезопасности. Не допускаются к работе лица, незнакомые с настоящим РЭ.

Категорически запрещается вставлять и извлекать блоки элементов и производить замену элементов в консольном процессоре при включенном электропитании ВК (М 1700).

Производить пайку разрешается паяльником при напряжении не более 36 В с исправной изоляцией токоведущих частей от корпуса. При пайке запрещается держать под нагревом контакты интегральных схем более 2-х с.

5. УСТРОЙСТВО И РАБОТА КОНСОЛЬНОГО ПРОЦЕССОРА

Консольный процессор обеспечивает основной интерфейс наладки и обслуживания для СЧ 1700. Блок-схема консольного процессора приведена в приложении 4 рис.1.

Основным логическим элементом в консольном процессоре является 8-битный микропроцессор с 16-ти разрядной адресацией памяти, который выполняет программу, управляющую всеми функциями консоли. С микропроцессором связано 4Кх8 бит ПЗУ, в котором записана резидентная часть программы консоли, и 16Кх8 бит ОЗУ для запоминания основной части консольной программы, загружаемой с устройства кассетной магнитной ленты во время начальной загрузки системы.

Другими логическими элементами, входящими в состав консольного процессора, являются таймер, три универсальных синхронно-асинхронных приемопередатчика (УСАПП) для взаимодействия микропроцессора с консольным терминалом, кассетным магнитофоном и линией удаленной диагностики.

В консольном процессоре информация об адресе и данных мультиплексируется и передается по двунаправленной шине АД. Эта 8-битная шина является продолжением внутренней шины адреса/данных микропроцессора (AD(I-U)).

Во время консольных операций информация об адресе сначала передается на внутреннюю шину. По сигналу чтение/запись данные передаются по шине между микропроцессором и адресуемым консольным устройством. Информация об адресе, передаваемом на шину АД - это либо младшие 8 бит адреса памяти (ПЗУ или ОЗУ), либо 8 бит адресов устройств ввода/вывода. Адресами ввода/вывода являются адреса консольных устройств, таких как внутренний таймер и УСАПП-ы. Другие устройства ввода/вывода консольного процессора - это внутренние регистры консоли. Адреса ввода/вывода, передаваемые микропроцессором на шину АД, передаются также на группу из 8-ми стандартных не мультиплексируемых адресных линий. Эти линии (A(15-08)) дублируют линии АД при обращении к устройствам ввода/вывода. Линии А служат для адресации памяти (вместе с шиной АД) и образуют старший байт адресов при обращении к памяти.

Основная часть консольного процессора расположена на плате WCS, лишь 8-битный регистр записи (CWR) и 8-битный регистр чтения (CRR) расположены на плате (DAP). Эти регистры служат для побитной передачи данных между консольным процессором и процессором. Регистр CWR адресуется и загружается консольным процессором и затем

считывается на шину "D" процессора с помощью микропрограммы. Регистр CRR адресуется и читается микропроцессором после того, как загружен микропрограммой процессора с шины "Y" процессора.

Передача информации между модулями WCS и DAP производится через 3-битную шину консоли (CONS BUS). Эта шина является расширением шины AD модуля WCS, развязанным через буфер. В результате информация об адресе и данных мультиплексируется на шине CONS BUS так же, как описано ранее для шины AD.

5.1. Микропроцессор

Микропроцессор использует восьмиразрядную шину данных и шестнадцатиразрядную шину адреса. Для него требуется всего один источник питания +5 V. Тактовая частота работы микропроцессора - 5 мHz.

Микропроцессор выполняет программу, записанную в ПЗУ и/или ОЗУ консоли. Блок-схема микропроцессора (см. приложение 4 рис.2). Контакты ввода-вывода приведены в табл.1.

выполнение инструкций в микропроцессоре состоит почти полностью из ряда операций обмена данными между микропроцессором, ПЗУ, ОЗУ и устройствами в/в консоли.

Каждая операция чтения или записи, которая происходит во время выполнения программы, называется машинным циклом. Каждая команда микропроцессора для своего выполнения требует от одного до пяти машинных циклов. В свою очередь, каждый машинный цикл содержит от трех до шести машинных фаз, где машинная фаза равна одному периоду (T) тактового импульса микропроцессора. Период импульса для микропроцессора, используемого в консольном процессоре - 200 ns.

Типы машинных циклов для микропроцессора:

выборка инструкций;

чтение из памяти;

запись в память;

чтение с устройств ввода/вывода;

запись в устройства ввода/вывода;

подтверждение прерывания;

освобождение шины.

5.055.430PЭ

В микропроцессоре используется мультиплексированная шина данных. Адрес передается по двум шинам: старший байт адреса — по шине адреса (A), а младший байт — по шине данных (AD). В начале каждого машинного цикла младший байт адреса поступает на шину данных. Этот младший байт адреса фиксируется в 8-разрядном буфере посредством подачи сигнала отпирания буфера адреса (ALE). В остальных фазах машинного цикла шина данных используется для передачи данных между микропроцессором и памятью или устройством ввода/вывода.

Микропроцессор вырабатывает также сигналы управления RD, WR, SU, ST, IO/M. Выходы ST, SU и IO/M устанавливаются к началу машинного цикла. Эти три выхода определяют тип машинного цикла. Сигнал IO/M устанавливается во время машинного цикла обращения к устройствам ввода/вывода (не памяти). ST, SU определяют машинный цикл как чтение или запись. ST, SU имеют значение:

01 — для цикла записи в память или устройства ввода/вывода;

10 — для цикла чтения из памяти и устройств ввода/вывода;

11 — для цикла выборки инструкций;

00 — для цикла освобождение шины (останов микропроцессора).

Сигнал RD, который появляется на соответствующем контакте после фазы T1 вслед за передачей адреса на шину AD, используется, чтобы выставить данные на шину из адресуемого консольного устройства.

Сигнал WR, задний фронт которого появляется во время фазы T3, используется, чтобы стробировать в адресуемом устройстве записываемые данные, установленные на шине AD.

Сигнал READY используется для синхронизации микропроцессора при его работе с медленнодействующими ОЗУ или устройствами ввода/вывода. Микропроцессор после выдачи адреса будет находиться в фазе ожидания TW (этому состоянию соответствует низкий уровень сигнала на линии READY) до тех пор, пока не поступит сигнал READY. В консольном процессоре при каждом обращении к ОЗУ или УСАПП-у сигнал READY устанавливается в логический 0, задавая время ожидания TW равное одному периоду T, чтобы обеспечить нормальную работу этих устройств.

Таблица 1
Входные-выходные контакты микропроцессора

Контакт	Функция															
A (15-08)	Выходы (три состояния). Линии адреса. Восемь старших битов адреса памяти или устройства ввода/вывода. A(15) - старший бит адреса.															
AD (07-00)	Мультиплексируемые входы-выходы (три состояния). Восемь младших битов адреса памяти или устройств ввода/вывода - в течение первой фазы (T1) машинного цикла. Данные - во время второй и третьей (T2 и T3) фаз машинного цикла.															
S1, S0	Выходы. Определяют функцию машинного цикла: <table data-bbox="630 861 1232 1043"> <thead> <tr> <th>S1</th> <th>S0</th> <th>Функция</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>останов</td> </tr> <tr> <td>0</td> <td>1</td> <td>запись</td> </tr> <tr> <td>1</td> <td>0</td> <td>чтение</td> </tr> <tr> <td>1</td> <td>1</td> <td>выборка инструкции</td> </tr> </tbody> </table>	S1	S0	Функция	0	0	останов	0	1	запись	1	0	чтение	1	1	выборка инструкции
S1	S0	Функция														
0	0	останов														
0	1	запись														
1	0	чтение														
1	1	выборка инструкции														
RD	Выход (три состояния). Чтение из памяти или с устройства ввода/вывода. Шина AD может быть использована для передачи данных. Устанавливается во время T2, снимается во время T3.															
WR	Выход (три состояния). Запись в память или вывод на устройство вывода. Устанавливается во время T2, снимается во время T3.															
IO/M	Выход (три состояния). Указывает на то, что в текущем машинном цикле производится операция чтения или записи на устройство ввода/вывода (отменяется во время чтения или записи в ОЗУ).															
HOLD	В консольном процессоре не используется.															
HOLDA	Не используется.															
INTR	Не используется.															
INTA	Не используется.															
TRAP	Не используется.															
RST 7.5 RST 6.5 RST 5.5	Входы. Используются для подачи сигналов прерываний. Производят автоматический переход на подпрограмму обслуживания прерываний.															

Продолжение табл. 1

Контакт	Функция
	вектор прерываний
RST 7.5	3C
RST 6.5	34
RST 5.5	2C
READY	Вход. Сигнал указывает, что на шину данных поступили данные. Если установлен к началу T2, то данные чтения/записи могут быть переданы во время следующего машинного цикла T3. Если не установлен, то микропроцессор ждет, что READY будет установлен для завершения операции чтения или записи.
SDV	Выход. Последовательный выход данных.
SID	Вход. Последовательный вход данных.
X1, X2	Входы. Импульсные входы (10 мHz) в консольном процессоре.
CLK	Выход. Выход внутренних тактовых импульсов (5 мHz) в консольном процессоре.
RST IN	Вход. Инициализация. Сбрасывает счетчик адреса на 0.
RST OUT	Выход. Указывает, что микропроцессор сброшен.

5.2. Универсальные синхронно-асинхронные приемо-передатчики (УСАПП-ы)

Консольный процессор содержит три УСАПП-а: один УСАПП соединен с консольным терминалом, второй - с каскадным магнитофоном, третий - с модемом линии удаленной диагностики. Каждый УСАПП работает в асинхронном режиме передачи данных по последовательной линии в/из устройства, к которому он присоединен. Блок-схема УСАПП-а см. приложение 4 рис.3. Табл.2 определяет входо-выходные контакты.

5.2.1. Основные операции

Одна из основных функций УСАПП-а - прием символьных кодов в параллельном формате и преобразование их в последовательный поток данных для передачи.

Другая функция - прием потока последовательных данных и преобразование его в параллельные коды символов. Преобразование данных из параллельной формы в последовательную производится следующим образом. Вначале данные загружаются в сдвиговый регистр. Содержимое сдвигового регистра сдвигается на один разряд при поступлении каждого тактового импульса; данные на выходе сдвигового регистра будут иметь последовательную форму. Биты слова данных, загружаемого в регистр, появляются на выходе друг за другом, начиная с младшего.

Для того, чтобы принять данные в последовательной форме и преобразовать их в параллельную форму, необходимо выполнить действия, обратные по отношению к описанным выше. Данные, поступающие в последовательной форме, вводятся бит за битом в сдвиговый регистр. После заполнения сдвигового регистра данные из него в параллельной форме передаются в микропроцессорную систему.

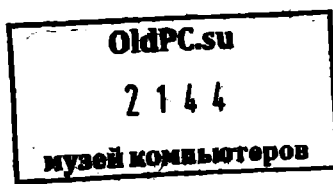
Расположение битов и адресов регистров ввода/вывода, которые могут управляться микропроцессором в УСАПП-е, см. в приложении 4 рис.4.

Адресуемые регистры УСАПП-ов задаются двумя адресными входами А1 и А0, которые присоединены к двум линиям А09 и А08 микропроцессора. Линии А выбирают адрес регистра данных, регистра режима, регистра команд или регистра состояния, когда произведена выборка УСАПП-а. Выборка соответствующего УСАПП-а описана ниже.

Должен ли адресуемый регистр считываться или записываться, определяется входом УСАПП-а WR. Сигнал 8085 WRT CYC устанавливается на этом входе во время операции записи на шине AD. Сигнал микропроцессора 8085 WRT CYC устанавливается на линии состояния (S1).

Микропроцессор загружает символ в регистр данных УСАПП-а, чтобы передать этот символ по последовательной линии; при записи по адресу регистра данных символ загружается в регистр хранения передатчика (TX) в УСАПП-е.

Если модем устанавливает вход CTS (готов к передаче) и если микропроцессор установил бит возможности передачи TXEN в регистре команд, УСАПП взводит бит готовности передачи TX RDY в регистре состояния. Проверив готовность передачи, микропроцессор посылает символ в регистр данных УСАПП-а, откуда он поступает в регистр хранения, и УСАПП устанавливает RTS (обращение к модему на передачу). Пока идет передача и символ, побитно через



регистр сдвига пересылается на линию, CTS сброшен, а в регистре состояния УСАПП-а сброшен TX RDY. Когда передача закончится, TX RDY устанавливается, т.е. УСАПП готов передать следующий символ.

УСАПП-у разрешено получение данных по последовательной линии, когда вход DCD (от модема) установлен, а также установлены бит разрешения приема RX EN в регистре команд. Всякий раз, когда полный символ собран в сдвиговом регистре УСАПП-а, он передается в регистр хранения приемника RX и устанавливается RX RDY (бит регистра состояния), чтобы указать, что микропроцессор может принять данные.

Чтобы принять переданный символ, микропроцессор устанавливает на шине AD адрес регистра данных УСАПП-а. При этом УСАПП передает содержимое регистра хранения RX на вход регистра данных и, таким образом, на шину AD консольного процессора. RX RDY взводится всякий раз, когда следующий символ собирается в сдвиговом регистре RX и передается в регистр хранения RX.

При приеме/передаче данных могут обнаруживаться ошибки пересылки (OVN), ошибки четности (PAR), ошибки стоп-бита (STB). Каждый тип ошибки устанавливает соответствующий бит в регистре состояния УСАПП-а, который может быть прочитан микропроцессором. Ошибка пересылки указывает, что данный символ не прочитан микропроцессором до того, как новый символ был собран и загружен в регистр хранения приемника RX. Ошибка четности указывает прием неверной четности. Ошибка стоп-бита указывает, что принятый знак не охвачен правильным числом стоп-битов; (передатчик добавляет к каждому передаваемому слову данных стартовый бит и стоп-бит; эти биты используются приемником для установления границ - начала и конца передаваемого слова данных). Биты ошибок очищаются, когда приемник отключается (бит RX EN в регистре команд сброшен), или при помощи команды очистки ошибок (в регистре команд установлен бит CLR ERR).

Флаг ошибки четности УСАПП-а три помощи консольной программы микропроцессора не проверяется.

Рабочий режим УСАПП-а (длина символа, число стоп-битов и т.д.) определяется соответствующими битами регистров режимов УСАПП-а.

5.2.2. Импульсы синхронизации

Все три УСАПП-а в консольном процессоре используют источник внешнего синхросигнала. Таким образом, контакты передающих и принимающих входов синхронизации для всех УСАПП-ов связаны между собой. В биты 7 и 0 регистра режима записывается двоичное число 10. Таким образом,

3.055.430PЭ

УСАПП делит частоту входных импульсов синхронизации на 16. Синхроимпульсы длительностью 200 нс делятся на 4 (CLK BAUD GEN) и, дополнительно откорректированные (63/64 CLK BAUD GEN) используются для того, чтобы стробировать счетчик. Выходы счетчика, после деления на 16, на УСАПП-е, обеспечивают импульсы скорости передачи в BOD - 38400 (для CM5218), 9600, 2400, 1200, 300. Кроме того, счетчик обеспечивает выход REC 3085 REF, который используется, чтобы инициализировать цикл восстановления для ОЗУ микропроцессора, основной период которого длится около 12,8 мкс.

УСАПП для кассетного магнитофона работает со скоростью 38400 BOD. Синхроимпульс для УСАПП-а, связанного с линией удаленной диагностики (REMOTE BAUD CK), также имеет скорость 38400 BOD, но только во время работы с АРТ. Во время проведения удаленной диагностики скорость работы выборочно устанавливается на 300 или 1200 BOD. Импульсы УСАПП-а консольного терминала (TER BAUD CK) также включаются выборочно и работа может вестись со скоростями 9600, 2400, 1200 и 300 BOD.

5.2.3. Передача данных терминала и ленты

Пути, по которым передаются данные в/из терминала или ленты, проходят через УСАПП-ы и различаются в зависимости от того, работает консольная программа в консольном режиме или в программном режиме.

В консольном режиме данные передаются только под управлением консольной программы. Это означает, что программа определяет, когда передать символы (например, подсказку или сообщения об ошибках на терминал) или когда читать и обрабатывать принятые символы. Программа проверяет бит УСАПП-а TX RDY, чтобы определить, когда следующий передающийся символ может быть загружен в регистр данных. Подобным образом она проверяет бит УСАПП-а RX RDY, чтобы определить, когда принятый символ может быть прочитан из регистра данных. Программа проверяет биты TX RDY, считывая регистр состояния битов готовности консольного процессора. Регистр состояния битов готовности имеет адрес 20. Он реализован аппаратно и считывается через мультиплексор ПЗУ. Если готовность есть, то символ соответственно передается или принимается консольной программой.

В программном режиме передачи данных терминала и ленты управляются программой консоли, но данные передаются или принимаются центральной частью вслед за прерываниями микропроцессора, вызванными командами MFPR и MTRK, в работающей программе центральной части. Чтобы управлять генерацией прерываний, консольная программа опрашивает регистры состояния УСАПП-ов для проверки би-

3.055.450PЭ

тов TX RDY и RX RDY (холостой цикл). Если один из битов любого УСАПП-а установлен, программа взводит соответствующий бит в 8-битном регистре сводок прерываний.

Регистр сводок прерываний - программный регистр, содержащийся в консольной программе. Формат его показан на рис.5 приложение 4. За каждым устройством в нем зарезервированы 4 бита - два бита на каждые TX RDY и два бита на каждые RX RDY (в данном случае не различаются терминал консоли или удаленная диагностика). Бит IS - установки прерывания и бит IE - разрешения прерывания.

С каждым из 4-х битов регистра сводок прерываний (IS), устанавливаемым в результате установки бита готовности УСАПП-а, ассоциирован бит разрешения прерывания (IE). За каждым устройством со стороны процессора закреплены две пары программных регистров - данных и управления/состояния.

Форматы регистров управления/состояния устройств консоли и ленты см. в приложении 4 рис.6 и 7.

Регистры управления/состояния передачи или приема для терминала - TXCS и RXCS. Для ленты - CSTS и CSRS (другие регистры на рисунках 6 и 7 - это буферные регистры передачи и приема данных для терминала и ленты).

Каждый бит разрешения прерывания (IE) устанавливается по команде MTPR (с битом U6 в данных регистра управления/состояния, равным 1), адресуемой соответствующий регистр управления/состояния передачи или приема терминала или ленты. Указанный бит в регистре управления/состояния определяет соответствующий бит в регистре сводок прерываний. Например, когда выполняется адресуемая регистр управления/состояния инструкция MTPR, микропрограмма процессора посылает бит прерывания от путей данных как часть пакета данных к консольному процессору через шину консоли. Консольная программа устанавливает при этом бит разрешения прерывания (IE) в регистр сводок прерываний консоли. Передача групп данных между микрокомандами процессора и консольным процессором описана ниже. Если бит разрешения прерывания в регистре сводок прерываний установлен при помощи MTPR, он взведет бит ожидания прерывания в другом программном регистре консоли - регистре приоритетов прерываний, если установлен соответствующий бит сводок прерываний (IS). Установка любого бита в регистре приоритетов вызовет запрос прерывания микропроцессора, генерируемый консольной программой. Консольная программа генерирует запрос прерывания по первому же установленному (одному из 4-х) биту регистра приоритетов прерываний (см. рис.6 приложение 4). Устанавливаемый бит зависит от того, какой бит сводки прерываний (и бит разрешения прерываний) установлен, т.е. какой тип запроса на передачу данных был сделан (запись или чтение с терминала; запись или чтение с ленты). Консольная программа устанавливает тогда сигнал CONS ATTN, чтобы прервать работу программы процессора. После установки CONS ATTN консольная программа возвраца-

ется в режим холостой программной петли, в которой продолжает проверять биты TX RDY и RX RDY УСАПП-ов. Всякий запрос прерывания (CONS ATTN) на передачу данных УСАПП-а обслуживается микропрограммой процессора. Пакет данных (один байт) посылается из путей данных к консольному процессору, чтобы подтвердить запрос. В ответ консольная программа снимает CONS ATTN и возвращает содержимое регистра приоритета прерываний другой группой данных (снова один байт), так, что микропрограмма может определять - запрос какого УСАПП-а вызвал прерывание.

Если имеется более одного запроса на передачу от УСАПП-а, консольная программа также подтверждает сигнал CONS ATTN. Когда микропрограмма центральной части получает содержимое регистра приоритетов прерываний, она инициализирует обращение к подпрограмме обслуживания прерывания командного уровня для запроса передачи данных УСАПП-а с наивысшим приоритетом. Считывание с ленты имеет наивысший приоритет, а операция передачи с терминала - наименьший.

Во время подпрограммы обслуживания прерываний командного уровня выполняется MTPR, чтобы переслать передаваемые данные к терминалу или ленте, или выполняется MFPR, чтобы переслать получаемые данные с терминала или с ленты. MTPR и MFPR адресуют регистр буфера данных. Регистры буфера данных для ленты - CSRD и CSTD; для терминала - TXDB, RXDB. В соответствии с рис. 6, 7, передаваемый символ (один байт) содержится в младших восьми битах данных регистра для TXDB и CSTD. Когда программой выполняется MTPR, ссылающаяся на один из этих буферных регистров, символ передается от процессора к консольному процессору (как часть пакета данных) микропрограммой центральной части. Консольная программа грузит посланный к TXDB символ в псевдобуфер передаваемых данных, откуда потом загружает символ в регистр данных соответствующего УСАПП-а для передачи местному или удаленному терминалу. Символ, посланный к CSTD, загружается прямо в регистр данных УСАПП-а для кассетного магнитофона. К тому же, для обоих регистров TXDB и CSTD консольная программа обнуляет соответствующий бит сводок прерывания в регистре сводок прерываний и соответствующий бит "ожидания прерывания" в регистре приоритетов прерываний. На этом обслуживание запроса прерывания заканчивается. Как только регистр данных УСАПП-а загружен, передаваемый символ высылается последовательно к терминалу или ленте. Поле "выборки" в регистре TXDB равно нулю при пересылке передаваемых данных. Оно также может быть равно F, чтобы указать разнообразные сообщения между микропрограммой центральной части и консольной программой. В этом случае поле "данных" определяет операцию, которая выполняется консольной программой (начальная загрузка и т.д.). Команда MFPR, обращаясь к RXDB и CSRD, заставляет консольную программу (после приема пакета командных данных от микропрограммы центральной части) читать как программный буфер данных, который содержит получаемый символ, так и программный регистр управления/состояния для УСАПП-а (принятый символ был считан из регистра данных УСАПП-а и

3.055.430P3

записан в программном буфере данных вслед за установкой бита RX RDY УСАПП-а). Принятый символ и программный регистр управления/состояния посылаются потом в микропрограмму центральной части, как часть двухбайтового набора данных. Микропрограмма центральной части распаковывает информацию и собирает ее в правильном формате для MFPR. Принятый символ размещается в младших восьми битах RXDB или CSRD. К тому же, бит ошибки в RXDB или CSRD (бит 15) станет равным 1, когда установлен любой из битов ошибки в информации управления/состояния, поступающей от консольной программы (это обнаруженные УСАПП-ом биты ошибок пересылки и останова). Ошибки четности, обнаруженные УСАПП-ом не будут вызывать установку бита ошибки MFPR. Это обусловлено тем, что бит ошибки УСАПП-а маскируется консольной программой всякий раз, когда регистр состояния УСАПП-а грузится в программный регистр управления/состояния.

Программный регистр управления/состояния также посылается консольной программой в микропрограмму центральной части, когда выполняется MFPR, адресующая TXCS, RXCS, CSRS, или CSTS. В этом случае микропрограмма центральной части использует бит TX RDY или RX RDY в информации управления/состояния, чтобы установить DONE (бит 07) в данные регистра при его сборке (бит DONE указывает программе состояние готовности приема или передачи для терминала или ленты). Состояние соответствующего бита разрешения прерывания становится также частью данных регистра, собираемых для MFPR. Этот бит не является частью данных управления/состояния, посылаемых консольной программой процессору в момент, когда он был установлен или очищен при помощи MTPR.

Таблица 2

Входные-выходные контакты УСАПП

Контакты	Функция																				
(7D-UD)	Вход/выходная шина данных (три состояния). Передает данные чтения/записи плюс информацию о команде и состоянии. 7D - старший бит.																				
A1, AU	Входы адреса. Выбирают внутренние регистры.																				
WR	Вход. Определяет чтение, когда сброшен.																				
EN	Вход разрешения для микросхемы. Включает выполнение операции, заданной A1, AU и WR:																				
	<table border="1"> <thead> <tr> <th>EN</th> <th>A1</th> <th>AU</th> <th>WR</th> <th>Функция</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>чтение регистра данных (прием, сохраняя).</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>1</td> <td>запись в регистр данных (передача, сохраняя).</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> <td>чтение регистра</td> </tr> </tbody> </table>	EN	A1	AU	WR	Функция	1	0	0	0	чтение регистра данных (прием, сохраняя).	1	0	0	1	запись в регистр данных (передача, сохраняя).	1	0	1	0	чтение регистра
EN	A1	AU	WR	Функция																	
1	0	0	0	чтение регистра данных (прием, сохраняя).																	
1	0	0	1	запись в регистр данных (передача, сохраняя).																	
1	0	1	0	чтение регистра																	

Продолжение табл. 2

Контакты	Функция				
	1	0	1	1	состояния. запись SIN1/SYN2/ DLE в регистры.
	1	1	0	0	чтение регистров режима 1 и 2.
	1	1	0	1	запись в регистры режима 1 и 2.
	1	1	1	0	чтение регистра команд.
	1	1	1	1	запись в регистр команд.
	0	-	-	-	нет передачи - шина данных отклю- чена (третье состояние).
TX RDY	Выход готовности передачи. Регистр хранения передачи готов принять символ. Сбрасывается когда регистр хранения загружен.				
RX RDY	Выход готовности приема. Регистр хранения приема содержит символ. Сбрасывается, когда регистр хранения прочитан.				
TXD	Выход передатчика на последовательную линию.				
RXD	Вход приемника с последовательной линии.				
DSR	Вход готовности набора данных.				
DCD*	Вход, определяющий носитель данных. Должен быть установлен, чтобы приемник работал.				
CTS*	Вход должен быть установлен, чтобы передатчик работал.				
DTR	Данные на выходе готовы.				
DSCRG	Не используется в консольном процессоре.				
BRCLR	Не используется в консольном процессоре.				
TXC	Вход. Синхронизация передатчика.				
RXC	Вход. Синхронизация приемника.				
RESET	Вход сброса. Сброс в исходное состояние. Режим очистки регистров команд и состояния.				

* Вход с модема. Всегда установлен (контакт заземлен) в
УСАПП-е терминала консоли (местный) и УСАПП-е кассетного
магнитофона.

5.3. Таймер

Одним из устройств, используемых в консольной подсистеме, является таймер. Благодаря таймеру можно аппаратным, а не программным путем "генерировать" временные задержки. В данном случае используется таймер, который представляет собой большую интегральную схему.

В состав микросхемы входят 5 16-разрядных универсальных счетчиков. В качестве входных сигналов для отдельных счетчиков можно выбрать внутренние источники частотных сигналов или внешние с выводов микросхемы. При этом имеется возможность программного задания полярности входного сигнала, т.е. выбора того, какой уровень сигнала, высокий или низкий, соответствует активному состоянию сигнала. Имеется возможность как аппаратного, так и программного стробирования каждого счетчика. Выходы каждого счетчика обеспечивают либо импульсные сигналы, либо сигналы с постоянным уровнем. Счетчики можно запрограммировать для счета либо в прямом, либо в обратном направлении и в двоичном или двоично-десятичном коде. Результат можно считывать, не нарушая при этом процесса счета. Счетчики в любом количестве могут объединяться для увеличения числа разрядов вплоть до 80.

5.3.1. Описание сигналов интерфейса

В табл. 3 приведены сигналы интерфейса для таймера и их обозначение.

Таблица 3
Перечень сигналов интерфейса таймера

сигнал	условное обозначение	тип	выводы
5 вольт	VCC	питание	1
Земля	VSS	питание	1
Кварцевый генератор	X1, X2	вход	2
Чтение	RD	вход	1
Запись	WR	вход	1
Выбор микросхемы	CS	вход	1

Продолжение табл. 3

сигнал	условное обозначение	тип	вывод
Управление/ данные	CD	вход	1
Источник N	SRC	вход	5
Строб N	GATE	вход	5
Шина данных	DB	вход/вывод	16
Выходная частота	FOUT	вывод	1
Выход N	OUT	вывод	5

X1, X2

- X1 и X2 представляют собой сигналы связи с внешним кристаллом, используемым для задания частоты внутреннего генератора.

FOUT (вывод частотного сигнала)

- выходной сигнал FOUT снимается с 4-разрядного счетчика, который может быть запрограммирован на деление своего входного значения на любое целое число от 1 до 16 включительно. Входной сигнал для счетчика снимается с любого из 15 источников, включая внутренний генератор частоты. Блокированием сигнала FOUT можно управлять с помощью программного обеспечения. В случае блокировки выход FOUT находится в состоянии с низким импедансом относительно земли. Управление выбором различных значений сигнала FOUT осуществляется записью необходимых данных в регистр главного режима работы. После включения питания сигнал FOUT представляет собой частоту, которая составляет 1/16 от частоты генератора.

GATE1-GATE5 (входные сигналы)

- входные строб-сигналы могут быть использованы для управления работой отдельных счетчиков, и в этом случае они определяют момент времени, когда счет может быть продолжен. Один и тот же входной строб-сигнал может управлять максимум 3-мя счетчиками. Выходы строб-сигналов могут быть также выбраны в качестве источников счета для любого из счетчиков и для делителя сигнала FOUT. Активная полярность для выбранного строб-сигнала программируется в каждом счетчике. Можно использовать как стробирование по уровню, так и стробирование по фронту. Если таймер, как это имеет место в консоли, работает с 8-разрядной внешней шиной данных, то активный уровень входного строб-сигнала определяется вспомогательным входным сигналом.

SRC1-SRC5 (входные сигналы)

- входы SOURCE обеспечивают внешние сигналы, которые могут подсчитываться любым из счетчиков. Любая линия SOURCE может быть подключена к любому или ко всем счетчикам и к делителю сигнала FOOT. Активная полярность для выбранного входного сигнала SRC программируется в каждом счетчике.

OUT1-OUT5 (выходные сигналы)

- каждый сигнал OUT непосредственно связан с отдельным счетчиком. В зависимости от конфигурации счетчика сигнал OUT может быть импульсным прямоугольным или иметь сложный цикл. Полярности OUT программируются для каждого счетчика в отдельности. Выходная схема обнаруживает состояние счетчика, при котором все разряды счетчика равны нулю, при условии, что отсутствовала повторная инициализация. Эта информация используется для генерирования сигнала выбранной формы. Режим вывода для счетчиков 1 и 2 исключает использование обычного режима вывода и обеспечивает истинное значение сигнала OUT в том случае, когда значение счетчика совпадает со значением регистра сигнализации.

DB0-DB7, DB8-DB15 (шина данных, вход/выход)

- линии шины данных используются для организации обмена данными с процессором. Высокий уровень сигнала на линии шины данных соответствует 1, низкий - 0. Эти линии действуют как линии ввода при активном уровне сигналов RD и CS. Если CS не активен, то эти линии переходят в состояние с низким импедансом. После запуска или установки в исходное состояние шина данных конфигурируется для 3-разрядного варианта шины и используются только выходы с DB0 по DB7 (в консольном процессоре обеспечивается только 8-битная конфигурация). DB0 - младший разряд, DB7 - старший. В режиме работы с 8-разрядной шиной необходимо, чтобы DB15-DB16 находились в состоянии с высоким логическим уровнем всегда, когда уровень сигналов CS и WR является активным.

CS (выбор микросхемы, входной сигнал)

- этот сигнал является активно низким и является разрешением для операций READ (чтение) и WRITE (запись) на шине данных. При высоком логическом уровне сигнала "выбор микросхемы" сигналы READ и WRITE игнорируются. Первый после запуска сигнал CS используется для сброса микросхемы в исходное состояние.

RD (чтение, входной сигнал)

- является активно-низким, зависит от сигнала "выбор микросхемы" и определяет, что внутренние данные подлежат передаче на шину данных. Источник данных определяется адресуемым портом, а для операции чтения через порт данных определяется содержимым регистра указателя данных. Сигналы WR и RD являются взаимоисключающими.

3.055.430PЭ

WR (запись, входной сигнал)

- является активно-низким, зависит от сигнала "выбор микросхемы" и определяет, что информация должна быть передана с шины данных во внутреннюю ячейку. Место назначения определяется адресуемым портом, а для операций записи через порт данных определяется содержимым регистра указателя данных.

CS/D (управление/данные, входной сигнал)

- сигнал управление/данные выбирает источник и приемник для операций чтения и записи, осуществляемых с помощью шины данных. Операции записи сигналов управления загружают регистр команд и указатель данных. Операции чтения сигнала управления осуществляют вывод содержимого регистра состояния. Операции чтения и записи данных работают со всеми остальными внутренними регистрами. Существует возможность косвенной адресации порта данных с помощью регистра указателя данных.

3.3.2. Функциональное описание

На структурной схеме БИС таймера (см. приложение 4 рис.8) показаны сигналы интерфейса. Все управляющие регистры и регистры данных подключены к общей внутренней 16-разрядной шине. Если внешняя шина имеет 8 разрядов, то внутренняя 16-разрядная информация мультиплексируется на выходы D20 - D27 шины данных, соответствующие младшим разрядам.

Внутренний генератор представляет собой источник частотных сигналов, используемых в качестве входных сигналов счетчика. Задание частоты генератора осуществляется через выходы X1 и X2 с помощью внешней реактивной схемы (кристалл). Входной сигнал генератора делится при помощи делителя частоты и таким образом обеспечивается несколько подчастот. Одна из получаемых в результате деления частот может быть выбрана в качестве входного сигнала делителя сигнала FOUT и затем снимается с интерфейсного вывода FOUT.

Микросхема адресуется через 2 порта: порт сигналов управления и порт данных. Порт сигналов управления обеспечивает возможность прямого доступа к регистру состояния и к регистру команд, а также позволяет пользователю модифицировать регистр указателя данных.

Порт данных используется для связи со всеми остальными адресуемыми внутренними ячейками. Регистр указателя данных управляет адресацией порта данных.

Через порт данных осуществляется доступ к регистру главного режима работы и к пяти регистрам режима работы счетчиков. Регистр главного режима работы управляет программируемыми функциями. Каждый из пяти универсальных

5.055.450P3

счетчиков представляет собой 16-разрядный счетчик и независимо управляется соответствующим ему регистром режима работы счетчика. С помощью этого регистра можно программным путем выбрать один из 16 источников в качестве входа счетчика, а также выбирать режимы стробирования и повторения, выбирать варианты счета в прямом или обратном направлении, двоичном или двоично-десятичном коде, выбирать активно-высокую или активно-низкую полярности входного и выходного сигналов.

С каждым счетчиком связаны регистр загрузки и регистр хранения, доступ к которым осуществляется через порт данных. Регистр загрузки используется для того, чтобы автоматически перезагружать в счетчик заранее определенное значение, управляя таким образом эффективным периодом счета. Регистр хранения используется для того, чтобы сохранять значения счетчиков без нарушения процесса счета, допуская тем самым считывание промежуточных значений счета.

Все пять счетчиков имеют одну и ту же основную управляющую логику и регистры управления. Счетчики 1 и 2 имеют дополнительные регистры сигнализации и связанные с ними компараторы, а также дополнительную логику, необходимую для работы в режиме счета 24-часового времени суток.

Каждый универсальный счетчик имеет свой вывод выходного сигнала. Этот вывод может быть выключен, когда выходной сигнал не используется. Существует возможность конфигурировать как входной сигнал, так и сигнал стробирования для отдельных счетчиков. Это позволяет с помощью отдельного вывода стробирования управлять более чем одним счетчиком. Действительно, отдельный вывод может быть строб-сигналом для одного счетчика и в то же время источником сигналов счета для другого счетчика.

Работа счетчиков, регистров и портов таймера программируется. Коды соответствующих инструкций приведены ниже. Все инструкции имеют свою мнемонику, такую как LOAD, ARM, DISARM и др.

Счетчик должен быть взведен посредством одной из команд ARM перед тем как счет может быть продолжен. После того, как счетчик взведен, процесс счета может быть в дальнейшем инициализирован или заблокирован с помощью аппаратных средств стробирования. Команды ARM и DISARM позволяют выполнить в некоторых режимах программное стробирование процесса счета.

Команда LOAD вызывает перезагрузку счетчика значением либо из связанного с этим счетчиком регистра загрузки, либо из регистра хранения.

Команда DISARM блокирует дальнейший счет. Заблокированный счетчик может быть перезагружен с помощью команды LOAD, инкрементирован или декрементирован с помощью команды STEP, считан с помощью команды SAVE. Про-

цесс счета можно возобновить с помощью команды ARM.

Команда SAVE передает содержимое счетчика в регистр хранения и служит для сохранения накапливаемого значения счета, чтобы оно могло быть считано в любой момент времени.

Другие команды позволяют: изменять содержимое счетчика на единицу, выполнять программный сброс, сбрасывать и устанавливать специальные разряды в регистре главного режима работы, загружать регистр указателя данных и т.д.

5.3.3. Регистры порта сигналов управления

Передачи через порт сигналов управления позволяют осуществлять прямой доступ к регистру команд при записи и к регистру состояния при чтении (доступ к регистрам см. приложение 4, рис.9). Передачи в порт сигналов управления и из него всегда выполняются по 8 разрядов за одну передачу. В результате каждого доступа к порту сигналов управления выполняется передача данных между регистром команд (запись) или регистром состояния (чтение) и выводами DB0-DB7 шины данных.

5.3.4. Регистры команд

Регистр команд обеспечивает прямое управление каждым из 5-ти счетчиков и доступом через порт данных, позволяя обновлять содержимое регистра указателя данных. В табл. 4. представлен перечень команд. Шесть команд используются для прямого программного управления процессом счета. Каждая из 6-ти команд имеет поле S. Каждый разряд поля S соответствует одному из 5-ти счетчиков. Если некоторый разряд поля S равен 1, то операция выполняется с соответствующим счетчиком, если S = 0 - никакой операции с данным счетчиком не выполняется.

Таблица 4

Набор команд таймера

Код команды								Название и описание команды
S7	S6	S5	S4	S3	S2	S1	S0	
0	0	0	E2	E1	G4	G2	G1	LOAD DATA:
								загрузка в регистр указателя

Продолжение табл. 4

Код команды								Название и описание команды
C7	C6	C5	C4	C3	C2	C1	C0	
								данных содержимого полей E и G.
0	0	1	S5	S4	S3	S2	S1	ARM: инициирование процесса счета для всех выбранных счетчиков.
0	1	0	S5	S4	S3	S2	S1	LOAD: загрузка содержимого указанного источника во все выбранные счетчики.
0	1	1	S5	S4	S3	S2	S1	LOAD - AND ARM: загрузка и взведение всех выбранных счетчиков.
1	0	0	S5	S4	S3	S2	S1	DISARM - AND SAVE: блокировка счета и сохранение содержимого всех выбранных счетчиков.
1	0	1	S5	S4	S3	S2	S1	SAVE: сохранение содержимого выбранных счетчиков в регистрах хранения.
1	1	0	S5	S4	S3	S2	S1	DISARM: блокировка счет во всех выбранных счетчиках.
1	1	1	U	1	N4	N2	N1	SET: установка выхода OUT счетчика N.
1	1	1	U	0	N4	N2	N1	CLEAR: сброс выхода OUT счетчика N.

Продолжение табл. 4

Код команды								Название и описание команды
C7	C6	C5	C4	C3	C2	C1	C0	
1	1	1	1	0	N4	N2	N1	STEP: продвижение на один шаг счетчика N.
1	1	1	0	1	0	0	0	DISABLE DATA: установка разряда MM14 регистра главного режима работы счетчика (блокировка последовательности указателя данных).
1	1	1	0	1	1	1	0	GATE OFF FOUT: установка разряда MM12 регистра главного режима работы счетчика (блокировка сигнала F OUT).
1	1	1	0	1	1	1	1	ENABLE 16-BIT DATA BUS: установка разряда MM13 регистра главного режима работы счетчика.
1	1	1	0	0	0	0	0	ENABLE DATA: сброс разряда MM14 регистра главного режима работы счетчика (инициализация последовательности указателя данных).
1	1	1	0	0	1	1	0	GATE ON FOUT: сброс разряда MM12 регистра главного режима работы счетчика (разрешение выхода

Продолжение табл. 4

Код команды								Название и описание команды
C7	C6	C5	C4	C3	C2	C1	C0	
								FOOT).
1	1	1	0	0	1	1	1	ENABLE 8-BIT DATA BUS: сбор разряда MM3 регистра глазного режима работы счетчика (звод режима работы с 8-разрядной шиной).
1	1	1	1	1	1	1	1	MASTER RESET: общий сброс.

5.5.5. Регистр указателя данных

Загрузка 16-разрядного регистра указателя данных выполняется по соответствующей команде, поданной через порт управления в регистр команд. Указатель данных состоит из 3-х разрядного указателя группы, 2-х разрядного указателя элемента и 1-разрядного указателя байта (рис.10 приложение 4). Разряд указателя байта показывает, какой байт 16-разрядного регистра должен передаваться через порт данных при выполнении очередного доступа. Каждый раз при загрузке регистра указателя данных разряд указателя байта устанавливается в 1 (т.е. будет передаваться младший байт). Разряд указателя байта перебрасывается после каждой передачи 8-разрядных данных при работе с 8-разрядной шиной данных. Указатели группы и элемента выбирают тот внутренний регистр, к которому необходимо осуществить доступ. Разряд указателя байта доступен для процессора в виде разряда регистра состояния. Произвольный доступ к любой внутренней ячейке данных можно выполнить посредством загрузки указателя данных с помощью команд, показанных в табл. 5, (командные коды от 00 до 1F грузят регистр указателя данных) с последующей инициализацией чтения или записи данных. Коды указателя приводятся в шестнадцатеричной системе.

Таблица 5

Команды загрузки указателя данных

	Цикл элемента			Цикл хранения
	Регистр режима	Регистр загрузки	Регистр хранения	Регистр хранения
Счетчик 1	FFD1	FFD9	FF11	FF19
Счетчик 2	FFD2	FFDA	FF12	FF1A
Счетчик 3	FFD3	FFDB	FF13	FF1B
Счетчик 4	FFD4	FFDC	FF14	FF1C
Счетчик 5	FFD5	FFDD	FF15	FF1D

Регистр главного режима = FF17

Регистр сигнализации 1 = FF07

Регистр сигнализации 2 = FF0F

Для реализации быстрого доступа к различным внутренним регистрам существует возможность автоматического задания последовательности операций указателя данных. Эта функция инициализируется установкой в 0 разряде MM14 регистра главного режима работы. Существует несколько типов задания последовательности в зависимости от занесенного начального значения указателя данных.

цикл элемента

- если E1=0, E2=0 и G4, G2, G1 определяют группу счетчика, то указатель данных перейдет к циклу элемента. Значение поля элемента будет автоматически проходить через три значения 00, 01 и 10, начиная с введенного значения. При выполнении перехода от 10 к 00 (E2, E1 = 11), значение поля группы циркулирует в пределах пяти кодов группы счетчика.

цикл хранения

- если E2, E1=11 и выбирается группа счетчика, то в процессе установления последовательности участвует только поле группы. Такой автоматический перебор называется циклом хранения. Он позволяет осуществить последовательный доступ к регистрам хранения, обходя при этом регистры режима работы и регистры загрузки.

3.055.430PЭ

Дополнительные циклы

- третий тип последовательности представляет собой цикл управления, в консоли он не используется. Если $G4, G2, G1 = 111$ и $E2, E1 = 11$, то не будет выполняться никакое приращение и только регистр состояния будет доступен через порт данных.

Для всех этих авторежимов характерно следующее: если используется 3-разрядная шина данных, то указатель байта перебрасывается после каждой передачи данных для того, чтобы младший и старший байты были переданы до наращивания значения полей элемента или группы.

5.3.6. Регистр состояния

8-разрядный регистр состояния (рис.11 приложение 4) допускающий только считывание, хранит состояние разряда "указатель байта" регистра указателя данных и состояние сигнала OUT для каждого из 5-ти универсальных счетчиков. Разряд OUT отражает состояние вывода OUT. Для счетчиков 1 и 2 вывод OUT идентифицирует выходной сигнал компаратора, если компараторы разблокированы. Обычно доступ к регистру состояния осуществляется посредством чтения порта сигналов управления.

5.3.7. Регистры порта данных. Логические группы счетчиков

Как показано на рис.12 и 13 (приложение 4) каждая из 5-ти логических групп счетчиков состоит из 16-разрядного универсального счетчика с соответствующей управляющей и выходной логикой, 16-разрядного регистра загрузки, 16-разрядного регистра хранения и 16-разрядного регистра режима работы. Кроме того, группы счетчиков 1 и 2 также включают 16-разрядные компараторы и 16-разрядные регистры сигнализации. Управление функциями сравнения/сигнализации осуществляется с помощью регистра главного режима работы. Функционирование регистров режима работы счетчиков идентично для всех пяти счетчиков. Консольный процессор может осуществлять как чтение, так и запись во все регистры логических групп счетчиков через порт данных, но не имеет прямого доступа к самому счетчику.

16-разрядный регистр загрузки, работающий в режиме чтения/записи, используется для управления периодом универсального счетчика. Любое 16-разрядное значение может быть записано в регистр загрузки. Это значение затем может передаваться в счетчик каждый раз, когда происходит событие TC. TC определяется как тот момент времени,

когда содержимое счетчика становится равным нулю, если никакое внешнее значение не было передано в счетчик. Таким образом частота события ТС представляет собой частоту выходного сигнала, заданную значением, хранящееся в регистре загрузки. Во всех режимах содержимые либо регистра загрузки, либо регистра хранения будут передаваться в счетчик при наступлении события ТС. Если значения должны накапливаться в счетчике, то все разряды регистра загрузки заполнятся нулями.

16-разрядный регистр хранения, работающий в режиме чтения/записи, является регистром двойного назначения. Его можно использовать в качестве регистра загрузки, а также можно использовать для хранения накапливаемых значений счетчика, подлежащих передаче в процессор.

5.3.8. Регистры режима работы счетчиков

16-разрядный регистр режима работы счетчика, работающий в режиме чтения/записи, управляет функциями стробирования, счета, а также функциями выбора выходного сигнала и источника входных сигналов в пределах каждой логической группы счетчика.

На рис.14 (приложение 4) показано назначение разрядов регистров режима работы счетчика.

5.3.9. Регистры сигнализации и компараторы

В группах счетчиков 1 и 2 содержатся 16-разрядные регистры сигнализации и 16-разрядные компараторы. В тот момент, когда значение счетчика достигает значения, хранящегося в регистре сигнализации, сигнал на выходе компаратора принимает значение "истинно".

Регистр главного режима работы содержит управляющие разряды, позволяющие осуществлять отпирание/блокировку отдельных компараторов. При отпирании компаратора его выходной сигнал появляется на выходе OUT. Полярность выходного сигнала компаратора будет полярностью с высоким уровнем активности, если поле управления выходным сигналом регистра режима работы счетчика содержит код 001 или 010 и будет полярностью с низким уровнем активности, если код 101.

5.3.10. Доступ к регистрам

Конфигурации сигналов управления для всех передач информации по шине данных БИС таймера приведены в табл.6.

Таблица 6

Передача по шине данных

Конфигурация сигналов				Операция с шиной данных
CS	CD	RD	WR	
0	0	0	1	Передача содержимого регистра, адресуемого указателем данных, на шину данных.
0	0	1	0	Передача содержимого шины данных в регистр данных, адресуемый указателем данных.
0	1	0	1	Передача содержимого регистра состояния на шину данных.
0	1	1	0	Передача содержимого шины данных в регистр команд.
x	x	1	1	Нет передачи.
1	x	x	x	Нет передачи.
x	x	0	0	Запрещенное состояние.

5.3.11. Управление главным режимом работы

16-разрядный регистр главного режима работы используется для управления теми внутренними операциями, управление которыми нельзя осуществлять с помощью отдельных регистров режима работы счетчиков. Этот набор операций включает в себя управление частотой, операцию подсчета времени суток, управление компараторами, управление разрядностью шины данных и установлением последовательности операций указателя данных. На рис.15 (приложение 4) показано назначение разрядов регистра главного

3.055.430PЭ

режима работы.

Разряды MM12, MM13, MM14 регистра главного режима работы могут быть по отдельности установлены или сброшены в исходное состояние с помощью команд, заносимых в регистр команд. Также они могут быть изменены посредством записи, осуществляемой непосредственно в регистр главного режима работы. После сброса питания или по команде общего сброса регистра главного режима, этот регистр устанавливается в состояние нуля во всех разрядах. Возникает следующее:

счет времени суток заблокирован;

входной частотой FOUT является F1;

делитель частотного сигнала FOUT установлен на деление на 16;

сигнал FOUT разблокирован;

8-разрядная шина данных;

установление последовательности указателя данных разблокировано;

делитель частоты осуществляет деление в двоичном коде.

Разряды MM0 и MM1 определяют варианты подсчета времени суток /TOD/.

Если разряды MM0=0 и MM1=0, то счетчики 1 и 2 работают как 3, 4 и 5. Если MM0=1 и MM1=1, то в схемах счетчиков 1 и 2 отпирается дополнительная логика дешифрирования и управления, которая вызывает переключение десятичных разрядов этих счетчиков при поступлении импульсов счета, что генерирует соответствующее накопление 24-часового времени суток /TOD/. На рис. 16 (приложение 4) показаны конфигурации счетчиков для операции TOD.

Разряды MM2 и MM3 управляют компараторами, связанными со счетчиками 1 и 2. Каждый из этих 2-х компараторов можно всегда использовать в любом режиме работы. При обращении к функциям контроля времени суток оба компаратора разблокированы. Результат работы компаратора 2 зависит от компаратора 1.

Разряды MM4-MM7 определяют входной сигнал источника для делителя сигнала FOUT (S - SOURCE; G - GATE; F - F). Комбинация из 4-х нулей используется для того, чтобы

3.055.430PЭ

убедиться, что активная частота присутствует на входе делителя FOUT после установки его в исходное состояние.

Разряды MM3-MM11 определяют коэффициент деления для делителя сигнала FOUT. Сигнал источника FOUT (выбранный с помощью разрядов MM4-MM7) делится на целое число от 1 до 16 включительно и затем передается в выходной буфер сигнала FOUT.

Разряд MM12 обеспечивает возможность программного стробирования сигнала FOUT.

Разряд MM13 управляет мультиплексированием интерфейса шины данных с целью конфигурирования схемы для 8-разрядной или 16-разрядной внешней шины.

Разряд MM14 управляет логикой указателя данных для инициализации или блокировки функций автоматического установления последовательности.

Разряд MM15 управляет конфигурацией счета для счетчика делителя частоты. Если MM15=0, то частоты генератора делятся так, что каждая последующая частота составляет 1/10 от предыдущей частоты. Если MM15=1, то выполняется двоично-десятичное деление так, что отношение смежных частот равно 1/10 (см. рис.1/ приложение 4).

3.3.12. Описание режимов работы счетчиков

Разряды SM15-SM13 и SM7-SM5 регистра режима работы счетчиков выбирают режим работы для каждого счетчика. Для упрощения режимы обозначены буквами от A до X и сведены в таблицу (см. таблицу приложения 5).

3.3.13. Варианты управления режимом работы счетчика

В состав каждой логической группы счетчика входит 16-разрядный регистр режима работы счетчика, используемый для управления всеми отдельными вариантами, реализуемыми с помощью связанного с этим регистром универсального счетчика. Эти варианты включают конфигурацию выходного сигнала, управление счетом, управление входным сигналом счета и управление стробированием. Загрузка регистра режима работы счетчика должна выполняться после того как счетчик заблокирован, иначе может быть ошибка в работе счетчика.

После установки в исходное состояние для запуска или при общем сбросе, регистры режима работы счетчика

инициализируются в состояние предустановки. Введенное значение представляет собой 0800 (шестнадцатиричное) и определяет следующую конфигурацию управления:

низкий импеданс входного сигнала относительно земли;

счет в обратном направлении;

счет двоичный;

счет одноразовый;

регистр загрузки в качестве источника перезагрузки;

нет перезапуска;

выбор F1 в качестве источника входного сигнала;

полярность входного сигнала "положительная - истинно";

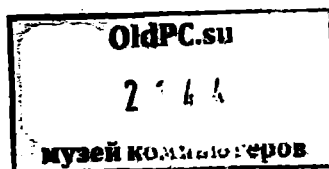
нет стробирования.

Разряды СМ0-СМ2 регистра режима работы счетчика определяют конфигурацию управления выходным сигналом.

Разряды СМ3-СМ7 определяют различные варианты прямого управления процессом счета. СМ3 и СМ4 функционируют независимо от других разрядов и управляют счетом в прямом (обратном) направлении и счетом в двоично-десятичном/двоичном коде. СМ5 управляет повторением счета. Если СМ5=1, то счет будет продолжаться, пока счетчик не будет заблокирован. Если СМ5=0, то процесс счета продолжается только в течение одного полного цикла счета. Затем счетчик автоматически блокируется.

Разряд СМ6 определяет варианты источника для перезагрузки счетчика. СМ6=0 - передается содержимое из регистра загрузки. СМ6=1 - в качестве ячейки перезагрузки выступает либо регистр загрузки, либо регистр хранения.

Разряд СМ7 управляет специальными функциями стробирования, которые позволяют осуществлять перезапуск и выбор регистра загрузки или регистра хранения в качестве источника для перезагрузки счетчика.



3.055.430PЭ

Разряды СМ8-СМ12 определяют источник, используемый в качестве входного сигнала и активный фронт, подлежащий счету. Разряд СМ12 управляет полярностью для всех источников входных сигналов: логический ноль соответствует счету нарастающих фронтов, логическая единица - счету падающих фронтов. Разряды СМ8-СМ11 выбирают один из 16 источников входного сигнала, подзаемного из вход счетчика.

Разряды СМ13-СМ15 регистра режима работы счетчика определяют варианты аппаратного стообирования.

5.3.14. Описание команд

Набор команд таймера позволяет консольному процессору вызывать и контролировать, в соответствии с требованиями, режимы работы и характеристики для конкретных приложений, инициализировать и обновлять как внутренние данные, так и управляющую информацию, а также манипулировать с рабочими разрядами в процессе работы. Команды вводятся непосредственно в 8-разрядный регистр команд посредством записи их в порт сигналов управления. В табл. 4 приведены коды команд и краткое описание каждой функции.

В плате WCS счетчики 4 и 5 объединяются в один 32-битный счетчик времени года. Счетчик 4 обеспечивает младшие 16 разрядов, а 5 - старшие разряды общего счетчика времени года.

На вход масштабатора подаются импульсы с внешнего генератора с частотой 1 кГц. Масштабатор в консоли запрограммирован для двоичного деления, но использован вывод F1, т.е. непосредственно частота генератора (см. таблицу 7).

Таблица 7

Выходы масштабатора

Выход	Двоично-десятичное масштабирование	Двоичное масштабирование
1	F1 - генератор	F1 - генератор.
2	F1/10	F1/16.
3	F1/100	F1/256.
4	F1/1000	F1/4096.
5	F1/10000	F1/65536.

Это означает, что регистр главного режима программируется так, чтобы входом делителя FOUT был вывод F1. Делитель FOUT при этом делит входной сигнал на 10, т.е. выходной сигнал делителя - 100 Hz. Этот сигнал снаружи присоединяется к контакту SOURCE 1 и обеспечивает импульсы для счетчика времени года.

5.3.15. Внутренний таймер центральной части

Внутренний таймер имеет задатчик времени 1 мкс. и может генерировать прерывания центральной части через определенные программой интервалы времени. Он может использоваться для измерения текущего времени с высокой разрешающей способностью. Возможности прерывания при помощи внутреннего таймера используются в основном при каталогизации в операционной системе.

Схемно основная часть внутреннего таймера центральной части содержится в группах логических счетчиков 1 и 2 БИС таймера. Два счетчика объединены в один 32-разрядный счетчик инкрементируемый каждую микросекунду. Внешняя логика таймера (рис. 18 приложения 4) генерирует сигнал TIMER INT, когда счетчик достигает заданного значения. Она генерирует также и импульсы сброса таймера.

В центральной части за таймером закреплены 3 псевдо-регистра:

ICR - регистр счета интервалов;

MICR - регистр счета следующих интервалов;

ICCS - регистр состояния и управления счетом интервалов.

Форматы этих регистров показаны на рис.19 приложения 4.

Обращение к псевдорегистрам осуществляется командами MFPR и MIPR.

3.055.430PЭ

длина интервала счета задается засылкой отрицательного числа "N" в NICK (N - период в микросекундах). Значение NICK обычно определяет длительность счета в следующем за текущим интервалом времени. Засылка в любой момент времени бита XFR в ICCS делает значение хранящееся в NICK счетом текущего интервала. Оба значения счета в ICR и NICK загружаются начально, когда таймер запускается.

Таймер запускается при установке бита RUN в ICCS и инкрементируется каждую микросекунду до тех пор, пока RUN не будет очищен. После очистки бита RUN таймер может двигаться по шагам. При записи бита SGL в ICCS происходит инкрементация счета на 1, если RUN обнулен. Текущее значение счета может быть считано в любой момент времени из регистра ICR.

В конце текущего интервала устанавливается бит прерывания INT в ICCS. Если при этом установлен бит INTEN - разрешение прерывания, то генерируется прерывание центральной части. Бит INT и прерывание центральной части очищаются при сбое INT = N. Можно произвести сброс INT записью дополнительной 1 в позицию INT.

Если бит INT не очищен к концу следующего счета, то взводится бит ошибки. Бит ошибки можно очистить, записав в его позицию единицу.

3.3.16. Связь псевдорегистров центральной части с консолью

Когда загружен NICK и ICCS командами MTPR, микропрограмма центральной части посылает данные регистров пакетом данных в консоль, где они затоминуются в закрепленных за ними ячейках ОЗУ консоли.

ICCS (шесть бит управления) посылаются одним байтом, а NICK - 4 байта дополняются до положительного числа, которое и пересылается в ОЗУ консоли. Если бит XFR в данных регистра ICCS был установлен, консольная программа загружает положительное значение времени счета, записанное в ячейках ОЗУ, зарезервированных для NICK, в два регистра тревоги счетчиков 1 и 2. Регистры хранения этих счетчиков были предварительно обнулены во время инициализации системы.

Если бит RUN устанавливается в ICCS, консольная программа взводит выход регистра команд "START INTRVL TMR", чтобы начать счет.

Счетчики логических групп 1 и 2 программируются как показано в табл. 3.

Таблица 8

Программирование регистров таймера

Регистр/функция	Выбранный режим	
	Счетчик 1	Счетчик 2
1. Регистр режима счетчика		
управление стробированием	низкий уровень, вход GATE1.	никакого.
выбор источника	источник 2, активный высокий.	активный высокий, TC(N-1).
управление выходом *	активный высокий, TC-импульс.	активный высокий, TC-импульс.
управление счетом	разрешен специальный строб-сигнал, перезагрузка из регистра загрузки, счет повторяется, двоячно, вверх.	разрешен специальный строб-сигнал, перезагрузка из регистра загрузки, счет повторяется, двоячно, вверх.
2. Регистр главного режима		
разрешение сравнения	разрешено	разрешено.
режим TOD	запрещается	запрещается.

*

Поле "сравнение разрешено" в регистре главного режима подавляет поле "управление выходом" в регистре режима счетчика, чтобы взводить выход логической группы всякий раз, когда значение счетчика станет равным регистру тревоги. Полярность выхода при этом продолжает зависеть от поля "управление выходом".

Сигнал START INTRVL TMR запускает счет, задавая низкий уровень на входах GATE 1 и 2.

Счетчик 1 (младшие 16 бит полного счетчика) начинает инкрементироваться на каждый поступающий импульс. Входные импульсы счетчика, поступающие на вход SOURCE 2 микросхемы, обеспечиваются внешним делителем на 5 тактового сигнала 5 мHz микропроцессора.

Счетчик 2 (старшие 16 бит полного счетчика) инкрементируется окончательным сигналом со счетчика 1. Когда счетчики достигают значения регистров тревоги, взводятся

цепи компараторов двух логических групп и во внешней логике таймера установится сигнал TIMER INT. Появится также импульс сброса, который установит высокий уровень на входе GATE-1 микросхемы. Счет остановится и произойдет пересылка нулей из регистра загрузки в счетчик 1. Счетчик 1 снова начнет инкрементироваться пока нет сигнала сброса. Временная диаграмма работы показана на рис. 20 приложения 4. Когда программа консоли обнаруживает, что установлен TIMER INT, она инициализирует счетчик 2, загружая в него нули из регистра загрузки (этот счетчик не очищается импульсом сброса), и генерирует запрос на прерывание микропроцессора, устанавливая INTRVL TMR INT на выходе регистра команд, если бит INT EN установлен в ICCS. Программа консоли также перезагружает регистр тревоги, если данные NISR изменялись со времени последней загрузки, и устанавливает бит INT в ICCS, если только он уже не установлен, в этом случае она устанавливает бит ERROR.

После того как INT (или ERROR) флаг установлен, программа консоли очищает триггер TIMER INT в логике внешнего управления, путем обнуления последовательного выхода микропроцессора. Таймер будет продолжать инкрементироваться, устанавливая TIMER INT в конце каждого заданного значения счета, до тех пор пока бит RUN в ICCS не очистят. Всякий раз, когда RUN очистят, установка бита SGL в ICCS будет произведена программой консоли, чтобы произвести загрузку кода команды в таймер, одиночным импульсом в счетчик 1, увеличивая значение счета интервалов на 1.

ПРИМЕЧАНИЕ. Во время обычных операций, значение счета в NISR уменьшается на три прежде, чем оно загружается в регистры тревоги. Это делается, чтобы скомпенсировать собственную задержку внешней логики при очистке счетчика младших 16 бит (не уменьшенное значение NISR грузится при одношаговом смещении счетчика). Таким образом задержка матобеспечения при очистке счетчика, обеспечивающего старшие 16 бит, препятствует соответствующей операции для очень малых временных интервалов.

Прерывания микропроцессора, вызванные таймером, генерируются не только во время перебега с INT EN, предварительно установленным в ICCR, но также во время счета, если бит INT устанавливается в ICCS, и ICCS перезагружается INT EN=1. Сигнал прерывания микропроцессора очищается консольной программой как только ICCS загружается INT=1 или INT EN=0.

Когда ICCS или ICR считываются командой MFPR, данные регистра передаются пакетом данных из программы консоли в микрокоманду центральной части. Если ICCS отсылается, микрокоманда должна распаковать один байт данных регистра (содержащий шесть битов управляющих ICCS) и запомнить в консоли в соответствующем 32-битном формате для передачи при помощи команды MFPR. Если пересылается ICR, программа консоли получает текущее значение таймера путем загрузки счетчиков 1 и 2 в соответствующие им ре-

гистры хранения.

Данные регистров хранения, положительные значения, далее считываются с микросхемы таймера и посылаются в микропрограмму центральной части (четыре байта), где они должны быть преобразованы в отрицательное значение счета, поставляемое программой консоли. Это величина, отрицательное значение счета, сохраняется микропрограммой все время пока N1CR записывается при помощи MTPR и до того, как оно преобразуется в положительное значение, которое запоминается при помощи программы консоли.

5.3.17. Часы времени года (группы логических счетчиков 4 и 5)

Часы времени года, которые образуются из логических групп счетчиков 4 и 5 в таймере, есть 32-битный двоичный счетчик, который инкрементируется каждые 10 мс. Он не имеет возможности прерываться, действуя только как индикатор времени большой длительности, для операционной системы.

Регистры режима счетчиков для групп логических счетчиков 4 и 5 программируются, как показано в таблице 9.

Основным источником импульсов времени года служит синхросерия 1 мHz поступающая на вход SRC 2 таймера. При питании от резервного источника, источником счета, служит внешний генератор с частотой 1 kHz, выход которого подается на выходной контакт FOUT таймера. Сигнал FOUT таймера поступает на 10 Hz задерживающую защелку. Эта защелка позволяет микропроцессору задерживать фронт импульса в тот момент, когда импульс времени года доступен (читается или пишется). Выход 100 Hz задерживающей защелки присоединен к контакту SOURCES и задает импульсы на счетчик 4. Этот счетчик обеспечивает 16 младших бит в значении времени года. Счетчик 5, образующий 16 старших бит, в этом значении, получает импульсы конца счета счетчика 4.

Таблица 9

Выбор режима счетчика для групп логических счетчиков 4,5

Функция	Выбранный режим	
	Счетчик 4	Счетчик 5
Управление синхронизацией	Никакой	Никакой.
Выбор источника	Высокий уровень, рабочий, источник 1.	Высокий уровень, рабочий ТC(N-1).

Продолжение табл. 9

Функция	Выбранный режим	
	Счетчик 4	Счетчик 5
Управление выходом	Неактивно.	Неактивно.
Управление счетом	Специальные данные отключения, перезагрузка из регистра загрузки, счет повторный, двоичный, сверх.	Специальные данные отключения, перезагрузка из регистра загрузки, счет повторный, двоичный, сверх.

К счетчику времени года можно обратиться через программу центральной части в любое время, обращаясь к регистру времени года (рис. 21 приложение 4) при помощи MTPR или MFPR. Данные регистра проходят между микропрограммой центральной части и консольной программой в виде пакета данных.

Когда регистр времени года пишется командой MTPR, программа консоли берет беззнаковое 32-битное двоичное значение счета, посылает через микропрограмму центральной части (4 байта) и записывает его в регистры загрузки для счетчиков 4 и 5. Она потом загружает счетчики из регистров загрузки.

Когда регистр времени года читается командой MFPR, консольная программа загружает текущее значение счета счетчиков 4 и 5 в соответствующие им регистры хранения, считывает регистры хранения и потом посылает значение счета в микропрограмму центральной части (4 байта).

На уровне команд нет управления включением/выключением часов времени года. Регистры режима счетчиков программируются консольной программой однажды, после этого счетчики будут непрерывно инкрементироваться пока подано напряжение на микросхему таймера, микросхему генератора 1 кГц и микросхему, на выходе генератора. При исчезновении питания в системе, программа консоли проверяет регистры режима для счетчиков 4 и 5, чтобы проверить, что они установлены правильно. Если обнаружена ошибка, значение времени года (даже если она вычисляется во время отсутствия общего электропитания благодаря опорному аварийному электропитанию) предполагается неточным и программа консоли останавливает его подсчет, очищает регистры и устанавливает в регистры режима правильное значение. Потом она перезапускает вычисление с нуля.

3.055.430PЭ

3.3.18. Таймер аварии питания (группа 3 логических счетчиков).

Группа 3 логических счетчиков резервируется для использования в программе консоли, как 2 мс таймер. Этот таймер представляет собой двоичный инкрементируемый счетчик, который использует тот же самый источник импульсов, что и таймер центральной части, заставляющий его инкрементироваться с периодом 1 мкс. Регистр режима счетчика программируется как показано в табл. 10.

Таблица 10

Выбор режима счетчика для группы 3 логического счетчика.

Функция	Выбранный режим - счетчик 3
Управление стробированием	Никакого.
Выбор источника	Реакция на высокий уровень, источник 2.
Управление выходом	Реакция на низкий уровень, импульс ТС.
Управление счетом	Отключает специальную дату, перезагружает с регистра загрузки, повторный счет, двоичный, вверх.

Таймер аварии питания используется, когда обнаруживается отказ питания в комплексе (UNIBUS AC LOW устанавливает SEE AC LO, который прерывает микропроцессор по его входу RST 7.5). Когда происходит отказ электропитания, программа консоли проверяет флаг в ОЗУ консоли. Флаг указывает, что микропроцессор был в режиме программы или в режиме ввода/вывода консоли, когда AC LO был установлен. Если микропроцессор был в режиме ввода/вывода, он переходит к своей собственной подпрограмме обработки аварии питания (описана ниже). Однако, если микропроцессор в программном режиме, то он устанавливает прерывание процессора по ошибке из-за аварии питания (уровень 1), пускает таймер аварии питания (2 миллисекунды) и уходит назад в нормальный цикл режима программы.

Центральная часть обрабатывает прерывание по аварии питания очень быстро (благодаря высокому уровню приори-

3.055.430PЭ

тета этого прерывания EI) и уйдет на подпрограмму обработки аварии питания в операционной системе. Подпрограмма аварии питания прекратит работу, сохранит всю необходимую информации и произведет прерывание собственной команды (навсегда).

Тем временем, консольный процессор работает в холостом цикле программы. Микропроцессор прозверяет в этом цикле таймер аварии питания и определяет окончание двухмиллисекундного периода. Микропроцессор тогда остановит подачу импульса процессору, делая невозможным обращение к главной памяти и уйдет в свою подпрограмму включения питания. Микропроцессор будет оставаться в этой подпрограмме до тех пор, пока электропитание полностью не отключится (первичное питание переменного тока потеряно или ключ в положении выключено) или до тех пор, пока электропитание переменного тока не вернется (ключ STD BY поставлен в положение включено). Если первичное электропитание вернулось, микропроцессор попытается начать рестарт, если ключ AUTO-RESTART/BOOT в положении "включено".

3.4. Операции чтения и записи

Работа микропроцессора, управляемая консольной программой, в основном состоит из последовательностей операций ввода/вывода из/на различные устройства консольного процессора (память и устройства ввода/вывода). Память состоит из ОЗУ и ПЗУ. Все остальные устройства, адресуемые микропроцессором - это устройства ввода/вывода (например, таймер и УС АПП-ы).

Основные типы операций чтения/записи памяти и устройств ввода/вывода, а также соответствующие им адреса, указаны в таблице 11. Адреса памяти - 16-битные, всего может адресоваться 64К-байта, однако в консольном процессоре используются лишь адреса 0000-FFFF для адресации 4К ПЗУ и 4000-7FFF для адресации 16К ОЗУ. Кроме того, адреса 1000-17FF зарезервированы под дополнительные 2К ПЗУ. Адреса ввода/вывода 8-битные, причем они выставляются на младшие 8 бит шины адреса и дублируются на старшие 8 бит, т.е. при адресации ввода/вывода сигнал A15 совпадает с A7, A14 с A6, ..., A8 с A0. Минимальный адрес ввода/вывода (в принятой терминологии - номер порта) - 00, максимальный - FF. В консольном процессоре максимальный используемый адрес ввода/вывода - 5C.

Таблица 11

Операции чтения и записи

Тип передачи (память или ввод/вывод)	Занимаемые адреса	Операция
Память	0000-0FFF	Чтение ПЗУ, основные 4К.
Память	1000-17FF	Чтение ПЗУ, дополнительные 2К.
Память	4000-7FFF	Чтение/запись 16К ОЗУ.
Ввод/вывод	00-07	Чтение состояния ключей и уровня питания.
Ввод/вывод	08-0A	Запись в регистр записи управляющей памяти.
Ввод/вывод	1C, 1D	Чтение/запись регистров таймера.
Ввод/вывод	20	Чтение регистра сводок готовности.
Ввод/вывод	20-2F	Запись в 0-й регистр команд консоли.
Ввод/вывод	30-3F	Запись в 1-й регистр команд консоли.
Ввод/вывод	4D-4B	Чтение/запись регистров УСАПП-ов.
Ввод/вывод	80-87	Чтение CRR (регистра чтения консоли) и состояния процессора.
Ввод/вывод	A0-AF	Запись во 2-й регистр команд консоли.
Ввод/вывод	CC	Запись в CWR (регистр записи консоли).
Ввод/вывод	EC	Запись по этому адресу вызывает загрузку CRR шины Y.

5.4.2. Операции чтения ПЗУ

ПЗУ консоли состоит из 2К х 4-битных микросхем. Микросхемы собираются в две (для 4К ПЗУ) или в три (6К ПЗУ) группы по 2 микросхемы в каждой. Группа имеет емкость 2Кх8 бит. При обращении микропроцессора к памяти два старших бита адреса ПЗУ (линии A12 и A11) выбирают группу, остальные адресные биты на линиях A10:A08 и шина AD (A07:A00) выбирают ячейку внутри группы (см. табл. 13).

Таблица 13

Адресация ПЗУ

Линии	Шина AD	Функция адресации
A12:A11		Выбор группы.
A10:A08	A07:A00	Выбор одной из 2к возможных ячеек внутри группы.

На блок-схеме консольного процессора показано, что линии шины AD адресуют ПЗУ напрямую. Однако, так как на шине AD мультиплексируются адрес и данные, то восемь младших бит адреса, устанавливаемые к началу рабочего цикла, запоминаются в буферном регистре. Буферизуемые линии шины AD, CONS DATA (7:0), присоединены к регистру. Выходы буферного регистра, присоединенные к микросхемам ПЗУ устанавливают во время машинного цикла младшие 8 бит адреса.

Временная диаграмма для операции чтения ПЗУ показана на рис.23 приложения 4.

Для фиксации младшей части адреса в буферном регистре используется сигнал ALE (LATCH ADRS - защелка адреса).

После того, как микропроцессор уберет с шины биты адреса, взводится сигнал SEL ROM MUX (сигнал выборки "чтение"), который стробирует данные из ПЗУ через мультиплексор на шину AD, откуда их считывает микропроцессор.

5.4.5. Работа ОЗУ

ОЗУ консольного процессора имеет емкость 15К байт и состоит из восьми параллельно соединенных МОП-микросхем 16Кх1. Микросхемы представляют собой динамическую память и должны периодически регенерироваться.

Каждая микросхема (16Кх1: 128 строк x 128 столбцов) имеет 7 линий адреса. На эти линии адреса во время операций чтения/записи ОЗУ подается 7-битный адрес строки, а затем - 7-битный адрес столбца. Адреса строк и столбцов вырабатываются из адреса памяти, выделенного микропроцессором на линии А и шину АД (защелкнутые на шине АД биты адреса памяти используются так же, как при адресации ПЗУ). Строку в микросхеме ОЗУ выбирают биты адреса, имеющие нечетные номера, столбец - четные (см. табл.14).

Таблица 14

Адресация ОЗУ

Номера битов на линии А	Номера битов на шине АД	Функция адресации
15, 11, 09	7, 5, 3, 1	Выбирают одну из 128 возможных строк.
12, 10, 08	6, 4, 2, 0	Выбирают один из 128 возможных столбцов.

Временная диаграмма операций чтения/записи ОЗУ показана на рис.24 приложения 4. При обращении к микросхемам динамического ОЗУ, по сравнению с обращением к ПЗУ, требуется дополнительная фаза ожидания (t_W). Наличие этой фазы организуется сигналом LONG CYCLE, который устанавливает вход RDY микропроцессора в 0. Логика управления ОЗУ вырабатывает LONG CYCLE из двух сигналов: START 8085 CYCLE и единицы на линии A14. При обращении к ОЗУ A14 всегда 1, а START 8085 CYCLE устанавливается сигналом LATCH ADRS, который взводится к началу каждого машинного цикла.

START 8085 CYCLE также генерирует первый из двух адресных стробов, по которым адреса строк и столбцов загружаются в микросхемы динамического ОЗУ. Первый строб, 8085 ROW STB - это установка триггера непосредственно сигналом START 8085 CYCLE. Второй строб, 8085 COL STB, обеспечивает удержание триггера в выключенном состоянии, пока адрес строки стробируется через мультиплексор ОЗУ. Адрес строки выбирается входами мультиплексора при нормально взведенном J-K триггере. После того, как

3.055.430PЭ

считан адрес строки (8085 ROW STB принят микросхемами ОЗУ), триггер меняет состояние, чтобы допустить установку 8085 COL STB и стробировать адрес столбца. При чтении из ОЗУ, после выдачи строба адреса столбца, данные из ячейки, указанной строкой и столбцом, передаются микросхемами ОЗУ на шину AD, откуда читаются микропроцессором. При записи в ОЗУ строб столбца вызывает загрузку данных с шины AD в выбранную ячейку ОЗУ (на шину AD микропроцессор выставляет данные для записи). Во время машинного цикла микропроцессор устанавливает сигналы на линиях состояния (S0, S1), из которых вырабатывается сигнал 8085 WRT CYC. Этот сигнал связан непосредственно со входом "разрешение записи" микросхем ОЗУ и задает операцию "запись в ОЗУ".

Чтобы предотвратить потерю данных, каждый запоминающий элемент в микросхемах динамического ОЗУ консольного процессора должен регенерироваться хотя бы один раз за интервал времени, называемый интервалом регенерации. Для 16x1 микросхем ОЗУ консольного процессора интервал регенерации - 2 мс или менее. Одиночный строб адреса строки регенерирует все элементы строки. Таким образом, элементы памяти для всех 8 микросхем ОЗУ могут быть регенерированы 128 стробами по всем возможным адресам строк. Один такой строб будет регенерировать по одной строке в каждой микросхеме. Одиночный строб для регенерации выдается каждые 12,8 мкс, что составляет интервал регенерации приблизительно 1,7 мс.

Временная диаграмма регенерации ОЗУ приводится на рис. 25 приложения 4.

Цикл регенерации (одиночный строб строки) инициализируется сигналом REQ 8085 REF. Этот сигнал задается одним из выходов самостоятельно работающего счетчика (D72), используемого также для генерации бод-импульсов для УСАПП-рв (см. ниже). REQ 8085 REF имеет период 12,8 мкс, и его положительный фронт устанавливает в управляющей логике ОЗУ сигнал 8085 REFR CYC. Этот сигнал отключает мультиплексор адреса ОЗУ и разрешает работу выходов двоичного циклического счетчика (счетчика регенерации), который и выставляет номер строки на линии адреса ОЗУ. Затем 8085 REFR CYC генерирует строб адреса строки (т.е. сигнал 8085 ROW STB), восстанавливая таким образом строку, адресованную текущим значением счетчика регенерации. Задний фронт 8085 REFR CYC инкрементирует счетчик регенерации, и на следующем цикле регенерации (через 12,8 мкс) будет восстанавливаться следующая строка. За интервал регенерации (12,8 мкс x 128 строк = 1,7 мс) счетчик проходит по всем возможным адресам строк.

Если запрос на регенерацию появляется во время выполнения операций чтения/записи ОЗУ, то установка сигнала 8085 REFR CYC и, как следствие, цикл регенерации задерживается до завершения чтения/записи.

Примечание: цикл регенерации памяти задерживается также, если выполняется операция чтения/записи регистров

УСАПП-ов (см. ниже).

Если запрос чтения/записи ОЗУ возникает (или должен возникнуть) во время регенерации, то взводится LONG CYCLE на время, достаточное для дополнительного ожидания в машинном цикле, и регенерация выполняется при установленном START 3085 CYCLE. Непосредственно чтение/запись производится сразу после цикла регенерации. Временная диаграмма расширенного машинного цикла представлена в нижней части рис. 25 приложения 4.

Для того, чтобы 3085 REFR CYC мог взвестись и затем сняться сразу же после (и только после) положительного фронта сигнала REQ 3085 REFR (период 12,8 мкс), используется синхронизирующий триггер. Триггер, который устанавливается сигналом 3085 REFR CYC и сбрасывается, когда REQ 3085 REFR убирается. Его сигнал называется CON STORE REFR и используется также для инициализации регенерации основной управляющей памяти. Цикл регенерации управляющей памяти инициируется отрицательным фронтом CON STORE REFR с тем же периодом, что и цикл регенерации ОЗУ контрольного процессора (12,8 мкс).

3.4.4. Операции ввода/вывода

Операции ввода/вывода консоли реализуются командами микропроцессора "IN" и "OUT" и имеют диапазон адресов ввода/вывода (т.е. номера портов) 00:0A, 1C:4B, 30:AF, CC и 2C (в шестнадцатиричной системе). Адреса 0B:19, 1E:1F, 4C:7F, 38:9F, 80:CB, CD:EB, E0:FF в консольном процессоре не используются.

Управляющие регистры и регистры данных УСАПП-ов и таймера (адреса 1C и 1D) обсуждались выше. По другим адресам доступны регистр чтения консоли CRR (адрес 80), регистр записи консоли CWR (адрес CC), регистр записи управляющей памяти (08, 09, 0A), регистр сволок битов готовности (20). Кроме того при записи в адреса 20:3F и A0:AF устанавливаются (или убираются) отдельные биты регистров команд консольного процессора. Например, запись любого байта по адресу AA устанавливает во втором регистре команд консоли бит "CONS ATTN". Биты состояния (состояния микропереключателей и пультовых переключателей, наличие сигналов от процессора, старшие биты UPC и CSR) могут быть получены чтением из адресов 00:07 и 82-87.

Временная диаграмма для операции чтения по различным адресам ввода/вывода приведена на рис. 26 приложения 4.

К началу машинного цикла адрес ввода/вывода передается микропроцессором на линии A, а также дуолируется на мультиплексируемую шину AD. Линии A связаны с большинст-

вом устройства ввода/вывода. Однако устройства процессора доступны лишь через шину консоли, которая является расширением шины AD. Адрес, установленный на шине AD к началу машинного цикла, запоминается в схеме защелки по сигналу LATCH ADRS так же, как при получении младшей части адреса памяти. Выход схемы защелки может быть использован для адресации устройств ввода/вывода на процессоре при полном цикле ввода/вывода.

После того, как микропроцессор установит адрес ввода/вывода и LATCH ADRS, а также сигнал 8085 I/O, управляющая логика чтения/записи консольного процессора заставит адресуемое устройство или передать данные на шину AD, или загрузить оттуда записываемые данные. Читательные данные будут затем считаны микропроцессором с шины AD.

5.4.3. Чтение и запись регистров УСАПП-ов

Временная диаграмма для чтения/записи регистров УСАПП-ов - приведена на рис. 26 приложения 4. При адресации регистров УСАПП-ов на линии A14 (и A06) всегда 1, и поэтому сигнал START 8085 CYCLE устанавливает LONG CYCLE, что обеспечивает дополнительную фазу ожидания во время машинного цикла.

START 8085 CYCLE также разрешает работу дешифратора, выход которого взводит один из трех триггеров SEL U1, SEL U2 или SEL U3 (см. рис. 22 лист 3 приложения 4). Какой именно из этих триггеров взводится, зависит от выбранного УСАПП-а, определяемого по состоянию адресных линий A11 и A10. Выходы триггеров присоединены непосредственно к входам "выбор микросхемы" УСАПП-ов.

Линии A09 и A08 также присоединены непосредственно к УСАПП-ам. Эти адресные биты определяют внутренний регистр УСАПП-а. При взведенной линии "разрешение микросхемы" заданный регистр в указанном УСАПП-е либо записывается (данные - с шины AD), либо читается (данные на шину AD). Тип операции, также как при обращении к ОЗУ, определяет сигнал 8085 WRT CYC, поступающий на входы "разрешение записи" УСАПП-ов.

Поскольку часть логики управления чтением/записью используется для управления как УСАПП-ами, так и ОЗУ (например, сигнал LONG CYCLE), запрос регенерации ОЗУ может увеличивать время обращения к УСАПП-у. Также как для расширенного цикла ОЗУ, при поступлении запроса регенерации в момент начала операции чтения/записи УСАПП-ов, машинный цикл увеличивается на состояние ожидания. Если же запрос на регенерацию пришел во время операции чтения/записи УСАПП-а, то регенерация будет задержана до завершения операции.

5.055.430PЭ

5.4.6. Чтение и запись других устройств

Сигналы выбора объектов чтения/записи устанавливаются во время машинного цикла при выдаче сигналов RD 8085A и WR 8085A. Момент, в который данные должны быть переданы устройством на шину AD, определяет передний фронт RD. Данные считываются с шины AD в устройство по заднему фронту WR.

Исключение составляет SEL TIMER - сигнал "выбор микросхемы таймера". Он взводится сигналом IO 8085 к началу машинного цикла, а RD и WR поступают непосредственно на микросхему таймера, что обеспечивает правильные временные соотношения для передачи данных при вводе/выводе.

Для операции чтения устройств ввода/вывода вырабатываются два сигнала выбора объектов - SEL STATUS и SEL ROM MUX (SEL ROM MUX взводится также для чтения ПЗУ). SEL STATUS разрешает доступ к ключам панели ЭЗМ и микропереключателям (разрешение самотестирования, уровень питания и другие). Состояние ключей и переключателей, называемое "биты состояния", считывается через пару мультиплексоров. Пара битов состояния считывается на линии AD0 и AD7 (каждый бит через свой мультиплексор). Линии A10, A09 и A08 указывают, какой именно бит состояния надо прочесть.

Во время чтения регистра сводок бит готовности устанавливается SEL ROM MUX. Это заставляет выходы "готовность" УСАПП-ов и "перерывание" и "тайм-аут питания" таймера стробироваться на шину AD через мультиплексор ПЗУ. Линия A13, которая при чтении регистра сводок бит готовности разна 1, запрещает выходы ПЗУ и разрешает биты состояния на входах мультиплексора ПЗУ.

При операции записи в регистр записи управляющей памяти взводится WRITE WWD REG. Каждый байт 24-битного регистра имеет свой адрес ввода/вывода, и соответственно, за один раз записывается один байт. Каждый байт записывается по своему синхросигналу: WRITE WWD REG разрешает работу дешифратора, один из выходов которого заставляет загружаться адресуемый байт. Взводимый выход дешифратора выбирается линиями A09 и A08. Регистр записи управляющей памяти загружается с шины AD по заднему фронту выхода дешифратора.

Сигналы выборки WRITE M0 и WRITE M1 заставляют одиночный бит записываться в 0-ой или 1-й регистр команд консоли.

Когда регистр команд, представляющий собой 8-битную схему заделки, указывается сигналом выборки, устанавливается или снимается сигнал на одном из выходов регистра. Какой именно сигнал меняется, а также снимается

3.055.430P3

он или устанавливается, зависит только от адреса ввода/вывода. Другие выходы регистра команд не меняются. Когда сигнал выборки снимается, все выходы фиксируются в текущем состоянии. Регистры команд не зависят от данных, выставляемых на шину AD. Три из четырех младших бит адреса ввода/вывода, на линиях, а выбирает номер бита регистра команд, а который загружается четвертый (младший адресный бит). Младший адресный бит используется в качестве данных, которые должны быть помещены в выбранный бит регистра команд консоли. Например, адрес ввода/вывода 20 (запись в 20 порт, 20 - 16-тиричное) сбрасывает 0-й бит 0-го регистра команд, т.е. запись в 20 порт - это сигнал CLEAR CPU RUN. Адрес 21 - SET CPU RUN.

Выбор устройств, адресуемых вводом/выводом консоли, но находящихся в центральной части, реализуется по выходам защелки адреса и дешифратора адреса. Дешифратор управляется вырабатываемым на плате консольного процессора сигналом SEL SPU REG1. Другой сигнал выборки (SEL CPU REG2), вырабатываемый в консольном процессоре, управляет приемо/передатчиками, которые стробируют данные с шины AD через консольную шину CONS BUS из процессора.

Во время записи SEL CPU REG 2 убирается, что составляет приемо передатчик передавать данные с шины AD в процессор. Для изменения направления передачи (чтение из устройств ввода/вывода, расположенных в центральной части) этот сигнал взводится.

Второй регистр команд консоли (он расположен в центральной части) загружается практически также, как и 0-й и 1-й регистры. Когда адресуется 2-й регистр, дешифратор адреса выставляет "выбор регистра", а четыре младших бита адреса выбирают и устанавливают/сбрасывают нужный бит регистра команд.

При адресации регистра записи консоли (CWR) он стробируется выходом дешифратора. Запись в регистр данных с шины консоли происходит по заднему фронту выхода дешифратора. Другой выход дешифратора определяет чтение из регистра чтения (CRR) на шину консоли. Младший бит регистра вводится на линию 0 шины консоли через мультиплексор входных данных. Три младших бита адреса ввода/вывода (для чтения CRR они равны 0), выбирают соответствующий вход мультиплексора. Другие значения младших битов (не нули) читают через тот же мультиплексор одиночные биты состояния процессора. Именно таким образом читаются, например, биты состояния "внимание от процессора" (CPU ATTN) и "готовность процессора" (CPU ACK).

Регистр чтения консоли (CRR) обычно загружается с шин "Y" микрокодом центральной части. Однако он может быть загружен с помощью консольной операции ввода/вывода (запись по адресу 5C). Адрес ввода/вывода просто устанавливает выход дешифратора адреса. По заднему фронту этого выхода данные с шины Y (но не данные ввода/вывода с шины консоли) и загружаются в CRR. Эта возможность ис-

5.055.43UP3

пользуется в микродиагностике, базирующейся на консольном процессоре, для проверки основных операций обмена консольного процессора с процессором. Тестовые данные загружаются в CWR, оттуда читаются на "D"-шину, и, через процессор записываются на шину "Y". Затем CWR выбирается, данные считываются в него с "Y"-шины, читаются и проверяются. Перед тестированием компонент процессора, через CWR и CRR эти регистры проверяются описанным методом, при этом используются различные типы тестовых данных.

ПРИМЕЧАНИЕ.

Для того, чтобы разрешить передачу из CWR на шину "D", затем через процессор на шину "Y", микродиагностика записывает в регистр управляющей памяти микрокоманду MOVE (НОМ D ADRS=FC). Регистр управляющей памяти загружается поочередно.

5.5. Связь между консольным процессором и процессором

Во время работы ЭВМ связь между программой, проходящей в консольном процессоре, и микропрограммой центральной части осуществляется при помощи передачи данных по шине консоли. Передача производится, в зависимости от ее направления, либо через регистр записи консоли (CWR), либо через регистр чтения консоли (CRR).

Консольная программа посылает байт в процессор, записывая его в порт номер 00 микропроцессора (команда OUT 00), что вызывает загрузку этого байта в CWR. После загрузки в CWR байт может быть считан в процессор микрокомандой MOVE (адрес регистра FC).

Микропрограмма центральной части посылает байт из процессора в консольный процессор, записывая его микрокомандой MISC (значение второго поля функции = 7) в CRR. После этого байт может быть считан в консольный процессор командой IN 80 (чтение из порта номер 80).

Информация передается по шине консоли группами байт (пакетами данных). Число байт в пакете и число пакетов зависят от режима работы машины (консольный или программный) и от того, какая функция выполняется.

5.5.1. Общение в консольном режиме

При работе в консольном режиме WCS является ведущим и центральная часть выполняет его запросы. При выдаче

оператором консольных команд, требующих действия микропрограмм центральной части, консольная программа организует связь с процессором через CWR и CRR (по шине консоли). Например, команду чтения памяти может обслужить только микропрограмма центральной части, которая должна получить из консоли адрес памяти. Микропрограмма центральной части, получив этот адрес, осуществляет обращение к памяти и возвращает информацию по шине консоли. Затем эта информация распечатывается консольной программой.

Таким образом, в результате консольных команд иницируется консольная программа, осуществляющая связь с процессором, причем пакеты данных, используемые для обмена информацией, имеют размер 10 байт. Исключение составляют команды передачи памяти: загрузка в память и команда "X", которые могут использовать пакеты переменной длины.

Направление передачи и форматы пакетов показаны на рис. 27 приложения 4. Консольная программа посылает 10-байтовый пакет, инициализирующий операцию центральной части. Пакет содержит: 1 байт кода операций, 1 байт - модификатор, 4 байта - резервируемые для адреса и 4 байта, резервируемые для данных. Даже если для операции не требуется модификатор, адрес или данные, все равно посылаются 10-байтовым пакетом.

Микропрограмма центральной части, получив пакет из консольной программы, пытается выполнить указанную в этом пакете операцию. Затем микропрограмма посылает в консольный процессор ответный 10-байтовый пакет, в котором и передает необходимую информацию. Ответный пакет имеет такой же, как у посланного консолю пакета, формат, но вместо кода операции там указывается код правильности выполнения.

Для всех консольных команд, кроме команд передачи памяти "L" и "X", обмен происходит только 10-байтовыми пакетами (один пакет иницирует операцию, а второй извещает о ее завершении). Однако если первый пакет определяет операцию передачи содержимого памяти, то информация из памяти передается в пакете переменной длины. Четыре байта, зарезервированные в командном пакете под данные, в данном случае указывают длину передачи из памяти в байтах. Непосредственно данные следуют за пакетом (командным или ответным). Направление передачи определяется знаком длины передачи в командном пакете. Если длина передачи положительна, реализуется передача в память (данные после командного пакета). Отрицательная длина соответствует передаче из памяти (данные после завершающего пакета).

Обмен по шине консоли также иницируют консольные команды "индикация", "занесение" и "продолжение операции", однако эти команды не передают дополнительных данных после 10-байтовых пакетов. Операции индикации и занесения работают с физической или виртуальной памятью, общими регистрами и специальными процессорными регист-

рами (т.е. PSL или CSR U, 1 и 2 в MCT). младшая часть модификатора в командном пакете указывает тип доступа (память, регистр и т.д.), а старшая определяет размер передачи (байт, слово или длинное слово).

Команда "инициализация", которая инициализирует различные регистры процессора и генерирует маски и константы в местной памяти, не требует ни модификатора, ни адреса, ни данных в командном пакете. Команда "продолжение" (продолжение выполнения программы процессора) также не требует информации по адресу и/или данным, но модификатор необходим - он указывает, продолжить ли выполнение обычным образом или по отдельным инструкциям (режим "пошаговое выполнение" устанавливается обычно другой консольной командой).

Состав пакета для передачи содержимого памяти переменной длины рассмотрен выше. Значение модификаторов и других параметров пакетов данных при различных обменах в консольном режиме рассмотрены в "листинге микрокода процессора".

Передача каждого байта в пакетах синхронизируется и управляется сигналами "внимание" и "подтверждение", вырабатываемыми и в консольном процессоре, и в DAP. Всего используется 4 сигнала: CONS ATTN - "внимание к консоли", CPU ATTN - "внимание к центральной части", CONS ACK - "подтверждение от консоли" и CPU ACK - "подтверждение от центральной части". Для синхронизации передачи в процессор используются CONS ATTN и CPU ACK, а для синхронизации передачи в консольный процессор - CPU ATTN и CONS ACK.

При передаче информации в процессор консольная программа изводит сигнал CONS ATTN, чтобы сообщить микрокодам центральной части, что первый байт информации уже в CWR. После этого микропрограмма центральной части читает этот байт и изводит сигнал CPU ACK, чтобы сообщить консольной программе о возможности загрузить в CWR следующий байт. Когда загружен следующий байт, консольная программа снимает сигнал CONS ATTN. Микропрограмма центральной части после чтения этого байта соответственно снимает CPU ACK. Такое чередование сигналов "внимание к консоли" и "подтверждение от центральной части" продолжается, пока не передадутся все байты пакета.

Передача из процессора в консольный процессор управляется и синхронизируется таким же способом, только вместо CPU ACK используется CPU ATTN (он выставляется или снимается микропрограммой центральной части после загрузки байта в CRR), а вместо CONS ATTN - CONS ACK (выставляется и снимается консольной программой после чтения очередного байта из CRR).

Сигнал CONS ATTN выставляется консольным процессором записью в порт номера AA, а снимается - записью в порт AB. Наличие сигнала CPU ACK проверяется чтением из

82-го порта, а CPU ATTN - чтением из 83-го порта консольного процессора.

Правильность операции во время обмена по шине консоли контролируется консольной программой. Если микропрограмма процессора не ответит на изменение сигнала CONS ATTN (или CONS ACK) изменением своего синхронизирующего сигнала CPU ACK (или, соответственно, CPU ATTN) за определенное время, произойдет тайм-аут в консольной программе, который сигнализирует об ошибке связи, напечатав сообщение об ошибке. Затем консольная программа перейдет в холостой цикл.

5.5.2. Общение в программном режиме

При работе в программном режиме центральная часть является ведущей, и консольный процессор выполняет ее запросы. Связи по шине консоли инициируются микропрограммами центральной части, в основном обслуживаемыми команды MTPR и MFPR, направленные к регистрам устройств, расположенных в консольном процессоре. Такими устройствами являются: консольный терминал (местный или удаленный), кассетный магнитофон, интервальный таймер и часы времени года.

Типы передач показаны на рис. 28 приложения 4.

В программном режиме пакет данных имеет переменный размер (от 1 до 5 байт), который зависит от конкретной операции. Первый, а иногда и единственный, байт, посылаемый микрокодом центральной части - это код операции. Код операции используется консольной программой как смещение для индексации таблицы выполняемых операций.

Один байт (только смещение) в консольную программу передают две операции: останов программы и UNIBUS INIT (инициализация ОШ). Останов программы вызывает переход в консольный режим, после чего микрокод центральной части посылает 10-байтовый пакет адреса останова. Все другие виды пересылок по шине консоли в программном режиме вызываются прерываниями или командами центральной части MTPR и MFPR, и в эти пересылки включаются дополнительные данные. Например, во время загрузки данных, управляющих таймером (команда MTPR, адресующая ICCS), вслед за смещением микрокод центральной части передает сам сайт данных (команду таймеру). Подобным образом вслед за смещением передаются 4 байта дополнительных данных при загрузке следующего счетчика интервала (MTPR, адресующая NICR).

Ряд других операций в программном режиме вызывает пересылку данных в микропрограмму центральной части после того, как она послала в консоль смещение. Это, естественно, происходит при обработке команды MFPR, адре-

5.055.430P3

сущей регистры консольного терминала, таймера, часов времени года и кассетного магнитофона. Например, при чтении состояния таймера (MFPR, адресующая ICSS) в ответ на смещение консольная программа возвращает микропрограмме центральной части один байт (состояние таймера). MFPR, адресующие другие регистры консольных устройств, могут вызвать возврат двух или четырех байт.

Кроме команды MFPR, возврат данных из консоли в процессор вызывает последовательность подтверждения прерывания. При работе консольного терминала и/или кассетного магнитофона по прерыванию, после получения байта с магнитофона или терминала консольная программа выставляет сигнал CONS ATTN. Сигнал CONS ATTN производит прерывание программы, выполняющейся в центральной части. В ответ на это микропрограмма центральной части посылает код операции (смещение), заставляющий консольную программу вернуть байт, запрос передачи которого вызвал прерывание.

В программном режиме передачи каждого байта данных управляется и синхронизируется сигналами "внимание к центральной части" (CPU ATTN) и "подтверждение от консоли" (CONS ACK). При пересылке данных из центральной части в консольный процессор микропрограмма центральной части взводит и убирает CPU ATTN, чтобы проинформировать о загрузке очередного байта в CRR. Консольная программа после чтения CRR изменяет сигнал CONS ACK.

При пересылке данных из консоли в процессор консольная программа использует установку/снятие CONS ACK для того, чтобы проинформировать о загрузке очередного байта в CWR, а микропрограмма центральной части после каждого чтения из CWR изменяет сигнал CPU ATTN.

В программном режиме консольная программа проверяет корректность ответа микрокодов центральной части также, как в консольном режиме. Если микропрограмма центральной части после изменения CONS ACK за определенный промежуток времени не ответит изменением CPU ATTN (если это требуется), то происходит прерывание программного режима по тайм-ауту и ЭМ входит в консольный режим. Консольная программа напечатает сообщение об ошибке связи и перейдет в холостой цикл.

5.6. Микропрограммная память

Блок-схема микропрограммной памяти приведена на рис.29 приложения 4. Основными компонентами являются 16К базовой микропрограммной памяти, 4К дополнительной пользовательской микропрограммной памяти, регистр записи микропрограммной памяти и регистр микрокоманд (CSR). Микроинструкции считываются либо из базовой, либо из пользовательской микропрограммной памяти на шину микроп-

рограммной памяти (CS) и в регистр микрокоманд (CSR). Шина CS также используется для загрузки микроинструкций в базовую или пользовательскую микропрограммную память из регистра записи микропрограммной памяти.

Главные отличия между базовой и пользовательской микропрограммными памятьми заключаются в следующем. Базовая микропрограммная память использует динамические микросхемы, которые должны периодически "регенерироваться". Поэтому требуются соответствующие вентиляные микросхемы, которые генерируют мультиплексированный адрес строки/столбца, счетчик адреса регенерации и схема синхронизации, которая вырабатывает соответствующие сигналы выборки адреса и сигналы стробирования.

Меньшая по объему пользовательская микропрограммная память использует статические микросхемы в качестве запоминающих элементов, поэтому обновления информации в памяти не требуется. Также нет необходимости в мультиплексировании адреса, имеется лишь дополнительная схема дешифрации банка памяти, которая выбирает соответствующие микросхемы.

Адреса базовой микропрограммной памяти лежат в пределах от 0 до 16К, а пользовательской памяти от 16К до 20К.

5.6.1. Базовая микропрограммная память

Динамическая базовая микропрограммная память построена на МОН чипах емкостью 16Кх1 бит. 24 чипа, по одному на каждый бит микрослова, образуют общую емкость памяти 16К 24-разрядных ячеек.

Базовая микропрограммная память имеет 7 адресных линий CS ADRS(6:0), которые параллельно соединены со всеми 24 чипами. Для доступа к ячейке в чипе (128 строк, 128 столбцов) адрес проходит через вентили, которые управляют выдачей на линии CS ADRS сначала 7-разрядного адреса строки, а затем 7-разрядного адреса столбца. Адреса строки и столбца получаются из адреса следующей микрокоманды, сформированного микросеквенсором на линиях NAD (см. табл. 15).

Таблица 15

Адресация базовой микропрограммной памяти

Линии NAD	Функция адресации
13:11, 03:00	выбор 1 из 128 возможных адресов строки.
10:04	выбор 1 из 128 возможных адресов столбца.

Поскольку адреса базовой микропрограммной памяти лежат в пределах от 0 до 16K (линии NAD(14:00) = 0000:3FFF), то старшая линия NAD (NAD14 = 1) используется для запрета доступа к базовой микропрограммной памяти путем запрета стробирования адреса столбца запоминающей матрицы. Формирование стробов адреса рассматривается в п.3.6.4.

3.6.2. Пользовательская микропрограммная память

Статическая пользовательская микропрограммная память построена на МУП чипах емкостью 1Kx4 бита. 24 чипа образуют общую емкость памяти 4K 24-разрядных ячеек. Все 24 чипа подразделяются на 4 банка (1K x 24 бит в каждом) по 6 чипов в каждом банке. Всякий раз, когда адресуется пользовательская микропрограммная память, выбирается один из этих банков в соответствии с одним из четырех выходов дешибратора выбора банка. Сигналы выбора банка S₂L XCF(3:0) соединяются со входами выборки соответствующих чипов (см. табл. 16).

Пользовательская микропрограммная память адресуется непосредственно линиями NAD микросеквенсера. Две линии NAD выбирают 1K-банк, а 10 младших линий NAD, которые параллельно соединены со всеми чипами в запоминающей матрице, выбирают ячейку внутри выбранного банка.

Таблица 16

Адресация пользовательской микропрограммной памяти

Линии NAD	функция адресации
11:10	Выбор одного из четырех возможных банков.
09:00	Выбор одной из 1к возможных ячеек внутри банка.

Поскольку адреса пользовательской микропрограммной памяти лежат в пределах от 16К до 20К (линии NAD(14:00) = 4000:43FF), то линия NAD(14) = 0 используется для блокировки дешифрации выбора банка и таким образом для запрета работы всех чипов в запоминающей матрице, когда нет обращения к пользовательской микропрограммной памяти.

5.6.3. Регистр микрокоманд (CSR)

Регистр микрокоманд обычно загружается непосредственно с шины микропрограммной памяти (CS) для хранения текущей микроинструкции. Это выполняется путем параллельной загрузки всех 24 разрядов микроинструкции либо из базовой, либо из пользовательской микропрограммной памяти. CSR имеет также возможность сдвига. Режимом загрузки CSR управляет сигнал PARAL LD CSR, который обычно вырабатывается консольным процессором.

Однако, консольная микродиагностика может загружать и прозирать CS путем сброса PARAL LD CSR и сдвига в младший бит CSR тестовых данных. Биты 07, 15 и 23 регистра CSR могут затем опрашиваться по мере того, как данные двигаются по регистру. CSR SHF IN, который может быть установлен или сброшен микродиагностикой, обеспечивает сдвиг входных данных. Строб регистра CSR (CLOCK CSR) вырабатывается микродиагностикой при операциях сдвига.

5.6.4. Основной микроцикл

Три фазы синхроимпульсов центральной части (0, 1 и 2) составляют микроцикл. Момент, в который CSR стробируется сигналом CLK CSR для загрузки текущей микроинструкции, определяется как T0; это является началом микроцикла и все другие времена в процессоре отсчитываются от этого момента.

Например, запись в местную память (LS) процессора производится по импульсу LS WRG EN, возбуждаемому от T225 до T270 микроцикла. Временная диаграмма основного микроцикла показана на рис. 30 приложения 4. Заметим, что T270 данного микроцикла совпадает с T0 следующего микроцикла.

Один микроцикл обычно соответствует одному микро-состоянию: т.е. 270 ns - интервал времени от одного синхроимпульса процессора фазы 0 до следующего. Однако, во время останова синхроимпульсов, импульс центральной части фазы 2 запрещается, поэтому CLK CSR и синхроимпульсы процессора не возбуждаются в течение некоторого отрезка времени в зависимости от условия останова. Как результат, T270 может быть задержан, тем самым продлевая микроцикл на время кратное 270 ns (т.е. 270 ns, 540 ns, 810 ns и т.д.).

Работу микропрограммной памяти во время основного микроцикла можно описать следующим образом.

В момент T0 восходящий фронт CLK CSR загружает содержимое следующего микроадреса с шины CS непосредственно в CSR. Эта новая микроинструкция становится текущей микроинструкцией, оставаясь в CSR весь микроцикл.

Как только текущая микроинструкция загружается в CSR, его управляющие биты начинают оказывать воздействие на аппаратуру. Функции, определенные микроинструкцией, выполняются в течение микроцикла. Большинство функций завершают выполнение в момент T270.

Управляющие биты текущей микроинструкции участвуют также в формировании следующего микроадреса, генерируемого микросеквенсором. Вскоре после загрузки CSR в начале микроцикла, новый адрес передается к управляющей памяти по линиям NAD(14:00) микросеквенсора. 15 линий NAD адресуют базовую и пользовательскую микропрограммные памяти, как это описано в пунктах 5.6.1. и 5.6.2.

Для адресов базовой памяти (NAD14 = 0) адреса управляющей памяти пропускаются на М0П чипы запоминающей матрицы поочередно: сначала адрес строки, затем адрес столбца. Мультиплексор управляется двумя сигналами: SEL ROW ADRS и SEL COL ADRS. Два других сигнала: ROW ADRS STB (RAS) и COL ADRS STB (CAS) "защелкивают" адреса строки и столбца в М0П чипах. Как только адрес столбца защелкивается, содержимое следующего микроадреса считывается на шину CS.

Для адресов пользовательской памяти (NAD14 = 1) никакого мультиплексирования адресных линий не требуется. Линии NAD поступают непосредственно на адресные входы чипов запоминающей матрицы, а содержимое следующего микроадреса читается на шину CS вслед за возбуждением сигнала выбора банка на дешифраторе выбора банка.

После того, как содержимое следующего микроадреса считано из базовой или пользовательской микропрограммной памяти, содержимое шины CS загружается в CSR, чтобы стать текущей микроинструкцией в момент появления восходящего фронта очередного сигнала CLOCK CSR.

При адресации базовой микропрограммной памяти строки адреса и управляющие сигналы генерируются схемой ZkTugdTkзвзкк. Qyf}b ZkTugdTk%зкк {fTfrkгweж вягацмWв. щие сигналы, используя основной синхроимпульс, задержанный основной синхроимпульс и фазы 0 и 1. Упрощенная схема базовой синхронизации приведена на рис. 31 приложения 4. Заметим, что здесь генерируется более, чем один сигнал выбора адреса строки и столбца. Это делается для того, чтобы не было совпадения с соответствующим стробом адреса, которое может повлиять на операции базовой микропрограммной памяти.

Как было установлено ранее, во время обращения к пользовательской микропрограммной памяти сигнал NAD14 = 1 запрещает генерацию строба адреса столбца. Сигналы выбора адресов строки и столбца хотя и генерируются, но не оказывают никакого воздействия на работу базовой памяти.

и после интервала записи (посредством WCS ENABLE = 0). Более того, в течение цикла регенерации (REFR CYCLE = 1) все выходы схемы синхронизации запрещены за исключением строба адреса столбца.

Только базовая микропрограммная память требует регенерации. Подобно динамическим микросхемам в консольном процессоре, использующем те же самые МОП чипы емкостью 16кx1 бит в корпусе (128 строк x 128 столбцов), базовая микропрограммная память регенерируется при помощи серии сигналов строба строки на все 128 адресуемых строк в течение интервала регенерации. Цикл регенерации (который обновляет один адрес строки) осуществляется каждые 12,8 мкс.

Цикл регенерации микропрограммной памяти инициируется схемой управления регенерацией в консольном процессоре. Сигнал запроса регенерации CON STORE REFR (период 12,8 мкс) вырабатывается счетчиком, который постоянно стробируется от генератора микропроцессора. Сигнал CON STORE REFR синхронизирован с микросостоянием микропрограммной памяти посредством трех триггеров синхронизации (триггера стробируются сигналом CLK REFR CNTR) для того, чтобы генерировать REFR REQ, когда уровень сигнала CON STORE REFR становится низким. Временная диаграмма показана на рис. 32 приложения 4. Сигнал REFR REQ, появляющийся на одно микросостояние, инициирует цикл регенера-

5.055.430PЭ

ции управляющей памяти. Он инициирует также цикл регенерации основной памяти путем возбуждения сигнала MAIN MEM REFR. Цикл регенерации основной памяти, который возникает после цикла регенерации микропрограммной памяти, обсуждается в техническом описании системы памяти.

Временная диаграмма цикла регенерации микропрограммной памяти, инициированного сигналом REFR REQ, приведена на рис. 33 приложения 4. Сигнал REFR REQ сначала возбуждает сигнал REFR CYCLE. Затем, в течение микросостояния, REFR REQ останавливает импульсы процессора фазы 2, чтобы приостановить микроцикл, а REFR CYCLE управляет вентилями stroba адреса микропрограммной памяти так, что выходы 7-разрядного счетчика регенерации (адрес строки) передаются на адресные линии микропрограммной памяти. Кроме того, REFR CYCLE запрещает все сигналы выбора и стробирования адреса микропрограммной памяти, за исключением stroba адреса строки. Этот сигнал (ROW ADRS STB) затем возбуждается в свое обычное время для регенерации выбранного адреса строки в конце цикла регенерации.

Счетчик регенерации REFR ADRS (6:0) инкрементируется по падающему фронту REFR CYCLE в конце каждого цикла регенерации и проходит по всем 128 возможным адресам в течение 1,7 мс интервала обновления.

ПРИМЕЧАНИЕ. Сигнал SEE DC LO от консольного процессора возбуждает REFR CYCLE и вызывает непрерывные циклы регенерации во время последовательностей включения и выключения питания системы. Скачки постоянного напряжения во время нормального цикла чтения микропрограммной памяти могут привести к изменению содержимого микропрограммной памяти.

5.6.6. Операция записи микропрограммной памяти

Микропрограммная память записывается консольным процессором во время начальной загрузки системы, посредством некоторых консольных команд, а также при микродиагностике. Операция выполняется, когда центральная часть находится в режиме обслуживания (импульсы центральной части фазы 2 остановлены). Одно 24-разрядное слово микропрограммной памяти записывается за одну операцию записи.

Консольная программа (или консольный микродиагностический монитор) загружает три байта записываемых данных в регистр записи микропрограммной памяти. Она же затем возбуждает сигнал WRITE WCS. Это заставляет набор четырех триггеров синхронизации (которые стробируются импульсом фазы 0) снять сигнал WCS ENABLE и возбудить сигналы EN WCS Wk DATA и WCS WRT EN. Временная диаграмма приведена на рис. 33 приложения 4.

ПРИМЕЧАНИЕ. Консольная программа (или микродиагностический монитор), работающая под управлением микропроцессора, записывает одну ячейку микропрограммной памяти путем сдвига микроадреса UPC (сигнал CLK UPC является одnorазовым в течение этой операции). При выполнении микроинструкции NOP, ранее загруженной консольным микропроцессором в CSR, микроадрес из регистра адреса следующей микрокоманды по линиям NAD передается на адресные входы микропрограммной памяти.

Обычно установленный сигнал WCS ENABLE разрешает генерирование стробов адреса для базовой микропрограммной памяти. Кроме того он разрешает выработку сигналов выбора банка для пользовательской микропрограммной памяти. Таким образом, когда WCS ENABLE сбрасывается в начале операции записи микропрограммной памяти, это предохраняет одну из запоминающих матриц от взаимодействия с шиной CS.

Когда CS неактивна, два других сигнала генерируются триггерами синхронизации для выполнения следующих действий. EN XCS WRT DATA устанавливается через 270 нс после того, как WCS ENABLE станет неактивным, и разрешает регистру записи микропрограммной памяти выдать данные на шину CS.

EN WCS WRT EN, генерируемый в то же самое время, что и EN XCS WRT DATA, разрешает запись в микропрограммную память (сигналы разрешения поступают на контакты разрешения записи в чипы микропрограммной памяти). Разрешение записи базовой микропрограммной памяти (CS WRITE EN) зависит от REFR REQ. Разрешение записи пользовательской микропрограммной памяти (WRT XCS) имеет место только тогда, когда NAD14 = 1.

Когда регистр записи микропрограммной памяти выходит на шину CS и чипы запоминающей микропрограммной памяти разрешены для записи, WCS ENABLE вновь переустанавливается триггерами синхронизации в течение всего времени, пока запрос записи, генерируемый консолью, остается истинным. Запрос будет истинным в течение нескольких микросостояний процессора из-за сравнительно медленной скорости консольного микропроцессора. В течение времени, когда WCS ENABLE вновь переустанавливается, опять разрешается формирование сигналов выбора банка и строба адреса столбца. Это время называется интервалом записи. В результате данные регистра записи, переданные по шине CS, записываются в адресуемую ячейку микропрограммной памяти.

В течение интервала записи одни и те же данные записываются по одному и тому же адресу в нескольких микросостояниях процессора. Временная диаграмма единичного цикла записи приведена на рис.33 приложения 4.

Если запрос на регенерацию возникнет во время операции записи базовой микропрограммной памяти, REFR CYCLE

5.055.430PЭ

запрещает выработку сигнала разрешения записи запоминающей микропрограммной матрицы. Поскольку цикл регенерации имеет место только в течение одного микросостояния, и поскольку запись в одну ячейку микропрограммной памяти повторяется в течение нескольких микросостояний, то запрос регенерации не оказывает никакого влияния на возникновение и протекание операции записи базовой микропрограммной памяти.

Запись в пользовательскую микропрограммную память, которая использует статические микросхемы, тем более не запрещается циклом регенерации. Оба цикла: записи в пользовательскую и регенерации базовой микропрограммной памяти могут происходить одновременно.

Интервал записи микропрограммной памяти заканчивается, когда консольный процессор сбрасывает $WRITE\ WCS$, который в свою очередь сбрасывает $WCS\ ENABLE$. Затем (через 270 ns) сбрасываются разрешение записи чипа и разрешение выходов регистра записи. Ничего не выдается на шину CS . Сигнал $WCS\ ENABLE$ возвращается к нормальному состоянию, завершая операцию записи микропрограммной памяти.

5.6.7. Паритет микропрограммной памяти и микросинхронизация

Все микроинструкции в микропрограммной памяти содержат бит паритета (бит 25), который генерируется программой микроассемблера при создании микрокода. Правильный паритет является нечетным и он проверяется всякий раз, когда текущая микроинструкция загружается в CSR .

Если паритет текущей микроинструкции неверный (четный), и если останов по ошибке разрешен консольным процессором ($HALT\ ON\ PE = 1$), то возникает сигнал $CS\ PARITY\ ERR$, чтобы остановить импульс базы Z до того, как микроинструкция может выполняться. Сигнал ошибки, возбуждая $PAR\ ERR$, прерывает консольный процессор. Прерывания возникают обычно в момент $T270$ микроцикла. Консольная программа может затем прочитать UPC , который содержит инкрементированный на единицу адрес неправильной микроинструкции, и прервать работу центральной части.

Схемы проверки паритета микропрограммной памяти могут использоваться для генерации синхроимпульса осциллографа, что является полезным средством при обслуживании комплекса $CM\ 170U$. Этот сигнал микросинхронизации генерируется путем загрузки неверного паритета по нужному микроадресу и путем сброса сигнала $HALT\ ON\ PE$, чтобы предотвратить останов по ошибке. Оба эти действия могут быть выполнены при помощи консольных команд. При этом центральная часть будет работать с нормальной скоростью, а сигнал синхронизации осциллографа будет генерироваться

всякий раз, когда микроинструкция с неверным паритетом будет становиться текущей микроинструкцией.

Сигнал микросинхронизации выводится на контрольную точку TP1 в модуле процессора для облегчения его использования. Этот сигнал возбуждается во время импульса фазы Z. Таким образом, выполнение микроинструкций, генерирующей сигнал синхронизации, может быть наблюдаемо на осциллографе.

Во время тестирования запоминающих матриц микропрограммной памяти при помощи микродиагностики, регистр CSR загружается тестовыми данными, имеющими неверный паритет. В этих случаях консольный процессор может предотвратить останов генератора (если не задан одношаговый режим работы консоли) путем сброса HALT ON PE.

6. ПОДГОТОВКА К РАБОТЕ

6.1. Порядок установки

Помещение, в котором устанавливается и будет эксплуатироваться консольный процессор, должно быть отапливаемым в зимнее время. Перед установкой консольного процессора необходимо его запаковать и провести расконсервацию. Распаковка в зимнее время должна проводиться в отапливаемом помещении, с предварительной выдержкой в нераспакованном виде в нормальных климатических условиях (температура воздуха от 15 до 25 градусов С, относительная влажность от 50 до 80%) в течении 12 часов. Должна быть проведена проверка комплектности эксплуатационных документов и микродиагностических тестов консольного процессора. Следует провести внешний осмотр, проконтролировать на отсутствие механических повреждений и дефектов, коррозии, царапин, отсутствие посторонних предметов и повреждений в блоках элементов, удалить пыль.

6.2. Персональное включение

Подача напряжений 5V, +15V и -15V осуществляется при включении комплекса СМ 1700. Проверка уровня электропитания осуществляется в комплексе СМ 1700.

7. ИЗМЕРЕНИЕ ПАРАМЕТРОВ, РЕГУЛИРОВАНИЕ И НАСТРОЙКА

Перед установкой консольного процессора в ВМ СМ 2700 необходимо убедиться в отсутствии короткого замыкания между шинами питания +5V и 0V, +/-15V и 0V. Измерение рекомендуется производить эвометром Ц 4353 (класс точности 1,5), установленным в режим измерения сопротивления. Возможно применение других приборов аналогичного класса. При отсутствии короткого замыкания между шинами питания, установить консольный процессор в ВМ СМ 2700, включить питание и измерить напряжение между выводами электролитического конденсатора С13 для +5V и на диодах VD4 для -15V и VD1 для +15V.

Дальнейшую проверку консольного процессора следует производить согласно разделу 9 настоящего РЭ.

8. ТЕХНИЧЕСКОЕ ОБСЛУЖИВАНИЕ

Техническое обслуживание консольного процессора осуществляется подготовленным персоналом, имеющим право на техническое обслуживание ЭВМ, в которую входит консольный процессор.

Техническое обслуживание консольного процессора должно производиться в помещениях, гарантирующих защиту консольного процессора от воздействия пыли, агрессивной среды и электромагнитных полей.

Для обеспечения бесперебойной работы консольного процессора необходимо производить ежеквартальные контрольно-профилактические работы. При проведении этих работ консольный процессор исключается из состава ЭВМ, в которой он находится.

Продолжительность проведения технического обслуживания зависит от состояния консольного процессора, но не превышает 1 часа.

В ежеквартальные контрольно-профилактические работы входят:

внешний осмотр;

удаление пыли и грязи;

чистка контактных блоков элементов;

проверка монтажных соединений.

Для проведения различных видов технического обслуживания требуются материалы:

отходы хлопчатобумажных материалов по ГОСТ 4644-75;

спирт этиловый ректификованный высшей сорта по ГОСТ 18300-72.

Объем и методика контрольно-профилактических работ:

внешний осмотр. На поверхности консольного процессора не должно быть царапин, вмятин и других пов-

реждений;

удаление пыли и грязи выполняется чистой сухой хлопчатобумажной тканью;

чистка контактов блоков элементов производится медицинской марлей, смоченной спиртом;

проверка монтажных соединений производится визуально. Проверяется состояние монтажных соединений и паяк, при этом не должно быть нарушений изоляции, пайки должны быть покрыты лаком.

9. ВОЗМОЖНЫЕ НЕИСПРАВНОСТИ, СПОСОБЫ ИХ ОБНАРУЖЕНИЯ И УСТРАНЕНИЯ

Возможными неисправностями консольного процессора являются:

выход из строя интегральных микросхем;
обрыв линий связи.

Методом обнаружения характерных неисправностей является прогон микродиагностических тестов (ENKBA, ENKBB, ENKBC, ENKBD, ENKBE, ENKBF) и теста самопроверки консольного процессора после включения питания.

Успешное выполнение теста самопроверки сопровождается вызовом на консольный терминал сообщения <ЭК><ПС> CM-1700.

Загрузка микродиагностических тестов консольного процессора:

поставить кассету с системой микродиагностических тестов в любой механизм кассетной магнитной ленты;

повернуть ключ включения питания в положение "ВКЛ" и ждать появления идентификатора системы микродиагностического обеспечения MIC> на консольном терминале;

набрать на консольном терминале следующие командные строки:

MIC>SE TR<ЭК> - установка флага трассы для наблюдения за выполнением микротестов;

MIC>DI BO WCS<ЭК> - запуск микротестов проверяющих консольный процессор;

MIC>DI SE ENKBF<ЭК> - запуск микротеста проверки микропрограммной памяти.

При прогоне микротестов консольного процессора в случае обнаружения ошибки на консольном терминале, появится сообщение:

SECT TST ERR EXP REC OTHER MASK MODULE

SECT - имя микротеста;

TST - номер микротеста, где ошибка;

ERR - номер ошибки;

EXP - ожидаемые данные;

REC - полученные данные;

OTHER - дополнительная информация;

MASK - маска ошибок. Биты установленные в 1 в маске, соответствуют битам результата, которые не проверяются;

MODULE - предполагаемый источник ошибок.

Подробно предполагаемые источники ошибок приведены в 00016-01 13 01 "Система микродиагностического обеспе-

3.055.430РЭ

чения ЭК СМ 1700, Описание программы."

10. ТРАНСПОРТИРОВАНИЕ И ХРАНЕНИЕ

Транспортирование консольного процессора производить в упакованном виде в транспортной таре всеми видами транспорта на любые расстояния при условии воздействия следующих климатических факторов:

температура окружающего воздуха от -50 до +50 градусов С;

относительная влажность воздуха при температуре +30 градусов С до 90%;

атмосферное давление от 84 до 107 кПа;

транспортная тряска с ускорением 3 G при частоте ударов от 80 до 120 в минуту.

При погрузке и транспортировании должны строго выполняться требования предупредительных надписей.

Консольный процессор должен храниться в складских помещениях в упаковке при температуре от +15 до +25 градусов С и относительной влажности воздуха не более 85% на стеллажах.

ПРИЛОЖЕНИЕ 1

Таблица 1
Распределение символов по контактам разъема ОШ

Номер разъема	Номер контакта ряда А	Сигнал	Номер разъема	Номер контакта ряда С	Сигнал
XP1	1		XP1	1	
	2			2	
	3			3	
	4			4	
	5	OV		5	OV
	6	ARRAY REF CYC L		6	
	7			7	BUS ARRAY A2 H
	8	BUS ARRAY A1 H		8	
	9			9	BUS ARRAY A3 H
	10	OV		10	OV
	11			11	BUS ARRAY A0 H
	12	BUS ARRAY A6 H		12	
	13	+5,0 V, A		13	+5,0 V, A
	14	MCTK CAS TIM L		14	MCTK DR EN
	15			15	0 V
	16	BUS ARRAY A5 H		16	
	17	MCTK WRT TIM L		17	XMIT OUT 2
	18			18	ARRAY RAS TIM L
	19	REC IN L		19	PANEL REMOTE H
	20	OV		20	OV
	21	CPU +5V SENSE H		21	
	22	RESET 8085 L		22	BUS ARRAY A4 H
	23			23	BUS ARRAY A7 H
	24			24	
	25			25	DISABLE

S.055.430PЭ

Продолжение табл. 1

Номер разъема	Номер контакта ряда А	Сигнал	Номер разъема	Номер контакта ряда С	Сигнал
					CLK L
	26			26	
	27			27	
	28			28	
	29			29	
	30	UV		30	UV
	31			31	
	32	+5,0V		32	+5,0V
XP2	1		XP2	1	
	2			2	
	3			3	
	4	MCTF ERR SUM H		4	UV
	5	UV		5	UV
	6			6	
	7			7	
	8			8	BUS MC D07 H
	9			9	
	10	UV		10	UV
	11			11	
	12			12	BUS MC D15 H
	13	+5,0V		13	+5,0V
	14	BUS MC D14 H		14	
	15	BUS MC D06 H		15	BUS MC D13 H
	16	UBS D15 L		16	BUS MC D05 H
	17	UBS D14 L		17	BUS MC D04 H
	18			18	UBS D15 L
	19	BUS MC D05 H		19	UBS D12 L
	20	UV		20	UV
	21	BUS MC D11 H		21	BUS MC D12 H
	22	BUS MC D02 H		22	UBS D11 L
	23	BUS MC D01 H		23	UBS D10 L
	24	BUS MC D10 H		24	UBS D09 L
	25	UBS D08 L		25	UV
	26			26	
	27	UBS D07 L		27	BUS MC D09 H
	28	UBS DCLO L		28	BUS MC D00 H
	29	UBS D04 L		29	BUS MC

Продолжение табл. 1

Номер разъема	Номер контакта ряда А	Сигнал	Номер разъема	Номер контакта ряда С	Сигнал
					D08 H
	30	0V		30	0V
	31	UBS D01 L		31	UBS D05 L
	32	+5,0V		32	+5,0V
XP3	1	+15V	XP3	1	+15V
	2			2	UBS D00 L
	3			3	UBS D03 L
	4	UBS D02 L		4	UBS D06 L
	5	0V		5	0V
	6	UBS ACLO L		6	CPU P2B H
	7	CPU CLOCK L		7	
	8	CPU P2 H		8	BUS MC
	9	MCT CLOCK L		9	D07 H
	10	0V		9	PORT
	11	UBS INT T L		10	CLOCK L
	12	BUS CONS		11	0V
	13	D07 H		11	SEL CPU
	14	+5V		12	REGS L
	15	DC ON L		12	LATCH
	16	RUN L		13	ADRS L
	17	REMOTE L		13	+5V
	18	DAPK MPC		14	BUS CONS
	19	SHF OUT H		14	D06 H
	20			15	0V
	21	BUS CONS		15	BUS CONS
	22	D02 H		16	D05 H
	23	0V		17	BUS CONS
	24	CLK(D) L		17	D04 H
	25	MCT CTL 0 H		18	BUS CONS
	26			18	D03 H
	27	DAPK NAD		19	MISC IN
	28	14 L		19	2 H
	29	DAPK NAD		20	0V
	30	13 L		20	BUS CONS
		-15V		21	D01 H
		DAPB CLOCK		21	BUS CONS
		STALL L		22	D00 H
		DAPK NAD		22	MCT CTL
		12 L		23	1 H
		DAPK NAD		23	CLK(1) L
		11 L		24	CSR 23 H
		0V		24	-15V
				25	CSR 22H
				26	
				27	CLOCK P2 L
				28	CSR 21 H
				29	
				30	0V

Продолжение табл. 7

Номер разъема	Номер контакта ряда А	Сигнал	Номер разъема	Номер контакта ряда С	Сигнал
XP4	31	PANEL BOOT H	XP4	31	CSR 20 H
	32	+5V		32	+5V
	1			1	
	2			2	PANEL DISABLE H
	3	CSR 19 H		3	PANEL HALT H
	4	CSR 18 H		4	UV
	5	UV		5	UV
	6	CSR 17 H		6	PANEL +5V FILT
	7	CSR 16 H		7	DAPL PARAL LD CSR H
	8	DAPK NAD U0 L		8	MCTM UB DATA EN L
	9	CSR 15 H		9	MEMORY ERR H
	10	UV		10	UV
	11	CSR 14 H		11	CINIT L
	12	MCTM ALLOW REF H		12	CSR 13 H
	13	+5V		13	+5V
	14	CSR 12 H		14	
	15	CSR 11 H		15	CSR 10 H
	16	START REF CYC L		16	CSR 09 H
	17	DAPK NAD U9 L		17	CSR 08 H
	18			18	
	19	CSR 07 H		19	DAPK NAD U8 L
	20	UV		20	UV
	21	CSR 06 H		21	
	22	UBS BBSY L		22	DAPK NAD U7 H
	23	CSR 05 H		23	DAPB CS PAR ERR H
	24	CSR 04 H		24	DAPK NAD U6 L
	25	DAPK NAD U5 L		25	UV
	26	CSR 00 H		26	
	27	DAPK NAD U4 L		27	CSR 03 H
	28	DAPK NAD U3 L		28	CSR 02 H
	29	DAPK NAD U2 L		29	CSR 01 H
30	UV	30	UV		
31	DAPK NAD	31	DAPK NAD		

Продолжение табл. 1

Номер разъема	Номер контакта ряда А	Сигнал	Номер разъема	Номер контакта ряда С	Сигнал
	32	00 L +5V		32	01 L +5V

ПРИЛОЖЕНИЕ 2

Система команд микропроцессора

Введение

Микропроцессор имеет шесть 8-битных регистров: В, С, D, E, H и L, которые могут объединяться в 16-битные пары BC, DE, HL, 8-битный аккумулятор А, 16-битный указатель стека SP, и счетчик адреса PC.

Номера регистров сведены в табл.1.

Таблица 1
Номера регистров микропроцессора

Имя регистра	Десятичный номер	Двоичный номер
В	0	000
С	1	001
D	2	010
E	3	011
H	4	100
L	5	101
A	7	111

При обращении к регистру номер 6 (110) реализуется обращение к байту памяти, адрес которого содержится в регистровой паре HL.

В микропроцессоре имеется 5 флагов условий, объединенных в специальный байт состояния (его имя "F"), значения которых приведены в табл. 2.

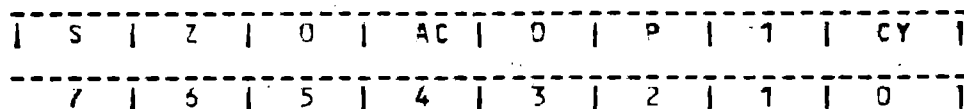
Система команд микропроцессора приведена в приложении 2, описания команд RIM и SIM в приложении 3.

Таблица 2

Флаги условий микропроцессора

Имя флага	Когда устанавливается
C (перенос)	При последней операции был перенос из старшего бита.
P (четность)	Количество единиц в результате четно.
AC (дополнительный перенос)	был перенос из 3-го разряда в 4-й.
Z (ноль)	Результат операции - ноль.
S (знак)	Результат операции отрицателен (т.е. 7-й бит = 1).

Структура байта состояния:



Принятые обозначения

1, 2

- один из регистров А, В, С, D, E, H, L или ссылка на память (номер регистра - б);

F - байт состояния (флаги условий);

SSS - двоичный номер регистра-источника;

DDD - двоичный номер регистра-приемника.

SP

- регистр-указатель стека;

PC

- счетчик адреса;

()

- содержимое регистра, регистровой пары или ячейки памяти, адрес или имя (для регистров) которой указан в скобках;

[] - содержимое ячейки памяти, адрес которой находится в регистровой паре, SP и т.д. Имя пары указано в скобках;

< > байт данных;

(SP), (SP) содержимое старшего (H) и младшего (L) байтов SP;

AA<-BB пересылка из BB в AA. Например A)<-L(B,C)] пересылка байта, адресуемого парой BC, в аккумулятор;

э побитовое исключающее "или";

l побитовое включающее "или".

Таблица 5

Таблица команд микропроцессора

Мнемоника	Код опции (байт 1)	Байт2	Байт3	Тактов	Описание операции
MOV R1, R2	01000SSS			4	(R1) ← (R2)
MOV M, R	01110SSS			7	(M) ← (R)
MOV R, M	01000110			7	(R) ← (M)
HLT	01110110			5	Останов
MVI R	00000110	B2		7	(R) ← <B2>
MVI M	00110110	B2		10	(M) ← <B2>
INR R	00000100			5	(R) ← (R) + 1
DCR R	00000101			5	(R) ← (R) - 1
INR M	00110100			10	(M) ← (M) + 1
DCR M	00110101			10	(M) ← (M) - 1
ADD R	10000SSS			4	(A) ← (A) + (R)
ADC R	10001SSS			4	(A) ← (A) + (R) + C
SUB R	10010SSS			4	(A) ← (A) - (R)
SBB R	10011SSS			4	(A) ← (A) - (R) - C

Продолжение табл. 3

Мнемоника	Код опции (байт 1)	Байт2	Байт3	Тактов	Описание операции
ANA R	10100SSS			4	(A)←-(A)&(R)
XRA R	10101SSS			4	(A)←-(A)э(R)
ORA R	10110SSS			4	(A)←-(A) (R)
CMP R	10111SSS			4	(A)-(R)
ADD M	10000110			7	(A)←-(A)+(M)
ADC M	10001110			7	(A)←-(A)+(M)+C
SUB M	10010110			7	(A)←-(A)-(M)
SBB M	10011110			7	(A)←-(A)-(M)-C
ANA M	10100110			7	(A)←-(A)&(M)
XRA M	10101110			7	(A)←-(A)э(M)
ORA M	10110110			7	(A)←-(A) (M)
SMP M	10111110			7	(A)-(M)
ADI	11000110	B2		7	(A)←-(A)+<B2>
ACI	11001110	B2		7	(A)←-(A)+<B2>+C
SUI	11010110	B2		7	(A)←-(A)-<B2>
SBI	11011110	B2		7	(A)←-(A)-<B2>-C
ANI	11100110	B2		7	(A)←-(A)&<B2>
XRI	11101110	B2		7	(A)←-(A)э<B2>
ORI	11110110	B2		7	(A)←-(A) <B2>
CPI	11111110	B2		7	(A)-<B2>
RLC	00000111			4	(A[M+1])←-(A[M]) (A[0])←-(A[7]) C←-(A[7])
RRC	00001111			4	(A[M-1])←-(A[M]) (A[7])←-(A[0]) C←-(A[0])
RAL	00010111			4	(A[M+1])←-(A[M]) (A[0])←-C C←-(A[7])
RAR	00011111			4	(A[M-1])←-(A[M])

Продолжение табл. 5

Мнемоника	Код оп-ции (байт 1)	Байт2	Байт3	Тактов	Описание операции
					(ALJ)←C C←(A[UJ])
DAA	00100111			4	десятичное пре- образование ак- кумулятора
CMA	00101111			4	(A)←~(A)
STC	00110111			4	C←1
CMC	00111111			4	C←~C
NOP	00000000			4	Отсутствие опе- рации
LXI B	00000001	B2	B3	10	(B)←<B3>, (C)←<B2>
LXI D	00010001	B2	B3	10	(D)←<B3>, (E)←<B2>
LXI H	00100001	B2	B3	10	(H)←<B3>, (L)←<B2>
LXI SP	00110001	B2	B3	10	(SP) LHJ←<B3> (SP) LLJ←<B2>
DAD B	00001001			11	(H)(L)← (H)(L)+(B)(C)
DAD D	00011001			11	(H)(L)← (H)(L)+(D)(E)
DAD H	00101001			11	(H)(L)← (H)(L)+(H)(L)
DAD SP	00111001			11	(H)(L)← (H)(L)+(SP)
STAX B	00000010			7	L(B)(C)J←(A)
STAX D	00010010			7	L(D)(E)J←(A)
LDAX B	00001010			7	(A)←L(B)(C)J
LDAX D	00011010			7	(A)←L(D)(E)J
SHLD	00100010	B2	B3	16	L<B3><B2>J←(L), L<B3><B2>+1J←(H)
LHLD	00101010	B2	B3	16	(L)←L<B3><B2>J, (H)←L<B3><B2>+1J

Продолжение табл. 3

Мнемоника	Код опции (Байт 1)	Байт2	Байт3	Тактов	Описание операции
STA	00110010	B2	B3	13	[<B3><B2>]<-(A)
LDA	00111010	B2	B3	13	(A)<- [<B3><B2>]
INX B	00000011			5	(B)(C)<-(B)(C)+1
INX D	00010011			5	(D)(E)<-(D)(E)+1
INX H	00100011			5	(H)(L)<-(H)(L)+1
INX SP	00110011			5	(SP)<-(SP)+1
DCX B	00001011			5	(B)(C)<-(B)(C)-1
DCX D	00011011			5	(D)(E)<-(D)(E)-1
DCX H	00101011			5	(H)(L)<-(H)(L)-1
DCX SP	00111011			5	(SP)<-(SP)-1
OUT	11010011	B2		10	Вывод на устрой- ство с номером B2
IN	11011011	B2		10	Ввод данных в аккумулятор с устройства B2
XTHL	11100011			17	(L)<->[SP], (H)<->[SP+1]
XCHG	11101011			4	(H)<->(D), (L)<->(E)
DI	11110011			4	Запрет преры- ваний
EI	11111011			4	Разрешение пре- рываний
PUSH B	11000101			11	[SP-1]<-(B), [SP-2]<-(C), (SP)<-(SP)-2
PUSH D	11010101			11	[SP-1]<-(D), [SP-2]<-(E), (SP)<-(SP)-2
PUSH H	11100101			11	[SP-1]<-(H), [SP-2]<-(L), (SP)<-(SP)-2

Продолжение табл. 5

Мнемоника	Код оп-ции (байт 1)	Байт2	Байт3	Тактов	Описание операции
PUSH PSW	11110101			11	LSP-1]←-(A), LSP-2]←-(F), (SP)←-(SP)-2
POP B	11000001			10	(C)←-LSP], (B)←-LSP+1], (SP)←-(SP)+2
POP D	11010001			10	(D)←-LSR], (E)←-LSP+1], (SP)←-(SP)+2
POP H	11100001			10	(H)←-LSR], (L)←-LSP+1], (SP)←-(SP)+2
POP PSW	11110001			10	(F)←-LSP], (A)←-LSP+1], (SP)←-(SP)+2
PCHL	11101001			5	(PC)←-(H)(L)
SPHL	11111001			4	(SP)←-(H)(L)
JMP	11000011	B2	B3	10	(PC)←-<B3><B2>
JNZ	11000010	B2	B3	10	Z=0 условный
JZ	11001010	B2	B3	10	Z=1 переход
JNC	11010010	B2	B3	10	C=0 Если усло-
JC	11011010	B2	B3	10	C=1 вие выпол-
JPO	11100010	B2	B3	10	P=0 няется,
JPE	11101010	B2	B3	10	P=1 (PC)←-
JP	11110010	B2	B3	10	S=0 <B3><B2>,
JM	11111010	B2	B3	10	S=1 иначе (PC)←-(PC)+3,
CALL	11001101	B2	B3	11	LSP-1]LSP-2]←- (PC), (SP)←-(SP)-2, (PC)←-<B3><B2>
CNZ	11000100	B2	B3	11/11	Z=0 Если усло-
CZ	11001100	B2	B3	11/11	Z=1 вие выпол-

Продолжение табл. 3

Мнемоника	Код опции (Байт 1)	Байт2	Байт3	Тактов	Описание операции
CNC	11010100	B2	B3	17/11	C=0 няется, [SP-1][SP-2]<-
CC	11011100	B2	B3	17/11	C=1 (PC),
CPO	11100100	B2	B3	17/11	P=0 (SP)<-
CPE	11101100	B2	B3	17/11	P=1 (SP)-2,
CP	11110100	B2	B3	17/11	S=0 (PC)<-
CM	11111100	B2	B3	17/11	S=1 <B3><B2>, иначе. (PC)<-(PC)+3
RET	11001001			11	(PC)<-[SP+1][SP], (SP)<-(SP)+2
RNZ	11000000			11/5	Z=0 Если усло-
RZ	11001000			11/5	Z=1 вие выпол-
RNC	11010000			11/5	C=0 няется,
RC	11011000			11/5	C=1 (PC)<-
RPO	11100000			11/5	P=0 [SP+1][SP],
RPE	11101000			11/5	P=1 (SP)<-
RP	11110000			11/5	S=0 (SP)+2,
RM	11111000			11/5	S=1 иначе (PC)<-(PC)+1
RST	11AAA111			11	[SP-1][SP-2]<- (PC), (SP)<- (SP)-2, (PC)<- <00000000AAA000>
RIM	00100000				Чтение маски прерываний
SIM	00110000				Установка маски прерываний

Таблица 4

Влияние команд на формирование флагов условий

Команда	Коды условия	C	Z	S	P	AC
INR			X	X	X	
DCR			X	X	X	
ADD		X	X	X	X	X
ADC		X	X	X	X	X
SUB		X	X	X	X	X
SBB		X	X	X	X	X
ANA		U	X	X	X	
XRA		U	X	X	X	
ORA		U	X	X	X	
CMP		X	X	X	X	X
ADI		X	X	X	X	X
ACI		X	X	X	X	X
SVI		X	X	X	X	X
SBI		X	X	X	X	X
ANI		U	X	X	X	
XRI		U	X	X	X	
ORI		U	X	X	X	
CPI		X	X	X	X	X
RLC		X				
RRC		X				
RAL		X				
RAR		X				
STC		1				
CMC		AC				
DAA		X	X	X	X	

Продолжение табл. 4

Команда	Коды условия	C	Z	S	P	AC
DAD		X				X
POP PSW		X	X	X	X	X

ПРИЛОЖЕНИЕ 3

Команды RIM и SIM

Микропроцессор, умея делать все, что и КЭ80ИК80, имеет следующие дополнения:

наличие системы из 3 маскируемых прерываний с жестко (аппаратно) заданными векторами и приоритетами. Прерывания имеют условные имена RST 5.5, RST 6.5, и RST 7.5 (расположены по возрастанию приоритета);

наличие возможности последовательного (поритного) ввода/вывода данных;

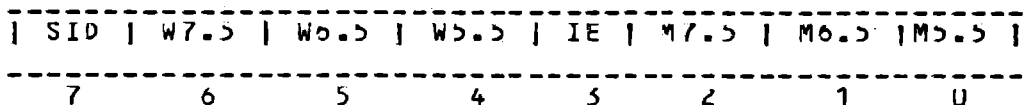
наличие немаскируемого прерывания с плавающим вектором (TRAP);

наличие команд RIM и SIM.

Прерывания (RST или INTR) могут быть запрещены (командой DI) или разрешены (командой EI). Если они разрешены, любое из 3-х прерываний RST может быть замаскировано командой SIM. Прерывание TRAP не маскируется и не запрещается.

Если прерывание RST приходит, когда прерывания запрещены, или прерывания разрешены, но пришедшее прерывание замаскировано, взводится соответствующий бит "ожидаемое прерывание" в маске прерываний. Когда реальный сигнал прерывания пропадает, бит "ожидаемое прерывание" сбрасывается для RST 5.5 и RST 6.5. Для RST 7.5 имеется триггер прерывания, взводимый по сигналу "прерывание RST 7.5" и сбрасываемый либо при переходе на обработку прерывания, либо явно командой SIM.

Команда RIM считывает маску прерываний в аккумулятор. После RIM аккумулятор имеет следующий формат:



где SID

этот бит соответствует сигналу на контакте SID микропроцессора и используется для последовательного ввода данных;

3.055.430PЭ

W7.5 W6.5 W5.5
биты "ожидание прерывание";

IE
бит "разрешение прерываний". Если IE=1, прерывания разрешены;

M7.5 M6.5 M5.5
маски прерываний. 1 - замаскировано.

Бит IE устанавливается и сбрасывается только командами DI и EI. Изменить маски прерываний и/или сбросить триггер прерывания RST 7.5 может команда SIM. Команда SIM интерпретирует содержимое аккумулятора следующим образом:

S0D	SDE	NU	R7.5	MSE	M7.5	M6.5	M5.5
7	6	5	4	3	2	1	0

- где
- S0D - данные для последовательного вывода. Если SDE=1, то содержимое бита S0D выводится на контакт S0D микропроцессора;
 - SDE - разрешение последовательного вывода. Если SDE=0, бит S0D игнорируется;
 - NU - не используется;
 - R7.5 - сброс триггера прерывания RST 7.5. Если R7.5=1, то запрос на прерывание RST 7.5, если был, отменяется;
 - MSE - разрешение включения маски. Если MSE=0, биты M7.5, M6.5, M5.5 игнорируются и маски прерываний остаются прежними;
 - M7.5 M6.5 M5.5 - маски прерываний. Устанавливаются только при MSE=1. Если маска=1, соответствующее прерывание маскируется.

векторы прерываний

Прерывание	Адрес вектора
RST 5.5	20
RST 6.5	34
RST 7.5	3C

Действия, выполняемые прерываниями RST, аналогичны команде микропроцессора "RST", за исключением того, что

адрес перехода жестко задан.

