

Контроллер клавиатуры на базе однокристалльной микроЭВМ KM1816BE48 обеспечивает двухсторонний обмен кодами команд, букв, знаков и цифр в коде КОИ-7 по последовательному интерфейсу Стык С2

Драйвер графической и алфавитно-цифровой информации для ОС РАФОС микроЭВМ «Электроника 60» расширяет функциональные возможности печатающего устройства УВВПЧ-30—004

Специализированные БИС управления K1806BP1-103 и K1806BP1-157 для схем памяти на цилиндрических магнитных доменах

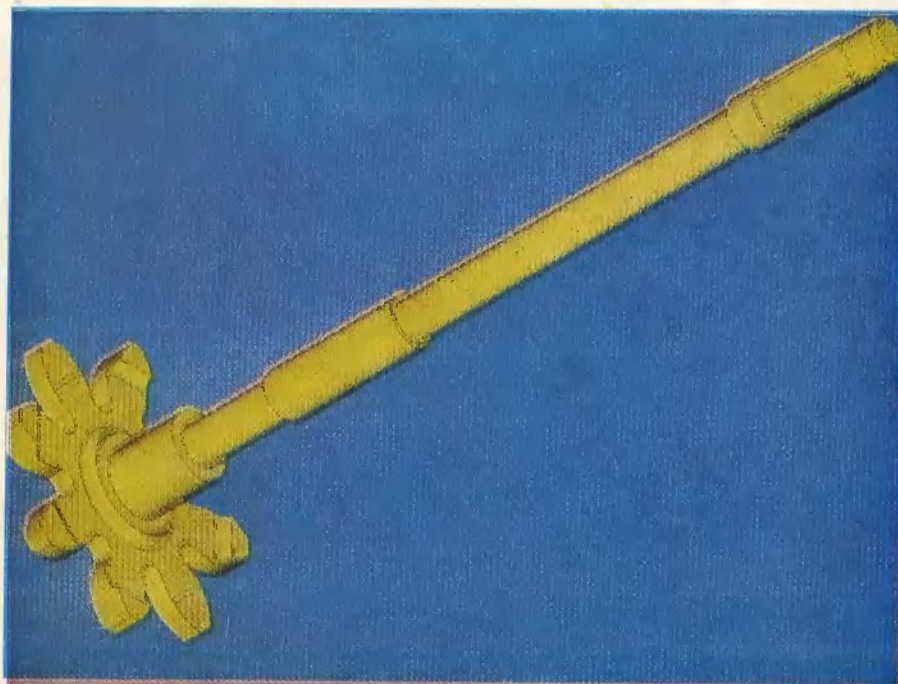
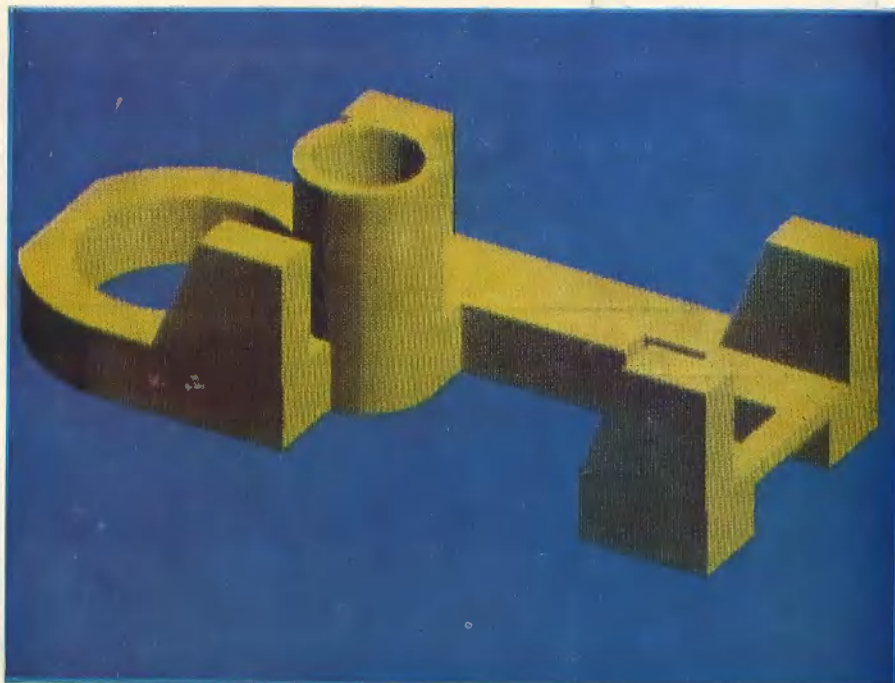
Программное обеспечение персональной ЭВМ «Ириша»: монитор—отладчик, программы стартового ПЗУ

Набор из семи модулей программирования для считывания — записи данных в ПЗУ серий K573, K558, K556, 2700

Стандартизация интерфейсов микропроцессорных систем: обзор информационно-логических, диагностических, структурных, временных, программных и физических характеристик 40 типов наиболее известных интерфейсов.



НА IV ВСЕСОЮЗНОЙ КОНФЕРЕНЦИИ ПО ПРОБЛЕМАМ МАШИННОЙ ГРАФИКИ (г. Протвино, сентябрь, 1987)



На конференции была представлена серия докладов о результатах работ по созданию интерактивного комплекса геометрического моделирования и реалистической визуализации машиностроительных объектов, выполненных авторским коллективом Института автоматизации проектирования АН СССР (Ереззвским М. А., Алексеенко М. М., Свириным Б. Н., Яблонским А. К., Якимом В. В.).

Комплекс предназначен для автоматизации проектирования деталей и узлов промышленных изделий и включает в себя систему геометрического моделирования и визуализации (СИГМа), а также интерактивный машиностроительный язык конструирования (МАЯК).

В СИГМе для достижения интерактивных (порядка 10с) генераций отдельных изображений, реалистично передающих цветовые и геометрические характеристики объекта, реализована параллельная структура базы данных и распараллелены основные алгоритмы визуализации. Это позволило использовать в качестве геометрического процессора присоединенный векторный процессор.

МАЯК представляет собой интерактивную надстройку над СИГМой и обеспечивает диалог в терминах геометрических и конструкторских операций. Средствами МАЯКа определяются фактические параметры примитивов, характеристики их агрегирования, корректируется геометрия конструируемого изделия с учетом топологических связей между компонентами.

Комплекс предназначен для работы на ЭВМ типа СМ-1700 и присоединенном векторном процессоре типа А-12, ИЗОТ-2001С, «Электроника МТ270М», а также ЕС ЭВМ и векторном процессоре ЕС-2706. Общий объем программного обеспечения комплекса составляет 10 тыс. операторов FORTRAN, PL/1 и ассемблера без учета библиотечных процедур векторного процессора.

Телефон для справок: 408-61-88, г. Москва [Свирин Борис Николаевич]

ОРГАН
ГОСУДАРСТВЕННОГО
КОМИТЕТА СССР
ПО ВЫЧИСЛИТЕЛЬНОЙ
ТЕХНИКЕ
И ИНФОРМАТИКЕ
Издается с 1984 года

МП МИКРО ПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ

ВЫХОДИТ ШЕСТЬ РАЗ В ГОД

НАУЧНО-ТЕХНИЧЕСКИЙ И ПРОИЗВОДСТВЕННЫЙ ЖУРНАЛ 6 / 1987 МОСКВА

СОДЕРЖАНИЕ МИКРОПРОЦЕССОРНАЯ ТЕХНИКА

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Отладка

Магистрالی МП-систем Справочная информация

Ершов А. П.— Колонка редактора	2
Шмат В. К.— Специализированные БИС управления для СБИС ЗУ ЦМД	3
Шмат В. К.— Контроллер на основе СБИС ЗУ ЦМД для измерительно-вычислительных комплексов	9
Петропавловский Р. Р., Ходак А. Е.— Драйвер для вывода графической и алфавитно-цифровой информации	15
Попов С. Н.— Программа «Имитатор ДОС» для 8-разрядных ПЭВМ	18
Игнатова И. Г., Преснухин Д. Л.— Выбор режимов адресации при программировании в системе команд микроЭВМ «Электроника 60»	21
Руденко Ю. М., Чернова Т. Ф.— Библиотека стандартных программ для БИС микроЭВМ серии К1816	22
Гостев А. В.— Ускорение считывания с диска в операционной системе CP/M-80	25
Добриневский С. Ф.— Драйвер графопостроителя ЭМ-7042АМ для ОС ДВК	27
Мешков Н. А.— Студенческий научно-производственный отряд математиков-программистов «Прометей»	29
Емелин В. П., Новоселов В. В., Пузанков Д. В.— Комфорт для микропрограммиста	30
Сизов К. А.— Программа обработки текстов с переменной шириной знакоформирующей матрицы	32
Хацкевич Л. Д., Койлис Л. Л.— СУБД для персональной ЭВМ «Электроника 85»	33
Степанов А. Н., Жуков В. А.— Комплекс аппаратно-программных средств отладки микропрограмм для разрядно-модульных микропроцессоров	34
Припоров С. В.— Экранный отладчик для микроЭВМ SM 1800	37
Бровко В. И.— Комплекс для отладки изделий на основе однокристалльной ЭВМ К1816ВЕ51	40
Альперович Е. И., Белый В. Г., Большинский С. И.— Двухмашинный отладочный комплекс для разработки устройств на базе БИС серии КР580	43
Ефремов В. Я.— Сигнатурный анализатор	46
Воевудский В. П., Каштанов В. В., Семенова О. С., Фокин А. А.— Система диагностирования в периферийном процессоре «Электроника МС 1603»	51
Каширин В. М., Понтанк М. С., Рубинштейн Д. И., Смолкин В. С.— Комплекс для отладки систем на базе однокристалльных ЭВМ серии КМ1816	53
Барышников В. Н., Бондарь И. Н., Воронов М. А., Романов В. Ю.— Программы стартового ПЗУ ПЭВМ «Ириша»	56
Барышников В. Н., Бондарь И. Н., Воронов М. А., Кулаков В. Б., Романов В. Ю.— Монитор-отладчик программ для ПЭВМ «Ириша»	61
Кипаренко В. В.— Стандартизация интерфейсов микропроцессорных систем	68
Ачкасов В. Н., Глазнев А. Н., Цыбин С. А., Мещеряков М. Я.— Быстродействующие нетактируемые статические ОЗУ с байтовой организацией КМ581РУ5	86
Матвеев С. С.— Семинар «МП»	90
Рефераты статей	92
Указатель статей, опубликованных в журнале за 1987 год	93

МОДЕМ ДЛЯ ПЭВМ — НЕДОРОГОЙ И НАДЕЖНЫЙ

Первые признаки нового явления проявились на стыке 60-х и 70-х годов по мере распространения систем разделения времени и накопления опыта работы с вычислительными сетями типа Арпанет и Сиклад. Эти системы и сети были созданы для реализации исходной идеи сближения рабочих мест сотрудников с централизованными источниками информационных и вычислительных мощностей. Однако, к тому времени, когда число терминалов перевалило за первую сотню тысяч, при одном из статистических обследований выяснилось, что в каждом пятом случае люди садились за терминал не для того, чтобы поработать за ЭВМ, а только для того, чтобы пообщаться друг с другом, используя систему разделения времени как молчаливого посредника.

Сейчас, когда количество ПЭВМ и терминалов в стране тоже близко к тому, чтобы перевалить за сто тысяч, пора заметить, что среди десятков всевозможных применений вычислительной техники, разнесенной на рабочие места, на одно из первых мест по социальной значимости и по принципиальной несводимости к традиционным применениям ЭВМ стала электронная почта. Судя по письмам в редакцию (одно из наиболее типичных для этой темы писем мы приводим на с. 91), с самых разных исходных точек зрения к этому же, по- существу, выводу приходят многие наши читатели.

Те, кто считает, что ничто не ново под луною, могут сказать, что электронная почта — это всего лишь личный телекс или тривиальная стыковка междугородного автоматического телефона с бытовым компьютером. На деле это «всего лишь» — это всего лишь надводная часть айсберга, подтверждающая, что появление электронной почты является реальностью, подготовленной развитием связной и вычислительной техники. Менее видной — к сожалению, главным образом для нас с вами, дорогой читатель, — частью айсберга являются те революционные изменения в механизме профессионального межличностного взаимодействия, которые происходят под воздействием электронной почты.

Оставим социологам широкие обобщения, поговорим о некоторых важных для нас частностях. Электронная почта позволяет синхронизировать работу небольших коллективов, не собирая их членов физически в одно место, не затрачивая времени на переписку и, главное, сохраняя темпы и непрерывность творческой активности каждого из участников. Идея, пришедшая вам в голову вечером, становится утром общим достоянием ваших коллег от Калининграда до Владивостока. В отличие от телефонного разговора электронная почта, сохраняя его спонтанность, автоматически обеспечивает полную документируемость информационного обмена. Азарт непринужденной и преходящей дискуссии сочетается с церемонностью нотариуса и скрупулезностью архивариуса. В результате темп и продуктивность интеллектуальной работы возрастает на порядок.

Что мы имеем на сегодня? Порядка десяти приличных моделей ПЭВМ. Очень далекую от насыщения, но быстро развивающуюся систему междугородной телефонной связи. У нас нет модема — небольшой коробочки, соединяющей ПЭВМ с телефонным каналом. В наших условиях его создать непросто. Он должен стоить не дороже телефонного аппарата и преодолевать невысокое качество коммутируемых каналов. Единственным компенсирующим послаблением является допущение невысокой скорости передачи: даже сотня бит/с для начала будет приемлемой.

Создание массового модема для персональных ЭВМ — благодатнейшая задача для специалистов. Работы хватит всем. Электронщикам надо разработать и наладить производство серии СВНСов, которые вберут в себя львиную долю проблем компактности и надежности. Схемотехники должны решить проблему интерфейса между собственно связной и компьютерной частями схем, выделить программно-управляемую часть модема. Разработчики ПЭВМ должны предусмотреть стандартный интерфейс с модемами, если необходимо, дополнительные платы управления. Связистам надо выработать оптимальные методы модуляции — демодуляции, определить меру компромисса между скоростью и надежностью, обеспечить стыковку модема с обычными телефонами. Сетевики должны определить наиболее подходящие сетевые протоколы для установления связи и контроля над ней. Программистам предстоит разработать врезки в стандартные операционные системы и создать нужный набор вспомогательных программ, поддерживающих основные виды работ: собственно электронная почта, «доска объявлений», телеконференции, запросы к публичным информационным системам.

Не надо думать, что модемы для терминалов и вычислительных сетей не разрабатываются. Их производство, однако, носит малосерийный характер в интересах специализированных систем связи, а сами конструкции либо дороги, либо громоздки, либо и то и другое вместе. В этом плане задача создания массового модема перекликается с задачей создания массовой, в частности, школьной ПЭВМ, о которой мы говорили на страницах нашего журнала (см. например, серию статей по ПЭВМ «Ириша», завершаемую в данном выпуске).

Аналогичным образом мы хотели бы поддержать в журнале все интересные технические и конструктивные находки на пути решения важнейшей и глобальной задачи — интеграции связного и вычислительного оборудования.

А. Ершов

Главный редактор

А. П. ЕРШОВ

Редакционная коллегия

А. Г. Алексенко

В. М. Брябрин

А. А. Васенков

[зам. главного редактора]

И. Я. Вельбицкий

А. Б. Венгеров

Г. Р. Громов

[ответственный секретарь]

В. П. Иванников

М. Б. Игнатьев

А. В. Каляев

И. З. Карась

В. П. Куприянов

С. С. Лавров

В. В. Липаев

К. А. Меликян, И. А. Мизин

Б. Н. Наумов

[зам. главного редактора]

С. М. Пеленов

[зам. главного редактора]

А. К. Платонов

А. А. Попов

Д. А. Распелов

Б. И. Рамеев

О. Л. Смирнов

А. А. Стогний

М. К. Сулим

Н. М. Шаруненко

Редакционный совет:

Р. Л. Ашастин

И. В. Бабынин

С. Н. Бушев

Е. П. Велюхов

Н. Н. Говорун

В. В. Корчагин

В. П. Макаревич

И. И. Малащенин

Ю. Е. Нестерихин

А. Р. Назарьян

А. Л. Нефедкин

И. В. Прангишвили

Л. Н. Преснухин

В. В. Пржилаковский

Н. Л. Прохоров, Г. Г. Рябов

К. Н. Трофимов

В. И. Хохлов

Н. Н. Шереметьевский

Е. В. Шильдин, А. В. Яковлев

Э. А. Якубайтис

Номер подготовили:

Е. И. Бабич, Г. Г. Глушкова,

В. М. Ларионова, С. С. Матвеев

Фото О. В. Чиркина

Адрес редакции журнала:

103051, Москва,

Малый Сухаревский пер., д. 9А

Телефоны: 208-73-23, 208-19-94.

Сдано в набор 08.11.87. Т-19095.

Подписано к печати 23.12.87.

Формат 84×108^{1/2}. Бумага № 1.

Высокая печать. Усл. печ. л. 10,08.

Уч.-изд. л. 14,8. Тираж 90 276 экз.

Заказ 296. Цена 1 р. 10 к.

Орган Государственного комитета

СССР по вычислительной технике

и информатике

Московская типография № 13

ПО «Периодика» ВО «Союзполи-

графпром» Госкомиздата СССР

107005. Москва. Денисовский пер.,

дом 30

На первой странице обложки —

Машинная графика.

УДК 681.323

В. К. Шмат

СПЕЦИАЛИЗИРОВАННЫЕ БИС УПРАВЛЕНИЯ ДЛЯ СБИС ЗУ ЦМД

Применение СБИС ЗУ ЦМД связано с целым рядом особенностей, обусловленных спецификой архитектуры СБИС и множеством функциональных сигналов управления памяти нового типа. Факторами, не имеющими аналогов в ЗУПВ на кремниевых структурах и традиционных носителях информации внешних запоминающих устройств, являются следующие:

большое число управляющих сигналов, составляющих временную диаграмму функционирования СБИС ЗУ ЦМД;

адресация и идентификация данных, так как ЦМД-память не содержит в своей структуре адресов;

алгоритмы доступа к данным; операции информационного обмена, не нарушающие выбранного алгоритма адресации.

Для того чтобы учесть эти факторы, приходится разрабатывать сложную аппаратуру управления и управляющие программы для вычислительных сред. При этом энергетические и массогабаритные характеристики устройств управления получаются сопоставимыми, а в отдельных случаях и превосходят соответствующие характеристики самой ЦМД-памяти. Эффективной альтернативой этому направлению является применение специализированных БИС управления ЗУ ЦМД. Оптимальным вариантом для пользователя можно считать представление СБИС ЗУ ЦМД как «аналога» ЗУПВ, работающего по общепринятому протоколу обмена: адрес → операция «чтение» или «запись». С учетом всех перечисленных факторов разработаны БИС управления К1806ВП1-103, К1806ВП1-157 для микросхем ЗУ ЦМД К1602РЦ2 (К1605РЦ1). Микросхемы памяти, работающие под управлением этих БИС, представлены как ЗУПВ с виртуальными адресами 0, 1, 2, ..., 1024, причем с каждым адресом связан блок 282×n бит информации (страница), где n — число параллельно работающих СБИС ЗУ ЦМД. Для выполнения операции достаточно указать виртуальный адрес и тип операции (поиск, запись, чтение). Информационный обмен производится одиночными блоками или блоками необходимой длины, кратными 282×n бит. При выполнении информационных операций автоматически ведется

поиск физического адреса страницы начала обмена, связанного с виртуальным адресом. Преобразование адресов происходит непосредственно в цикле поиска и не требует дополнительных временных затрат.

Основные характеристики систем памяти на СБИС ЗУ ЦМД типа К1602РЦ2 при этом следующие: среднее время доступа к произвольной странице — 5,12 мс при частоте поля продвижения доменов (ВМП), равной 100 кГц; время обслуживания информационного блока составляет 5,12 мс с учетом времени обмена на данными, равного 2,82 мс; время доступа к последовательным виртуальным адресам равно 0.

Микросхема К1806ВП1-103 (рис. 1) имеет свободный интерфейс, что дает возможность использовать ее в аппаратуре различной архитектуры. Назначение выходов БИС приведено в табл. 1. Микросхема служит для

формирования временной последовательности сигналов управления для СБИС ЗУ ЦМД типа К1602РЦ2, К1605РЦ1, организации стартового режима работы СБИС ЗУ ЦМД, выполнения операции поиска, записи/чтения информации в (из) СБИС ЗУ ЦМД. По функциональному назначению выводы микросхемы разделяются на 4 группы:

интерфейс пользователя: выходы \overline{SR} , \overline{ST} , \overline{STOP} , \overline{WR} , \overline{RD} , \overline{WRCT} , \overline{WRRG} , \overline{INC} , \overline{EB} совместно с двуправленной шиной данных $\overline{DB1} \dots \overline{DB11}$;

интерфейс управления СБИС ЗУ ЦМД: выходы \overline{FA} , \overline{FB} , \overline{FC} , \overline{FD} , \overline{COIL} , \overline{PROM} , \overline{ROC} , \overline{ROS} , \overline{GEN} , \overline{TO} , \overline{TI} , \overline{CDR} , \overline{ODD} ;

флажки состояний: \overline{FL} и \overline{ER} ; внутренний интерфейс комплекта

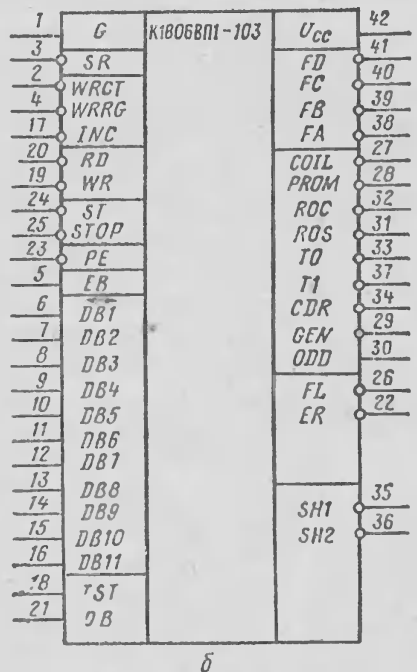
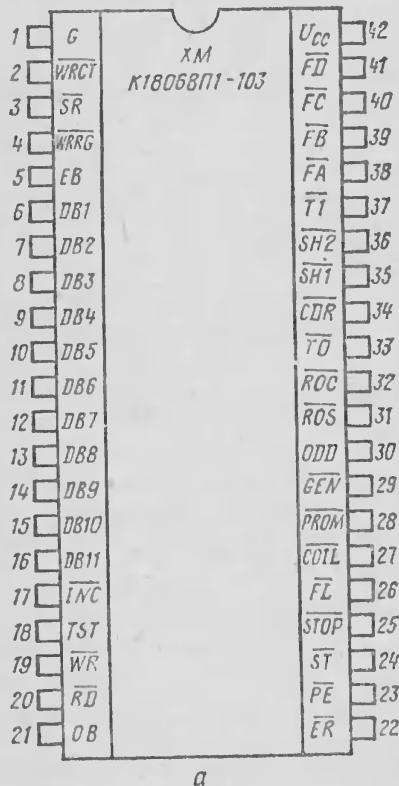


Рис. 1. Условное графическое изображение микросхемы К1806ВП1-103: а) по порядку расположения выводов; б) по функциональному расположению выводов

Таблица 1
Назначение выводов микросхем
K1806ВП1-103

Вывод	Обозначение	Назначение
1	G	Внешний генератор
2	WRCT	Запись текущего адреса
3	SR	Начальная установка
4	WRRG	Запись адреса поиска
5	EB	Управление шиной
6...16	DB1...DB11	Входы-выходы разрядов шины данных
17	INC	Инкремент текущего адреса
18	TST	Контроль функционирования
19	WR	Запись
20	RD	Чтение
21	OB	Общий вывод
22	ER	Ошибка
23	PE	Авария
24	ST	Старт
25	STOP	Стоп
26	FL	Выход сигнала «завершено»
27	COIL	Выход сигнала ВМП
28	PROM	Фаза данных
29	GEN	Генератор
30	ODD	Выход сигнала «чет/нечет»
31	ROS	Репликатор 2
32	ROC	Репликатор 1
33	TO	Переключатель вывода
34	CDR	Строб чтения
35	SH1	Сдвиг 1
36	SH2	Сдвиг 2
37	TI	Переключатель ввода
38	FA	Фаза А
39	EB	Фаза В
40	FC	Фаза С
41	FD	Фаза D
42	U _{cc}	Напряжение питания

БИС: FL, SH1, SH2. Вывод TST необходимо подключать к шине нулевого потенциала. Для работы микросхемы на вход G подается сигнал от внешнего тактового генератора с частотой 3,2 МГц для обеспечения частоты ВМП, равной 100 кГц.

Пользователь имеет доступ к двум внутренним 11-разрядным регистрам микросхемы по шине DB: регистру текущего адреса (PTA) и регистру адреса поиска (PAП). Последний доступен только для записи. При высоком уровне на входе EB шина DB работает на прием данных, используя сигналы управления WRCT и WRRG (рис. 2). При EB = «Лог. 0» на шину DB будет транслироваться содержимое PTA — текущего виртуального адреса страницы для каждого такта ВМП. Используя управляющий вход INC, можно инкрементировать содержимое PTA в соответствии с реализованным в микросхеме алгоритмом виртуальной адресации: адрес (n+1) = адрес n+1 (mod 1025). Для указания типа операций, выполняемых СБИС ЗУ ЦМД, используются входы WR и RD (табл. 2). Необходимая комбинация сигналов WR и RD не должна изменяться в течение всего цикла операции.

Начальная установка микросхемы при включении питания приводит ее в детерминированное состояние. Содержимое PTA и PAП при этом ста-

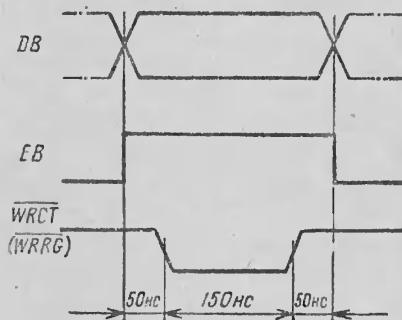


Рис. 2. Временная диаграмма записи регистров PTA и PAП

Таблица 2
Операции контроллера

WR	RD	Операция
1	1	Поиск
0	1	Запись
1	0	Чтение
1	1	Недопустимая операция, устанавливается флаг ошибки ER=0

вится равным 0. Выполнение операций инициируется выдачей импульсного сигнала ST (длительности всех импульсных сигналов ST, STOP, WRCT, WRRG, INC не менее 150 нс). Операции прекращаются с приходом импульсного сигнала STOP по следующему алгоритму:

по ближайшей фазе FD при выполнении «чистой» операции поиск (WR=RD=1);

по фазе FD 512-го такта текущей страницы (WR и RD в соответствии с табл. 2). Для своего завершения чистая операция поиска не требует выдачи сигнала STOP и прекращается при содержимом PTA равным PAП, с одновременной выдачей импульсного сигнала.

В ходе информационных операций, операция поиска скрыта от пользователя и всегда происходит автоматически, если PTA ≠ PAП. Если операция многоблочная, то она будет выполняться одновременно с поиском начальной страницы обмена, так как информационные страницы расположены последовательно в соответствии с виртуальными адресами: страница n, страница n+1...

Управляющий вход PE предназначен для обработки аварийных ситуаций реального времени (авария источника питания системы или СБИС ЗУ ЦМД, сбой источника питания, аварийный останов операции) по следующему алгоритму для обеспечения сохранности информации в ЦМД-памяти:

при активном сигнале PE=0 микросхема не запускает на выполнение никакую операцию;

если сигнал PE=0 выработан во время выполнения операции поиска, то выполнение операции прекращается с корректной остановкой поля ВМП по ближайшей фазе FD с установкой флага ошибки ER=0;

если сигнал PE становится активным во время выполнения информационной операции, то микросхема начинает анализировать его длительность: при длительности сигнала больше 10 мкс (для тактовой частоты 3,2 МГц) операция прекращается в соответствии с предыдущим условием; при длительности сигнала меньше 10 мкс операция прекращается по завершении работы с текущей страницей и выдается флаг ошибки.

Если вход PE используется для аварийного останова, то длительность PE=0 должна быть не менее 10 мкс, если не используется, то его необходимо подключить к шине единичного потенциала, например к U_{cc} = 5 В.

Содержимое PTA и PAП проверяется на допустимое значение (≤1024) виртуального адреса. Если

их значение больше допустимого (недействительный адрес), то устанавливается флаг ошибки $ER=0$ и запрещается выполнение любой операции. При устранении причины, вызвавшей установку флага ошибки (недопустимая операция, недопустимое значение виртуального адреса, активный сигнал PE), происходит сброс флага ошибки: $ER=1$.

Выполнение информационных операций всегда сопровождается установкой флага $FI=0$ в конце блока, как сигнала об окончании работы с блоком (страницей) информации. Флаги FL и ER могут использоваться для организации двухуровневой системы прерываний ведущей системы.

Функциональные сигналы управления (интерфейс СБИС ЗУ ЦМД) приведены в табл. 3. Временные диаграммы функциональных сигналов — на рис. 3. В соответствии с реализованным алгоритмом страничной адресации страницы с последовательными виртуальными адресами расположены через 512 тактов ВМП.

Временная диаграмма работы микросхемы в режиме поиска приведена

Таблица 3
Функциональные сигналы управления СБИС ЗУ ЦМД

Выход	Функциональный сигнал
COIL	Сигнал разрешения работы с памятью. Удерживается в состоянии «Лог. 0» при работе формирователей токов продвижения доменов
PROM	Указывает на информационную часть COILA: 282 такта ВМП при чтении, 283 такта ВМП при записи
FA, FB, FC, FD	Сигналы для управления формирователями токов продвижения
GEN	Разрешение работы генератора (только при записи)
TO	Переключатель вывода
TI	Переключатель ввода
ROC, ROS	Сигналы для репликации доменов (при чтении)
CDR	Строб для усилителей считывания
ODD	Сигнал разделения каналов считывания

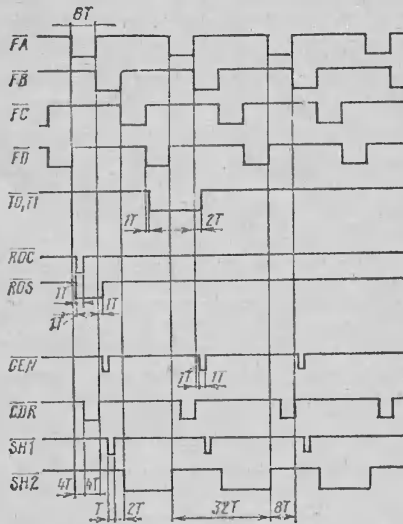


Рис. 3. Временная диаграмма функциональных сигналов управления СБИС ЗУ ЦМД

на рис. 4. В РТА содержится значение адреса, равное 2, в РАП — равное 1021. По сигналу ST начинается формирование сигнала $COIL$ и фаз FA, FB, FC, FD . Цикл поиска равен трем тактам ВМП СБИС ЗУ ЦМД. Во время первых 2/3 длительности фазы FA и FB, FC, FD на шину данных DB ($EB=0$) транслируются действительные значения текущего адреса. При достижении условия равенства текущего адреса и адреса поиска происходит останов ВМП по фазе FD и установка «Лог. 0» на выходе FL .

Временная диаграмма операции записи приведена на рис. 5. Операция начинается после поступления импульсного сигнала ST . Значения РТА и РАП равны (отсутствует операция

поиска при информационной операции). Так как на входе WR поддерживается значение 0, то в 1 и 2 тактах ВМП будет выработан сигнал TO (переключатель вывода). При достижении 209 такта на выходе $PROM$ установится значение низкого уровня, которое будет удерживаться до 492. Сигнал $PROM=0$ разрешает формирование сигналов $GEN, SH1$ и $SH2$. В тактах 510 и 511 ВМП будет выработан сигнал «переключатель ввода» TI . Импульсный сигнал $STOP$ запоминается микросхемой и обеспечивает останов ВМП снятием сигнала $COIL$ по концу FD 512 такта. При этом в фазе FD 512 такта установится значение «Лог. 0» на выходе FL сигнализирующее об окончании цикла работы с ЗУ ЦМД. После окончания операции в РТА будет находиться адрес следующей страницы СБИС ЗУ ЦМД. По сигналу FL допускается снятие сигнала WR или RD , если операция завершена.

Временная диаграмма операции чтения двух страниц, начиная с адреса 1024, дана на рис. 6; текущее значение РТА=1.

Сначала (по сигналу ST) выполняется операция поиска, равная одному такту ВМП, затем по условию $РТА=РАП$ начинается цикл чтения.

Во втором такте будут выработаны сигналы ROC и ROS . В начале 86-го такта ВМП установится сигнал $PROM$, разрешающий формирование сигналов $CDR, ODD, SH1, SH2$. Сигнал $PROM$ удерживается 282 такта ВМП и сбрасывается по началу 369-го такта. Во время 512-го такта чтения страницы 1024 в фазе FD установится сигнал FL (конец страницы). Во время чтения следующей

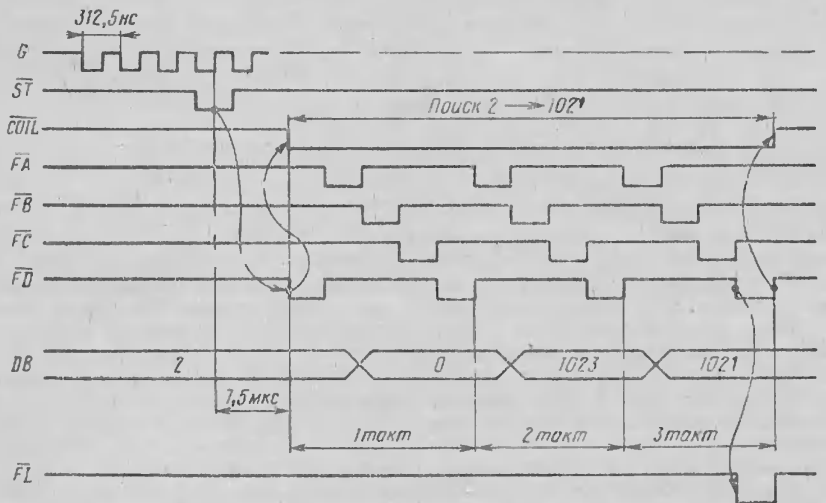


Рис. 4. Временная диаграмма операции поиска

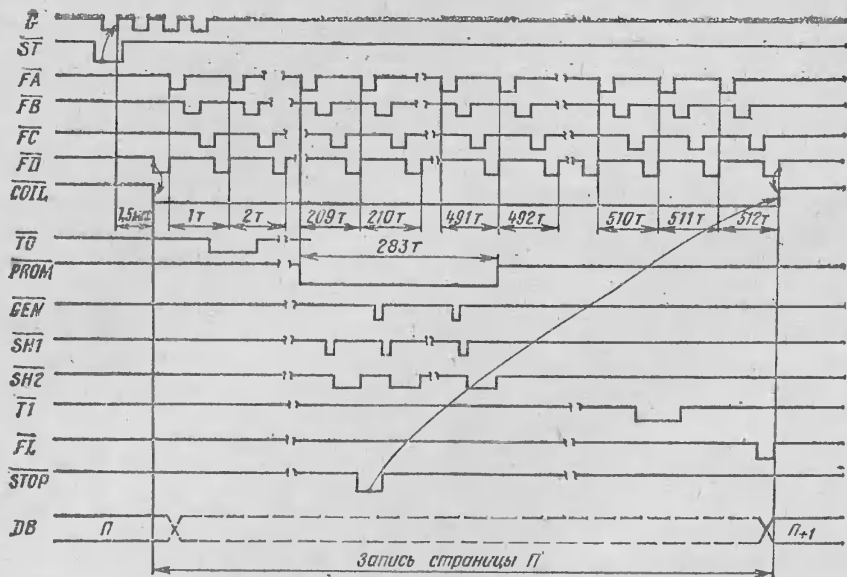
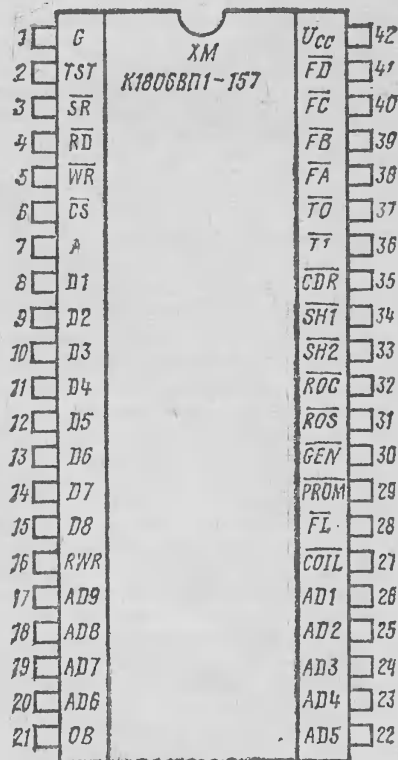


Рис. 5. Временная диаграмма операции запись одной страницы



а

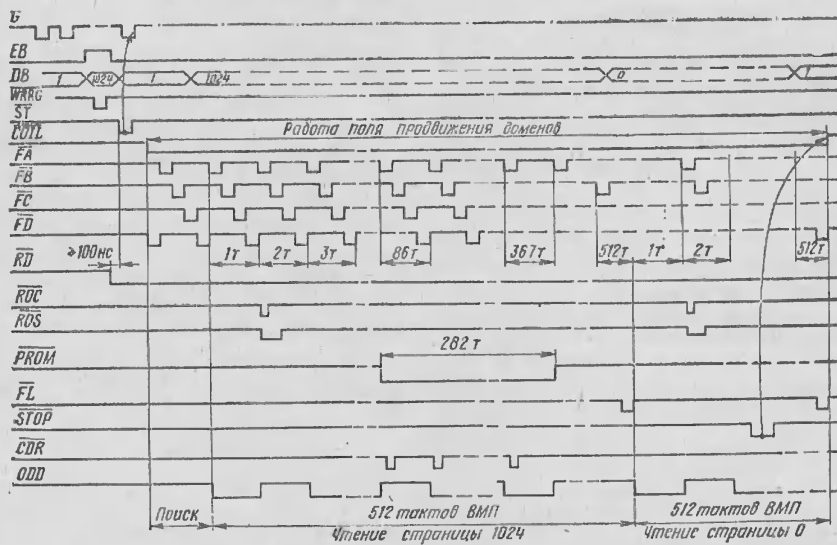
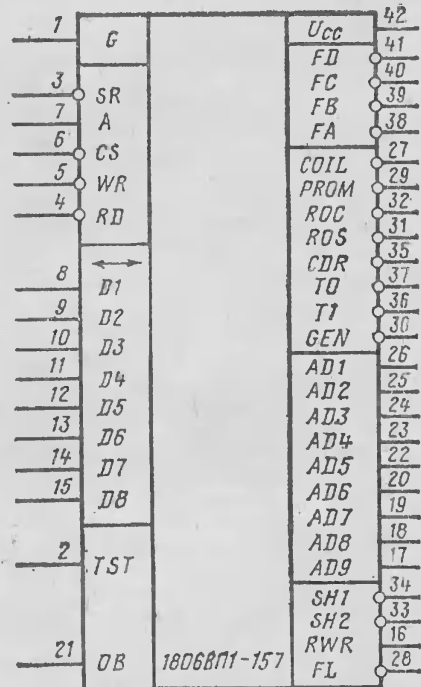


Рис. 6. Временная диаграмма операции чтение блочное



б

Рис. 7. Условное графическое изображение микросхемы K1806BP1-157: а) по порядку расположения выводов; б) по функциональному расположению выводов

нулевой страницы выдается сигнал STOP. По завершении чтения нулевой страницы происходит останов ВМП в конце фазы FD 512-го цикла. Значение РТА будет равно 1.

Микросхема K1806BP1-157 имеет байтовую шину и может использоваться в составе микропроцессорных систем с адресацией, соответствующей порту ввода-вывода или ячейкам памяти в адресном пространстве микропроцессора. Условное обозначение микросхемы приведено на рис. 7. Функциональные сигналы управления, алгоритмы страничной адре-

сация соответствуют микросхеме K1806BP1-103.

Группа управляющих выводов (A, CS, WR, RD) вместе с двунаправленной шиной данных DB1...DB8 образует байтовый интерфейс пользователя. В отличие от микросхемы K1806BP1-103 дополнительно реализован узел формирования адресов ПЗУ (ОЗУ) карты годности СБИС ЗУ ЦМД — выводы AD1...AD9. Вывод RWR транслирует тип опознанной операции — «запись» или «чтение»; при операции «запись» RWR = 0.

Таблица 4

Косвенная адресация регистрового файла (A=1)

Содержимое адресного регистра РК		Адресуемый регистр РФ
бит 07	бит 06	
0	0	РТА (LSB — младший байт)
0	1	РТА (MSB — старший байт)
1	0	РАП — LSB } при за-
1	1	РАП — MSB } писи

СВИС ЗУ ЦМД; АПЗУ — формирователь адресов карты годности при операциях запись/чтение.

Одновременно с начальной установкой микросхемы (сигнал SR=0) производится сброс содержимого регистрового файла. Связь с микросхемой возможна при активном сигнале CS=0. Выводы RD и WR используются для выполнения стандартных процедур записи/чтения РФ. Адресная шина A адресует регистр команд или регистровый файл, доступный пользователю. Регистровый файл кроме РК содержит РТА (2 байта) и РАП (2 байта). Форматы регистров

Таблица 5

Система команд микросхемы K1806BP1-157

Команда	Тип операции	Код команды (восьмеричный)
Системный сброс	Начальная установка микросхемы (эквивалент SR)	000
Стоп	Прекратить операцию: поиск по ближайшей фазе; информационная операция (по завершении работы с текущей страницей)	200
Поиск	Поиск страницы, заданной в РАП. Если РТА=РАП, операции не инициализируются	001
Чтение блочное	Чтение п. блоков, начиная со страницы, указанной в РАП	005
Запись блочная	Требуется команда «стоп»	003
Чтение страницы	Запись п. блоков, начиная со страницы, указанной в РАП	025
Запись страницы	Чтение одной страницы, указанной в РАП	023
Чтение блочное с текущего адреса	Запись п. блоков, начиная с адреса, указанного в РТА	015
Запись блочная с текущего адреса	Требуется команда «стоп»	013
Чтение текущей страницы	Чтение одной страницы с адреса, указанного в РТА	035
Запись текущей страницы	Запись одной страницы с адреса, указанного в РТА	033

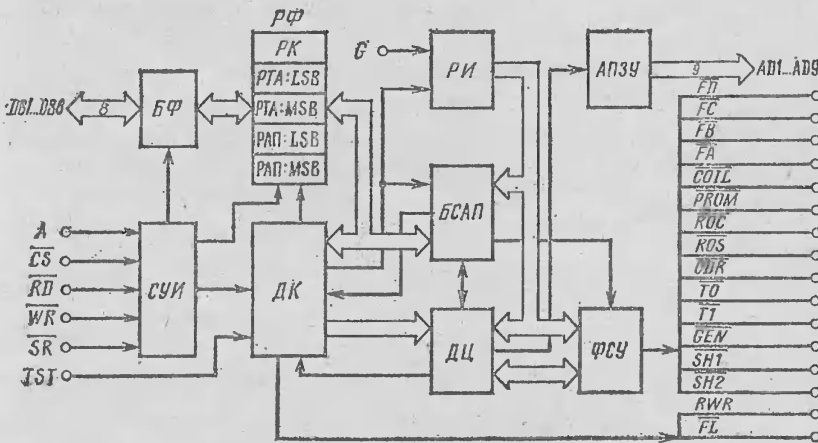


Рис. 8. Структурная схема микросхемы K1806BP1-157

Микросхема содержит следующие функциональные блоки (рис. 8):

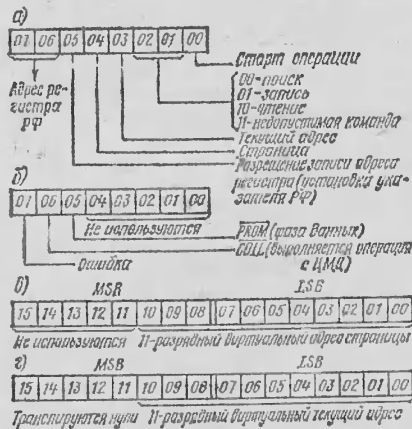


Рис. 9. Форматы регистров микросхемы K1806BP1-157:

а) регистр команд в режиме записи; б) регистр команд в режиме чтения; в) РТА и РАП в режиме записи; г) РТА в режиме чтения

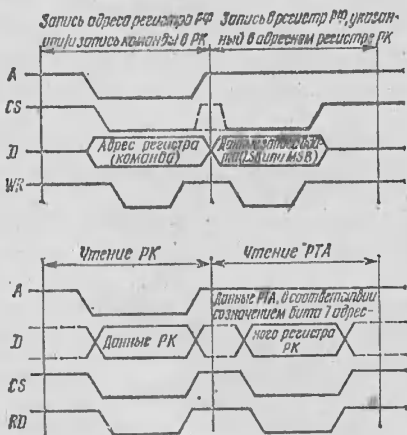


Рис. 10. Временные диаграммы записи и чтения регистрового файла микросхемы K1806BP1-157

показаны на рис. 9. При $A=0$ адресуется РК. При $A=1$ адресуется один из четырех байтов РФ (табл. 4), обозначенный в указателе адреса РК (косвенная адресация через РК). РАП доступен только для записи. Временные диаграммы записи/чтения РФ показаны на рис. 10. Схема управления интерфейсом дешифрует сигналы управления интерфейсом (A, RD, WR) и через буфер БФ производит запись/чтение РФ.

Запись команды в РК инициирует выполнение операции, заданной кодом команды. Дешифратор команд анализирует содержимое РК, разрешает работу распределителя импульсов, начинает выполнение операции.

Алгоритм страничной адресации СБИС ЗУ ЦМД реализуется в БСАП, который выполняет при необходимости операцию поиска нужной страницы, заданной в РАП, и разрешает работу ДЦ при выполнении информационных операций. На выходе RWR устанавливается тип опознанной выполняемой операции со СБИС ЗУ ЦМД. При выполнении операции запись $RWR=0$.

Дешифратор циклов и РИ разрешают ФСУ выработку сигналов управления ЦМД. Формирователь адресов АПЗУ генерирует адреса ПЗУ (ОЗУ) для считывания карты годности (данных о дефектных регистрах СБИС ЗУ ЦМД). Сигнал $COIL=0$ указывает на цикл работы со СБИС ЗУ ЦМД. Для управления формирователями токов продвижения доменов (ВМП) при $COIL=0$ формируются четыре серии импульсов: FA, FB, FC, FD (см. рис. 3).

Сигнал $PROM=0$ указывает на выполнение информационного обмена со СБИС ЗУ ЦМД. В зависимости от типа информационной операции будут вырабатываться сигналы репликатора ROC, ROS , строба усилителя чтения CDR в режиме чтения или TO, TI и генератора GEN в режиме записи соответственно. По окончании операции вырабатывается сигнал $FL=0$, который может быть использован для прерывания ведущей системы, например микропроцессора. При многостраничных операциях этот сигнал вырабатывается по завершении операции с каждой текущей страницей СБИС ЗУ ЦМД. Выполнение многостраничных операций требует записи команды «стоп» в РК. Анализируя содержимое РК, можно определить фазу текущей операции, проверить бит ошибки, а по состоянию РТА — следить за текущими адресами страниц. Вывод младшего разряда AD1 используется для разделения каналов усилителя считывания (чет/нечет).

В отличие от БИС K1806ВП1-103 в микросхеме K1806ВП1-157 реализован набор команд, позволяющий уп-

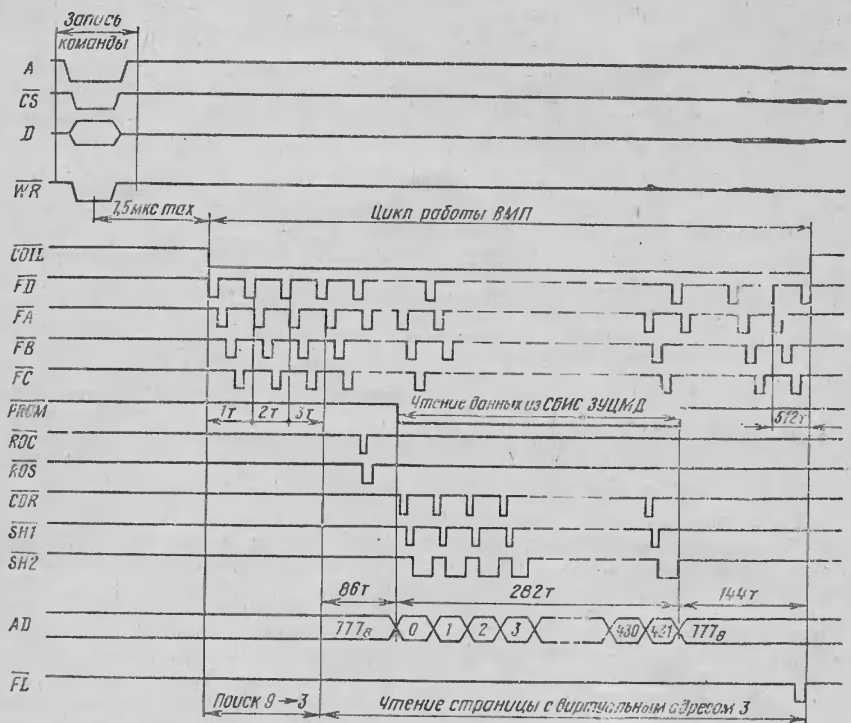


Рис. 11. Временная диаграмма операции чтение страницы

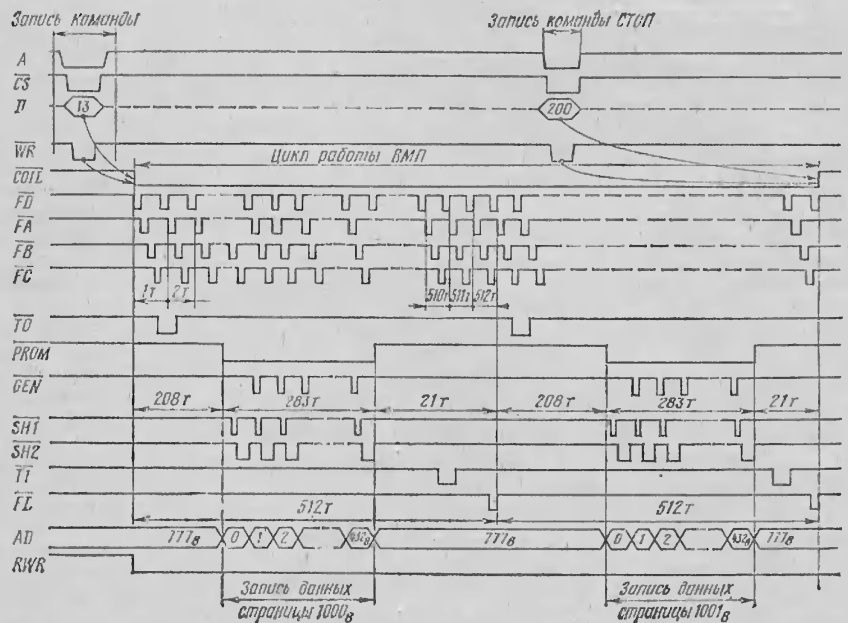


Рис. 12 Временная диаграмма операции запись блочная с текущего адреса

ростить аппаратуру и расширить возможности контроллера устройств управления системами памяти, особенно устройств с ограниченными программными и аппаратными возможностями.

Система команд, описание типов

операций и коды команд приведены в табл. 5. Для выполнения операций со СБИС ЗУ ЦМД необходимо с помощью соответствующего кода указать адрес страницы в РАП (кроме операций с текущим адресом) и тип операции. При этом автоматически

выполняются операции преобразования виртуальных адресов в физические и осуществляется алгоритм поиска. Для начала работы СБИС ЗУ ЦМД в РТА необходимо занести адрес, соответствующий действительному адресу текущей страницы. Задавать чистую операцию поиска в процессе выполнения информационных

операций не обязательно, так как она является их составной частью.

Информационные операции делятся на одно- и многостраничные. Содержимое РТА и РАП анализируется на допустимое значение адреса (≤ 1024). При превышении этого значения инициация операции запрещается ДК и в РК устанавливается бит ошибки.

Временные диаграммы операций «поиск», «запись», «чтение» приведены на рис. 11 и 12. Микросхемы выполнены по КМОП-технологии, требуют для своей работы один источник питания $U_{cc} = 5 \text{ В} \pm 10\%$, совместимы с ТТЛ-логикой и характеризуются низким энергопотреблением — $P_{ст} \leq 6 \text{ мВт}$.

Статья поступила 21 мая 1987 г.

УДК 681.142

В. К. Шмат

КОНТРОЛЛЕР НА ОСНОВЕ СБИС ЗУ ЦМД ДЛЯ ИЗМЕРИТЕЛЬНО-ВЫЧИСЛИТЕЛЬНЫХ КОМПЛЕКСОВ

Применение внешних запоминающих устройств (ВЗУ) на основе СБИС ЗУ ЦМД в микропроцессорных измерительно-вычислительных комплексах вместо ИГМД позволяет значительно повысить надежность работы аппаратуры, особенно в экстремальных условиях: вибрации, повышенное содержание влаги и пыли и т. д.

Усилия разработчиков контроллеров таких ВЗУ в последнее время направлены на эмуляцию традиционных ВЗУ типа ИГМД, что обеспечивает совместимость на уровне аппаратного драйвера устройства, но не позволяет реализовать все преимущества памяти нового типа. Рассматриваемый ниже достаточно простой контроллер поддерживается драй-

вером, допускающим размещение в ВЗУ на СБИС ЗУ ЦМД операционных систем РАФОС, ОС ДВК и не повторяет аппаратную эмуляцию традиционных ВЗУ.

Структура системы памяти на СБИС ЗУ ЦМД приведена на рис. 1. Архитектуру системы и управление обеспечивает контроллер. Сопряжение с каналом ЭВМ осуществляется через стандартное устройство параллельного обмена типа И1 или И2. Система памяти включает 1...8 модулей. Каждый модуль памяти содержит 1...8 СБИС ЗУ ЦМД типа К1602РЦ2 или К1605РЦ1 емкостью 256 кбит каждая. Суммарная емкость памяти достигает 2М байт. Внутренний интерфейс ВЗУ образуют шины управления и адреса модулей, дву-

направленная байтовая шина данных и шина данных карты годности. В каждый момент времени контроллер работает с одним модулем памяти, выбираемым по соответствующему адресу. Наличие модуля с данным адресом подтверждается сигналом ответа на шине УВ («устройство выбрано»), реализованной по принципу «монтажное ИЛИ». По шине данных карты годности контроллер имеет доступ к ПЗУ модуля, где хранится информация о негодных (бракованных) регистрах микросхем памяти данного модуля. Управление работой модулей осуществляется 15 сигналами с шины управления. Информационное поле памяти всех модулей памяти можно рассматривать как одно устройство суммарной емкости или как независимые устройства файловой структуры, имеющие собственные каталоги и авторизованный доступ. Необходимую конфигурацию поддерживает соответствующий драйвер. Допускается применение модулей различной емкости: с одной, двумя, четырьмя микросхемами памяти. Разрешается работа контроллера под управлением нескольких драйверов. Следовательно, контроллер дает возможность максимально использовать имеющуюся память в зависимости от требований к системе: от 32К байт (одна микросхема) до 2М байт (64 микросхемы).

Ядром контроллера (рис. 2) является БИС управления К1806ВП1-103 D25, формирующая временную диаграмму работы СБИС ЗУ ЦМД и сигналы синхронизации для работы блока форматора данных (сигналы ШН1, ШН2), а также реализующая алгоритмы страничной адресации и поиска. Запись/чтение внутренних регистров микросхемы D25 (регистра текущего адреса РТА и адреса поиска РАП) производится через буферные микросхемы D18...D20 К589АП16. Синхронизация устройства производится тактовым генератором, реализованным на микросхемах D23, D24 и кварцевом резонаторе ВР1 на 6,4 МГц. Можно выделить следующие функциональные узлы контроллера:

— форматор данных, состоящий из сдвиговых регистров D31...D33 типа К155ИР13, счетчика битов D21 К555ИЕ7, триггера управления D24.1 К555ТМ2, комбинационных схем ло-

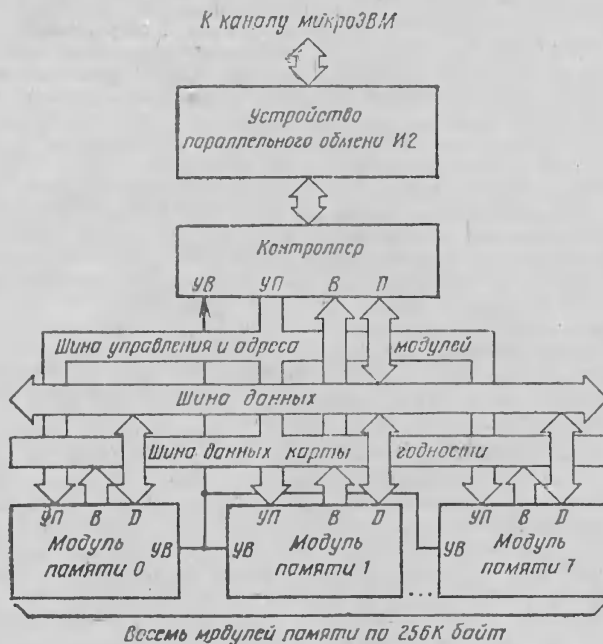


Рис. 1. Структура системы памяти для ВЗУ на СБИС ЗУ ЦМД объемом до 2М байт

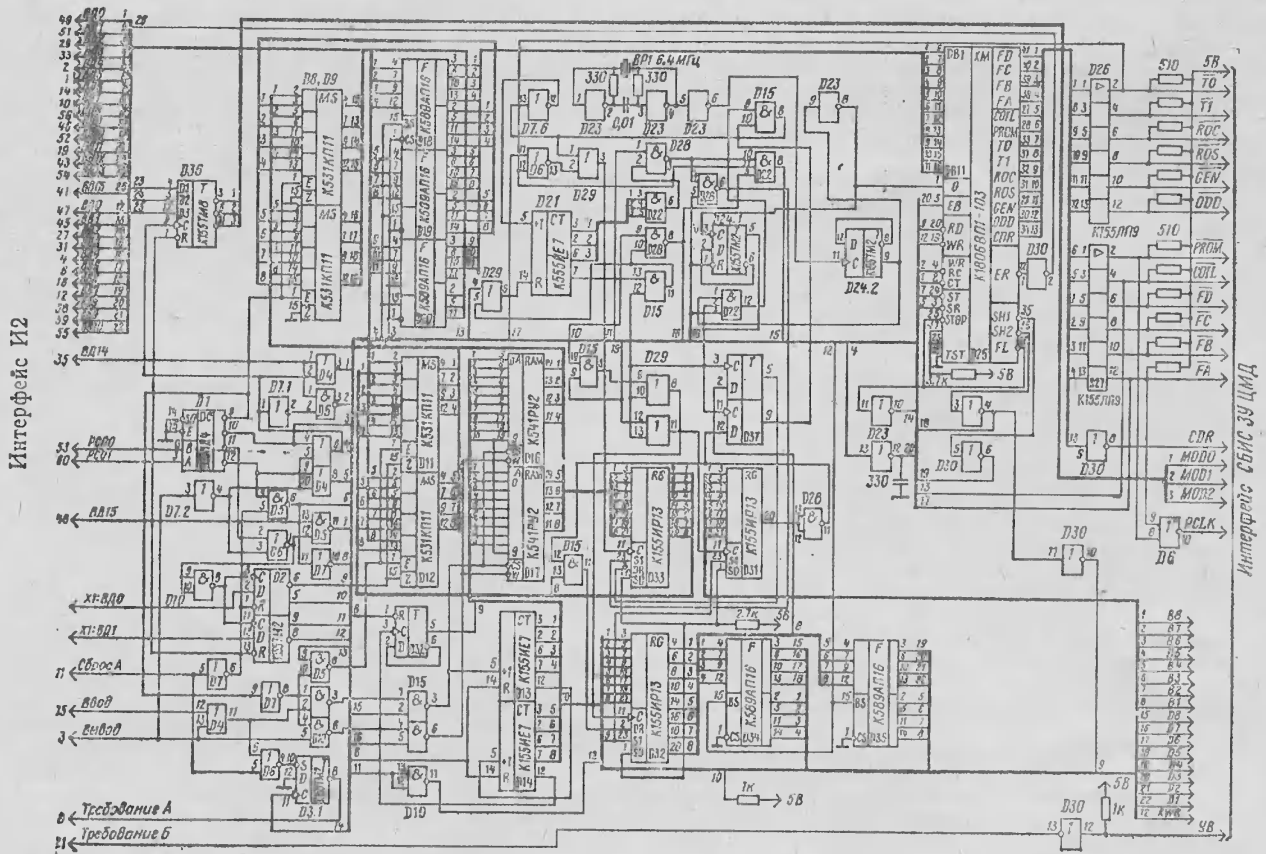


Рис. 2. Принципиальная электрическая схема контроллера

гики и регистра расширения данных на триггере D37;

— блок буферного ОЗУ D16, D17 емкостью до 1К байта на микросхемах K541PУ2 со счетчиком адреса D13, D14 на микросхемах K555IE7 и триггере D3 K555TM2;

— дешифратор режима работы контроллера на микросхеме K155ИД4 D1 и логики формирования сигналов записи/чтения регистров микросхемы K1806ВП1-103, начальной установки, сброса счетчика адреса ОЗУ, инкремента адреса ОЗУ и сигналов записи/чтения ОЗУ, коммутации путей данных мультиплексорами D8, D9, D11, D12 K531КП11;

— регистр операции на триггере D2;

— регистр адреса модуля D36 на триггере K155TM8;

— триггер D3 завершения операции контроллера со СБИС 3У ЦМД.

Микросхемы D34, D35 K589АП16 реализуют интерфейс с внутренней

двухнаправленной шиной данных В3У.

Для формирования поля продвижения доменов с частотой 100 кГц сигнал тактового генератора частотой 6,4 МГц делится триггером D24 и на вход G микросхемы D25 поступает тактовый сигнал с частотой 3,2 МГц.

В зависимости от состояний дешифратора D1 контроллер может на-

ходиться в одном из четырех режимов работы:

Выходы дешифратора	Режим работы контроллера
9 10 11 12	работа с ОЗУ
0 1 1 1	запись/чтение РТА/РАП
1 0 1 1	задание операции K1806ВП1-103
1 1 0 1	начальная установка
1 1 1 0	сброс счетчика ОЗУ

Любая информационная операция начинается с задания адреса страницы обмена и типа операции в триггере D2. Контроллер реализует стартовый режим работы СБИС 3У ЦМД в режиме обмена одиночными страницами. Для этого требуется формирование сигнала старта операции, поступающего на вход ST микросхемы K1806ВП1-103. По завершении операции происходит останов сигналом STOP (вывод 26 FL микросхемы D25 соединен со входом STOP—25). Сигнал FL устанавливает также триггер завершения операции D3 по его тактовому входу С. Алгоритм выполнения информационных операций следующий:

Чтение страницы ЦМД:
1. Сброс счетчика ОЗУ. 2. Запись РАП (адрес страницы обмена). 3. Задание операции контроллеру. 4. Проверка завершения операции (сигнал

Требование А). 5. Сброс счетчика ОЗУ. 6. Чтение буферного ОЗУ процессором ЭВМ с проверкой контрольной суммы.

Запись страницы ЦМД:
1. Сброс счетчика ОЗУ. 2. Запись буферного ОЗУ процессором ЭВМ, включая контрольную сумму. 3. Запись РАП. 4. Задание операции контроллеру. 5. Проверка завершения операции.

Информационный обмен ЭВМ — ОЗУ. К буферному ОЗУ имеется программный доступ со стороны процессора ЭВМ через интерфейс И2. Для чтения ОЗУ процессором ЭВМ необходимо установить режим «работа с ОЗУ», задаваемый дешифратором D1. Предварительно необходимо очистить регистр операций D2. После этого пути данных будут следующие: входы-выходы данных микросхем D16, D17 через мультиплексоры D8, D9 на шину ВВ интерфейса И2, Мультиплексоры D11, D12 будут находиться в состоянии «отключено». При поступлении каждого сигнала «ввод данных» от платы И2 на шину ВВ будут транслироваться данные ОЗУ. По заднему фронту сигнала «ввод данных» будет инкрементировано значение счетчика адреса ОЗУ и при последующем обращении на шине ВВ будут установлены данные, соответствующие ин-

крементированному значению адреса ОЗУ. Перед началом цикла «чтение ОЗУ» необходимо сбросить счетчик адреса.

В режиме записи данных от ЭВМ в буферное ОЗУ значение регистра операции должно быть равно «01» — высокий уровень на выходе 5 и низкий на 9 выводе микросхемы D2. Коммутация данных следующая: шина ВД интерфейса И2, мультиплексоры D11, D12, выходы-входы микросхем памяти D16, D17. Запись производится по сигналу вывод данных от И2, по срезу сигнала «вывод данных» происходит увеличение значения счетчика адреса ОЗУ. Перед выполнением цикла «запись ОЗУ» также необходимо сбросить счетчик адреса ОЗУ.

Для правильного соответствия адресов ОЗУ перед выполнением операций запись/чтение СБИС ЗУ ЦМД также необходимо сбросить счетчик адреса ОЗУ. Следовательно, буферное ОЗУ функционирует в режиме стека FIFO.

Блок форматора данных обеспечивает работу контроллера с несколькими микросхемами памяти (до восьми) и поддерживает формат данных, приемлемый для ОС. На входе-выходе блока данные всегда представлены в байтовом формате (регистр D33). В соответствии с текущей картой годности они произвольным образом перераспределяются по микросхемам памяти. Принцип формирования массивов информации показан в таблице. Если встречается дефектный регистр, то соответствующий информационный бит сдвигается в первый годный регистр, который может принадлежать любой из микросхем памяти. Аналогично происходит формирование массива при чтении: форматор данных воспринимает данные из годных регистров, задерживая создание информационного байта при наличии дефектных регистров.

Если модуль содержит две микросхемы, то они могут быть подключе-

Формирование массива данных страницы

Номер регистра хранения	Номер микросхемы							
	1	2	3	4	5	6	7	8
1	a ₁	a ₂	X	X	a ₅	a ₄	a ₅	a ₆
2	X	a ₇	a ₈	b ₁	b ₂	b ₃	X	b ₄
3	X	b ₅	b ₆	b ₇	X	b ₈	c ₁	c ₂
4	c ₃	c ₄	c ₅	c ₆	X	c ₇	c ₈	d ₁
5	d ₂	d ₃	d ₄	d ₅	d ₆	d ₇	d ₈	l ₁
.
282

Примечание. X — бракованный регистр хранения микросхемы памяти на ЦМД; a_i — бит байта a

ны к любым двум разрядам шины данных, необходимо лишь их соответствие разрядам шины данных карты годности (браков). На остальных разрядах шины данных карты годности поддерживается потенциал высокого уровня, что обеспечит их обход как полностью дефектных (см. таблицу). По сигналу SH1 от микросхемы K1806ВП1-103 данные о годных регистрах микросхем памяти записываются в регистр браков D31.

При выполнении операции чтение ЦМД по этому же сигналу происходит запись данных с шины данных ВЗУ в регистр D32. Сигнал SH2 формирует временное окно для обработки данной порции информации, разрешая работу триггера D24. Регистры D31 и D32 синхронно сдвигаются, информация из регистров запоминается в триггере D37. Если регистр микросхемы памяти годный (отсутствует сигнал брака на выходе 9 микросхемы D37), то разрешается сдвиг регистра формирования байта D33. Информация с выхода микросхемы D37 вдвигается в регистр D33. При этом синхронно увеличивается значение счетчика байта D21. Если байт сформирован, т. е. произошло 8 сдвигов регистра D33, то данные с выходов регистра D33 переписываются в ОЗУ, увеличивается значение адреса ОЗУ и происходит сброс счетчика битов D21. Так как поток данных о бракованных регистрах микросхем памяти случайный, то байт может быть сформирован в любой момент временного окна. После записи в ОЗУ продолжается обработка остальных информационных битов порции информации. Триггер D24 обеспечивает прохождение 8 сдвигов регистров D31 и D32 во время активного сигнала SH2. Алгоритм операции «запись ЦМД» отличается от алгоритма операции чтения, так как информация, подлежащая записи в ЦМД, обрабатывается с учетом данных о дефектных регистрах и данных записи в регистре D33. Синхронный сдвиг регистров D31 и D32 происходит раньше, чем сдвиг регистра D33. В начале выполнения операции во время действия сигнала TO микросхемы K1806ВП1-103 первая информация поступает из ОЗУ на вход регистра D33. По сигналу SH1 записываются данные о дефектных регистрах в регистр браков D31. Во время активного сигнала SH2 формируются данные записи в регистре D32. Синхронный сдвиг регистров D31 и D32 в регистр D32 всегда соответствует записи «Лог. 0» с выхода 11 элемента D15 при высоком уровне на выходе 20 регистра D31, так как при этом блокируется прохождение информации с выхода 20 регистра D33 через элемент D15. Одновременно будут заблокированы сдвиг регистра D33 и работа счетчика битов D21. Информация с выхода ре-

гистра D33 записывается только в том случае, если в триггере D37 нет данных о дефектности регистра микросхемы памяти. После семи сдвигов регистра D33 восьмой сдвиг заменяется занесением информации в регистр D33 из ОЗУ, так как сдвиг регистра D32 всегда осуществляется раньше. Развязка выходов регистра D33 от шины данных ОЗУ D16 и D17 производится мультиплексорами D11 и D12 с тристабильными выходами.

Интерфейс управления СБИС ЗУ ЦМД образуют следующие выходы контроллера:

- COIL — разрешение работы формирователей токов продвижения доменов;
 - $\overline{FA}, \overline{FB}, \overline{FC}, \overline{FD}$ — четыре импульса, сдвинутых относительно друг друга на четверть периода следования для управления формирователями токов продвижения, последовательность фаз следующая: $\overline{FD}, \overline{FA}, \overline{FB}, \overline{FC}, \overline{FD}, \dots, \overline{FC}, \overline{FD}$;
 - PROM — разрешение работы ПЗУ браков модулей памяти;
 - $\overline{ROC}, \overline{ROS}$ — сигналы репликации доменов;
 - \overline{GEN} — сигнал разрешения работы генератора доменов;
 - TO — переключатель вывода;
 - TI — переключатель ввода;
 - \overline{CDR} — строб усилителя чтения;
 - ODD — разделение каналов детекторов ЦМД;
 - \overline{PCLK} — продвижение счетчика адреса ПЗУ браков модулей памяти;
 - RWR — тип операции, выполняемой СБИС ЗУ ЦМД (используется также для переключения приема-передатчиков шины данных выбранного модуля);
 - MOD0...MOD2 — адрес модуля;
 - UB — сигнал отства модуля памяти;
 - B1...B8 — шина данных карты годности, на которую транслируются данные ПЗУ браков выбранного модуля;
 - D1...D8 — двунаправленная шина данных.
- Передатчики сигналов управления выполнены на буферных формирователях с открытым коллектором K155ЛП9.

Программный интерфейс. Обмен информацией между контроллером и центральным процессором ЭВМ осуществляется посредством программных операций с опросом флага или программных операций с использованием средств прерывания программы через устройство параллельного обмена И2.

Устройство И2 способно хранить 16 разрядов одного выходного слова или двух выходных байтов в регистре «выходной буфер». Записанные в выходной буфер данные выводятся ВДО...15 (см. рис. 2), передаются в контроллер через кабель связи. Любая программная операция, которая загружает байт или слово в выходной буфер, вызывает появление сигнала «вывод данных», информирующего контроллер о передаче данных.

Входные данные (выводы ВВО...15 (см. рис. 2)) считываются в канал ЭВМ в течение цикла «ввод». После считывания вырабатывается и посылается в контроллер сигнал «ввод данных».

Регистры интерфейса занимают три последовательных слова адресного пространства страницы ввода-вывода микроЭВМ. Переключки, установленные на предприятии-изготовителе, определяют следующие адреса:

регистр команд и состояний — 167770 (регистр VKCS); выходной буфер — 167772 (регистр VKWR); входной буфер — 167774 (регистр VKRD).

Формат и назначение битов регистров контроллера приведены на рис. 3. Используя соответствующие биты регистров, пользователь может организовать программный обмен с регистрами микросхемы К1806ВП1-103, буферным ОЗУ, задать режим работы, начать операции поиска, записи, чтения СБИС ЗУ ЦМД, изменить адрес модуля памяти и проверить его наличие в системе. Устанавливая разрешение прерывания в регистре

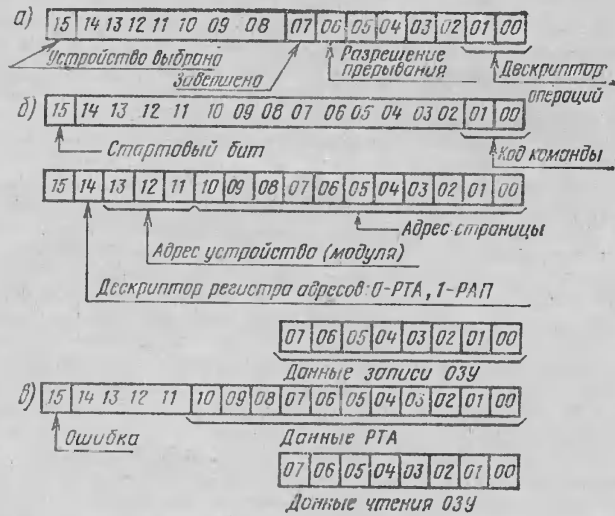


Рис. 3. Форматы регистров контроллера:

а) регистр команд и состояний VKCS; б) регистр записи VKWR; в) регистр чтения VKRD

VKCS, можно прервать программу по завершении операции.

Для включения системы памяти в состав ОС необходима программа-драйвер (рис. 4), обслуживающая до восьми устройств памяти, содержащих 4...8 микросхем. В модулях с четырьмя микросхемами страница памяти СБИС ЗУ ЦМД равна 128 байтам, при восьми — 256 байтам. Драйвер легко настроить на работу с другими модулями, однако необходимо помнить, что размер страницы для одной микросхемы ра-

вен 32 байтам. Формирование контрольной суммы при записи в ЦМД осуществляется в драйвере. В режиме чтения драйвер сравнивает вычисленное и считанное значения контрольной суммы для обнаружения ошибок.

Драйвер предназначен для работы с ОС РАФОС и ФОДОС. Каждый модуль памяти представлен в ОС как устройство с именем VK(N) (N=0, 1, ..., 7). Для работы с VK служат стандартные команды монитора, например:

INIT VK(N) — инициализация каталога модуля N;
 COP/SYS — SY: — YKO: — копирование ОС, например с НГМД;
 COP/BOOT — VKO: RT11FB.SYS — VKO: — копирование системного загрузчика в соответствующую область устройства;
 BOOT VKO: — загрузка операционной системы с VKO.
 Для запуска ОС разработан аппаратный загрузчик.

```

TITLE VK BUBBLE-MEMORY "DISK" HANDLER
ENABL LC,CBL
IDENT /V01.R5/
SBTTL НАСТРОЙКА ГЕНЕРАЦИИ ДРАЙВЕРА
*****
* ДРАЙВЕР ЧИТА-ПАНТИ *
* КОНТРОЛЛЕР К1806ВП1-103 *
* ВЕРСИЯ ДЛЯ ОС РАФОС *
* РАЗРАБОТЧИК: ИНАТ В.К. *
* ДИМ-1 — 256 БАЙТОВ *
* ДИМ-2 — 512 БАЙТОВ *
* MASK-МНОГОМОДУЛЬНЫЙ РЕЖИМ *
* ДРАЙВЕР ОБСЛУЖИВАЕТ *
* 8 УСТРОЙСТВ *****
PARAMETERS
DIR#2
SCALL DRDEF
IF EQ DIR-1
OP1 =6314
OPER =6303
CNT =128
RAZM =256
IF
OP1 =240
OPER =240
CNT =256
RAZM =512
ENDC
SBTTL DEFINITIONS
VKCS =167770 ;АДРЕС
VKWR =VKCS+2 ;РЕГИСТР
VKRD =VKCS+4 ;УСТРОЙСТВА
VKVCS =300 ;ВЕКТОР
DELDF
WCNT =Q.WCNT-Q.BLKN
WBUF =Q.WBUF-Q.BLKN
FUNC =Q.FUNC-Q.BLKN
  
```

ОПИСАНИЕ ДЕСКРИПТОРОВ КОМАНД

```

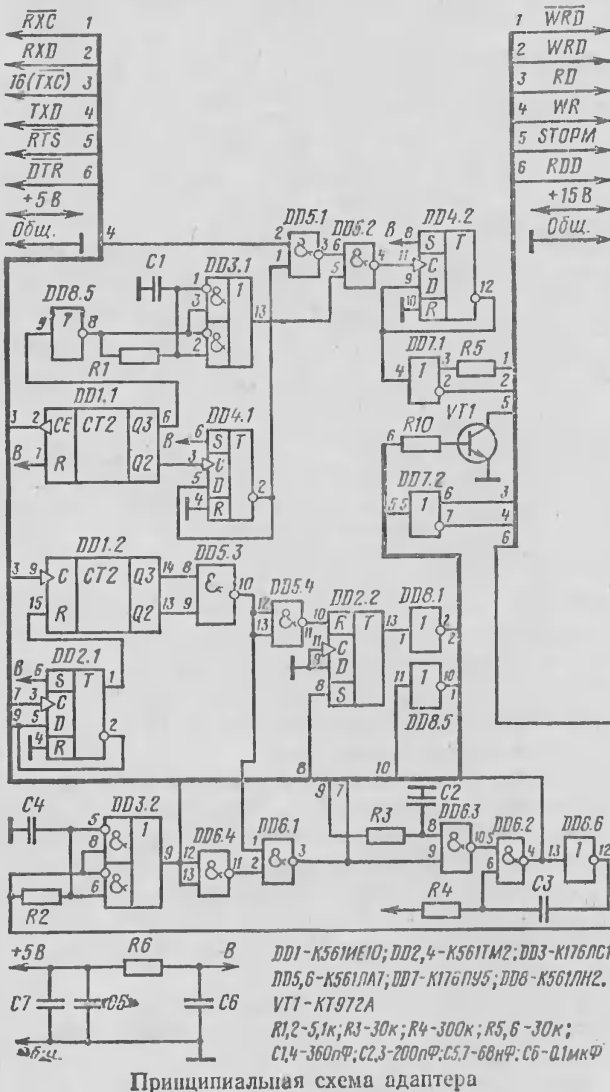
; VKCS
DSKZU =0 ;ДЕСКРИПТОР ОЗУ
DSKREG =1 ;ДЕСКРИПТОР РЕГИСТРОВ
DSKOPR =2 ;ДЕСКРИПТОР ОПЕРАЦИИ
SH =3 ;СБИС
; VKWR
CR0 =1 ;КОД КОМАНД ЧТЕНИЯ
CR1 =2 ;ЗАПИСЬ
GO =100000 ;СТАРТ. БИТ ОПЕРАЦИИ
DSKMAP =40000 ;ДЕСКРИПТОР РЕГИСТРА
;АДРЕСА СТРАНИЦ
; DRDEF VK,0,FILST#,RAZM,VKCS,VKVEC
RETRY #=
SBTTL ВХОД В ДРАЙВЕР В НОРМАЛЬНОМ РЕЖИМЕ
DRUEG VK
MOV #RETRY,CNTRR
MOV #VKCS,RO
MOV #VKWR,RT
MOV #VKRD,R2
MOV #VKVCS,R4
ASL (4)
WORD OP1
TSTB FUNC(4)
BNE 1#
; IF DF MASK
MOV 2(4),RS
BIC #CCHASK,RS
BEQ 51#
CALL MODUL
BCC 50#
CALL RETMOD
JMP ERDONE
51# CLR NMOD
CLR DFST
ENDC
50# ASL WENT(4)
BEQ 7#
BCC BLADR
JMP WRITEN
JMP DONE
; IF DF MASK
MODUL: MOV RS,R3
; SWAB R3
; ASL R5
; ASL R5
; ASL R5
; ASL R5
; MOV RS,NMOD
; MOV #DSKREG,(0)
; MOV (2),AM0
; BIC #C3777,AM0
; ASL R5
; MOV R3,OFST
; MOV PC,R5
; ADD #ANO-X-2,R5
; ADD R3,R5
; MOV (5),R5
; DIS NMOD,R5
; MOV R3,(1)
; NOP
; TST (0)
; BPL 7#
; TST (PC)+
; SEC
; RTS
PC
; AND: WORD 0 ;РЕГИСТРОВЫЕ
; WORD 0 ;ФАЙЛ
; WORD 0 ;ТЕКУЩИХ
; WORD 0 ;АДРЕСОВ
; WORD 0 ;УСТРОЙСТВА
; WORD 0
; WORD 0
  
```


Ю. Ю. Сорокин, С. П. Максимьяк, В. В. Субач

ПРОСТОЙ АДАПТЕР ДЛЯ ПОДКЛЮЧЕНИЯ КАССЕТНОГО МАГНИТОФОНА К ЛИНИИ ПОСЛЕДОВАТЕЛЬНОГО ОБМЕНА

Адаптер используется для подключения кассетного магнитофона к последовательному каналу, формируемому микросхемой КР580ВВ51 в режиме «асинхронный 1»; аппаратно реализует алгоритм фазового кодирования (СМ1) с малой чувствительностью к детонации лентопротяжного механизма (ЛПМ) и перекосам магнитной ленты, не требует стирающей головки и позволяет вести асинхронный побайтный обмен со скоростью до 6К бод даже при использовании ЛПМ третьего класса.

Адаптер сохраняет работоспособность при отклонении скорости движения магнитной ленты на $\pm 10\%$ от номинальной, так как синхронизация при чтении осуществ-



вляется не только по стартовому но и по каждому отдельному биту в байте.

Сигналы управления модемом (RIS DIR) в микросхеме КР580ВВ51 позволяют организовать через адаптер программное управление пуском и остановом двигателя ЛПМ, переключение режимов запись-чтение (при использовании универсальной магнитной головки), работу со «сквозным» каналом.

Адаптер выполнен на 8 микросхемах серий К561 и К176 и транзисторе, имеет размеры 130×45×18 мм.

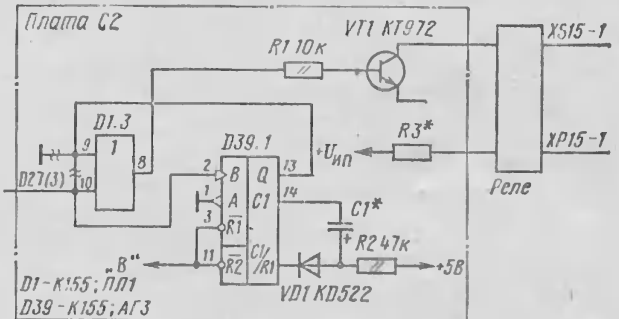
Телефон для справок: 246-05-80, Москва
Сообщение поступило 23 февраля 1987 г.

УДК 681.3

Ю. Ю. Сорокин, С. П. Максимьяк, В. В. Субач

АВТОМАТИЧЕСКИЙ ОСТАНОВ ПРИВОДА В НАКОПИТЕЛЕ «ЭЛЕКТРОНИКА ГМД-7012»

Останов вращения гибкого диска в приводе накопителя на гибких магнитных дисках (НГМД) на время длительного отсутствия обмена информацией между НГМД и ЭВМ существенно увеличивает срок эксплуатации дисков [1].



Программно-управляемый останов привода аппаратно реализован в НГМД «Электроника 6012/6022», СМ-5640, однако он полностью отсутствует в накопителе «Электроника ГМД-7012». Простая доработка контроллера этого устройства (см. рисунок), заключающаяся в установке нескольких дополнительных элементов на плату С2 [2], позволяет организовать автоматический останов привода сразу или через произвольное время (определяемое емкостью конденсатора) после отвода магнитной головки от поверхности дискеты. Последнее означает завершение контроллером операции обмена с гибким диском [3]. Для коммутации цепи питания асинхронного двигателя, привода можно применить электромагнитное или тиристорное (желательно с оптрошной развязкой) реле [4].

ЛИТЕРАТУРА

1. Адасько В. И., Каган Б. М., Пац В. Б. Основы проектирования запоминающих устройств большой емкости. — М.: Энергоатомиздат, 1984. — 288 с.
2. Устройство ввода-вывода информации на гибких магнитных дисках «Электроника 7012» Ц6М2.791.005 ПС. Паспорт.
3. Устройство ввода-вывода информации на гибких магнитных дисках «Электроника 7012» Ц6М2.791.005 ОС. Опись.
4. Шопен Л. В. Бесконтактные электрические аппараты автоматики. — М.: Энергоатомиздат, 1986. — 210 с.

Статья поступила 7 июля 1987 г.

УДК 681.3.06

Р. Р. Петропавловский, А. Е. Ходак

ДРАЙВЕР ДЛЯ ВЫВОДА ГРАФИЧЕСКОЙ И АЛФАВИТНО-ЦИФРОВОЙ ИНФОРМАЦИИ

Предлагается программное обеспечение, рассчитанное на работу микроЭВМ типа «Электроника 60» в операционной системе RT-11 (РАФОС или ФОДОС), для более эффективного использования возможностей печатающего устройства УВВПЧ-30—004.

Системная программа выполнена в виде драйвера внешнего устройства УВВПЧ. Драйвер совместим сверху вниз с обычным драйвером печатающего устройства в RT-11 и может заменять его в системах, использующих УВВПЧ. Оформление программы в виде драйвера позволяет органично использовать ее в сочетании с существующим программным обеспечением RT-11. Программа написана на языке MACRO-11 (рис. 1). Рабочему системному файлу присвоено имя GP.SYS (Grafic Printer). Размер файла на диске — четыре блока. В загруженном состоянии драйвер занимает 660 слов оперативной памяти.

Алфавитно-цифровая информация выводится драйвером тремя 7-битными наборами кодов для печати латинского, русского и смешанного (русско-латинского) шрифтов (ГОСТ 13052—74), 8-битным набором (ГОСТ 19768—74) и набором кодов для печати греческого шрифта. Переходы между наборами выполняются с помощью стандартных символов перехода. В программе предусмотрена возможность изменения в процессе печати расстояния между символами и строками, многократной пропечатки символов, размеров и вида шрифта, положения текста на странице. Изменение параметра алфавитно-цифровых литер в рабочей программе кодируется последовательностями из двух байтов. Первый байт идентифицирует изменяемый параметр, второй — значение этого параметра. Например, последовательность байтов 003 002 устанавливает интервал между символами (байт 003), равный двум элементарным шагам печатающей головки (байт 002). Изменение этих параметров производится также командой монитора SET. Идентифицирующие коды и соответствующие им параметры приведены в табл. 1.

Для определения положения выводимой алфавитно-цифровой и графической информации в драйвере предусмотрена система координат, горизонтальная ось которой направлена слева направо, а вертикальная — снизу вверх. В начале печати точка отсчета системы координат устанавливается по горизонтали в крайнем левом положении, а по вертикали — в том месте, где находится печатающая головка. При переводе формата начало координат устанавливается на следующей позиции формата. Позиции формата располагаются на бумажной ленте через одинаковые интервалы относительно начального положения печатающей головки. После перевода формата, а также выполнения любой функции управления выводом графической информации текст начинает печататься от начала координат. При переводе строки положение текста смещается вниз по вертикальной оси на величину, определяемую размером символов и интервалом между строками. При выполнении команды «возврат каретки» печатающая головка устанавливается в начало текущей строки.

SET-параметры драйвера SHIFT, FORMAT, HOR, NOHOR (см. табл. 1) обеспечивают дополнительный сервис: сдвиг начального положения системы координат относительно левого края бумажной ленты; изменение величины формата; установку горизонтального или вертикального режима печати. При вертикальном режиме печати система координат поворачивается на 90° против часовой стрелки относительно обычного горизонтального режима. Эти параметры не имеют идентифицирующего кода, так как не могут быть изменены в процессе печати.

Для графического отображения функций драйвер имеет ряд специфических возможностей. Функция, заданная таблицей, может быть выведена либо дискретно по точкам, либо в виде ломаной линии, соединяющей точки, для чего в драйвер встроены линейный интерполятор. Точки на графике изображаются произвольными символами. Это особенно удобно при выводе экспериментальных данных. Функция, определенная в точках, равноотстоящих по одной из осей, отображается с автоматическим инкрементом по соответствующей оси и линейной интерполяцией. Предусмотрена произвольная установка начала координат и масштабов по каждой из осей, а также установка параметров пунктира, т. е. числа печатаемых и непечатаемых точек. Имеется возможность (например, для копирования экрана гра-

Таблица 1

Идентифицирующий код	SET-параметр	Изменяемый параметр
001	RPT	Число пропечаток каждого символа
002	SMOP	Вид модификации шрифта
003	SINT	Интервал между символами
004	LINT	Интервал между строками
016	RUS	Установка русского 7-битного набора
017	LAT	Установка латинского 7-битного набора
022	ASC7	Установка смешанного 7-битного набора
023	ASC8	Установка 8-битного набора
024	USER	Установка греческого набора
—	SHIFT	Сдвиг текста (графика) относительно левого края бумаги
—	FORMAT	Изменения величины формата
—	HOR	Переход в горизонтальный режим
—	NOHOR	Переход в вертикальный режим

```

.TITLE GP - GRAFIC PRINTER HANDLER
.IDENT /Y05.01/
; MODAK 1986
; LPI TOT
; Leningrad tel 5526407
.NLIST COM
.SBTTL Macros and defenitions
.MCALL .DRDEF,GRFDEF
.DRDEF GP,376,MONLYX,0,177504,174
IE = 100
RDY = 200
ERR = 100000
XXMAX = 479.
XXCHAR = 5.
YXCHAR = 7.
GRFDEF
XPNT = 222
FORM = 111
SMOD = 114
INT = 120
ADDX = 131
ADDY = 132
PRS = 136
PRL = 137
.SBTTL Set options
.ENABLE LSB
.DRSET CSR,6PB-0.WORD-2,0.CSR,OCT
.DRSET VECTOR,GPSTRT-0.WORD-2,0.WORD,OCT
.DRSET HANG,<BNI GPERR-ORHANG+.>,0.HANG,RO
.DRSET RPT,ORRPT-M.CHP,0.CHP,NUM
.DRSET SMOD,0.SMOD-M.CHP,0.CHP,OCT
.DRSET SINT,ORSINT-M.CHP,0.CHP,NUM
.DRSET LINT,ORLINT-M.CHP,0.CHP,NUM
.DRSET FORMAT,ORFFS-0.WORD-2,0.FORM,NUM
.DRSET SHIFT,Y.SHFT-0.WORD,0.SHFT,NUM
TXTSET
HOR
O.FORM: NEG R0
O.WORD: ADD PC,R3
MOV RO,ER3
DR 0.GOOD
O.CHP: ADD PC,R3
M.CHP: MOVB RO,ER3
O.GOOD: CLC
RTS PC
O.HANG: MOV (PC)+,R3
BNI RETURN-ORHANG+.
MOV R3,ORHANG
RTS PC
O.CSR: MOV RO,176
MOV RO,6PS
ADD #2,RO
BR 0.WORD
.SBTTL Driver entry
.ENABLE LSB
.DRBE6 GP
MOV GPCQE,R4
Driver entry
ADD #QXWCNT,R4
ASL OR4
BCC SPERR
TST -QXWCNT(R4)
BEQ BEGIN
JMP CNT
.SBTTL Interrupt service
.DRAST GP,4,GPEXIT
BIT #ERR:RDY,a(PC)+
GPS: GPXCSR
ORHANG: BNI RETURN
BEQ RETURN
CLR #GPS
.FORK GPF3LK
MOV GPCQE,R4
ADD #QXWCNT,R4
MOV #4,R3
PRINT
MOV R3,(PC)+
4X: 0
RETURN: BIS #IE,EGPS
RTS PC
GPERR:
BIS #HDERRX,ORXCSW-QXWCNT(R4)
GPEXIT: JMP GPFIN
.SBTTL Support subroutines
TRPCB
BEGIN: CALL SETPAR
RESET: SHIFT

```

```

JSR R5,PRINT
.BYTE 1,CR
CLR Y.OLD
CLR Y.OLD
BR RSTXY
.SBTTL Text printing
.ENABLE LSB
1X: CMPB R5,#SPACE
BEQ 2X
MOVB R5,P.BYT
CALL TEXT
2X: ADD X.CHAR,X.NEW
ASLB (PC)+
TABFLG: 0
RPT: BNE CHRPRI
JSR R5,INPBYT
CMPB R5,#SPACE
BLO CRTST
CHRPRI: ASLB (PC)+
TABCNT: 1
BNE 3X
INC TABCNT
3X: CALL TEST
BNI 2X
TXT
MOV X.NEW,R3
MOV R3,R1
ADD (PC)+,R3
Text printing
X.SIMB: XXCHAR
CMP R3,(PC)+
X.MAX: XXMAX
ORBR: BGT 1X
CMP Y.NEW,Y.OLD
BNE 1X
TST (PC)+
T.PRI: 0
BEQ 4X
BPL 1X
MOV ORRPT,RO
MOV RO,T.PRI
SOB RO,1X
DIV (PC)+,RO
X.CHAR: 0
MOV R1,T.PRI
BNE 1X
MOVB RO,X.PNT
JSR R5,PRINT
.BYTE 3,US,XPNT
X.PNT: .BYTE 0
4X: MOV R3,X.OLD
MOV R5,5X
JSR R5,PRINT
.BYTE 1
5X: .BYTE 0
BR 2X
.SBTTL Text support
TABSET: MOV TABCNT,TABFLB
MOV R3,R5
BR CHRPRI
PROC
CHRTST: JSR R5,CHANGE
ORLFS:Y.CHAR: CHANGE LF,YXCHAR,LFSET
ORFFS: CHANGE FF,-700.,FFSET
CHANGE HT,#SPACE,TABSET
CHANGE CR,1,CRSET
CHANGE ESC,1,GRFTST
CHANGE SI,TRLAT,PRMT
CHANGE SO,TRRUS,PRMT
CHANGE DC4,TRASC7,PRMT
CHANGE DC3,TRASC8,PRMT
CHANGE DC2,TRUSER,PRMT
CHANGE 1,ORRPT-M.SIM,PRMS
CHANGE 2,0.SMOD-M.SIM,PRMS
CHANGE 3,ORSINT-M.SIM,PRMS
CHANGE 4,ORLINT-M.SIM,PRMS
0
BR RPT1
FFSET: MOV R3,RO
MOV Y.OLD,F.OLD
1010R: CMP Y.NEW,R3
BGT FFP
ADD RO,R3
BR 1010R
LFSET: JSR R5,PRINT
.BYTE 3,0,LF,0

```

Рис. 1. (Начало)

Text support

```

SUB      R3,Y.OLD
OXLFF:  SUB      R3,Y.NEW
        BR       RPT1
PRMT:   MOVB    ORCODE,RO
        CALL    O.TEXT
        CMFB    ORCODE,RO
        BNE    9X
        JSR    R5,PRINT
        .BYTE  1,0
        BR     RPT1
PRMS:   ADD     PC,R3
H.SIM:  MOV     R3,33X
        JSR    R5,IMPBYT
        MOVB   R5,@(PC)+
38X:    0
9X:     CALL   SETPAK
RPT2:   BR     RPT1
.SBTTL  GRAFIC PRINTING ROUTINES
.ENABLE LSB
ABS:    CALL   XYBEG
        MOV    R1,X.ZERO
        MOV    R0,Y.ZERO
GRFRPT: BR     RSTXY
GRFBEG: CLR    T.FUNC
4X:     MOV    (PC)+,R3
C.PRI:  177777
        COM    (PC)+
T.FUNC: 0
        BNE   GRAFIC
        MOV   (PC)+,R3
C.IGN:  0
GRAFIC: TST    R3
        BEQ   4X
        DEC   R3
        MOVB  T.FUNC,R.BYT
        BEQ   6X
GRFPNT: CALL   TEST
        BLOS  TEXT
6X:     RETURN
GRFTST: JSR    R5,IMPBYT
        JSR   R5,CHANGE
        CHANGE 0,1,ABS
        CHANGE 1,<.BYTE C.PRI-M.PRM/2,C.IGN-M.PRM/2>,PRM
        CHANGE 2,<.BYTE X.MUL-M.PRM/2,X.DIV-M.PRM/2>,PRM
        CHANGE 3,<.BYTE Y.MUL-M.PRM/2,Y.DIV-M.PRM/2>,PRM
        CHANGE 4,200,MASSIV
        CHANGE 5,100000,MASSIV
        CHANGE 6,1,MASSIV
        0
        CMFB  R5,*SPACE
        BLOS  RPT2
        MOVB  R5,P.BYT
        CALL  INPUT
        MOV   R5,R3
        CLR  T.XY
5X:     CALL  XYIMP
        SUB  (PC)+,R1

```

GRAFIC PRINTING ROUTINES

```

X.HALF: IMCHAR/2
        MOV    R1,X.NEW
        ADD   (PC)+,RO
Y.HALF: YRCHAR/2
        MOV    R0,Y.NEW
        CALL  GRFPNT
        SOB   R3,1X
        BR   GRFRPT
PRM:    CALL  INPUT
        MOVB  R3,R1
        ASL   R1
        ADD  PC,R1
M.PRM:  MOV   R5,@R1
        CLRB R3
        SWAB R3
        BNE  PRM
        BR  RPT2
MASSIV: GRF
.SBTTL  SUPPORT SUBROUTINES
BPROG
CNT:    MOV   97X,-(SP)
        MOV   98X,R5
INPBYT: TST   @R4
        BEQ  6PDONE
        INC  @R4
        MOVB @-(R4),@SP
        INC  (R4)+
        RTS  R5
6PDONE: MOV   (SP)+,(PC)+
97X:    0
        MOV   R5,(PC)+
98X:    0
        JSR  R5,PRINT
        .BYTE 1,US
6PFIN:  CLR   @6PS
        CLR  6PFBK+2
        .DRFIN 6P
6PFBK:  .WORD 0,0,0,0
        .DREND 6P
        .END

```

Errors detected: 0

*** Assembler statistics

```

Work file reads: 32
Work file writes: 27
Size of work file: 13494 Words ( 53 Pages)
Size of core pool: 12288 Words ( 48 Pages)
Operating system: RT-11

```

Elapsed time: 00:00:00.00

,LP:6PTXY=DX:6FTXT,DK:6PAC/H/N:SEQ:LOC:BIN:END:TOC:SYM

Рис. 1

фического монитора) растрового вывода изображений, использующего всю матрицу печатающей головки.

Вывод графиков управляется символом ESC (033). Код, непосредственно следующий за ESC, идентифицирующий. Координаты точек и графические параметры вводятся в целом формате (INTEGER * 2 для языка Фортран IV), причем передается сначала младший, а затем — старший байт. Это позволяет непосредственно, без дополнительной обработки, вводить в драйвер последовательности целых чисел. Идентифицирующие коды и соответствующие им функции приведены в табл. 2. Рассмотрим, например, последовательность из шести байт (033 000 000 001 377 377): байт 033 (ESC) указывает на то, что последующие байты управляют выводом графической информации; 000 — идентифицирует установку нового начала координат; третий байт 000 является младшей, а четвертый 001 — старшей частью числа 256 (десятичное) в 16-битном целом формате. Это число, умноженное на соответствующий масштаб, определяет горизонтальное смещение новой системы координат относительно старой; 377 является младшей, а 377 — старшей частью числа — 1, определяющего вер-

тикальное смещение новой системы координат. В начале печати все масштабы устанавливаются равными единице, а параметры пунктира — на вывод сплошной линии, когда число непечатаемых точек равно нулю.

При растровом выводе изображений идущая за идентифицирующим кодом последовательность байтов определяет вид столбцов растра. Старший бит этих байтов должен быть установлен, следующий бит соответствует верхней позиции столбца, младший бит — нижней. Если бит установлен, то соответствующая позиция столбца должна быть ударной. Первый байт со сброшенным старшим битом определяет конец растрового вывода. Если растр выводится после выполнения функции управления выводов графиков, то он печатается из начала координат. Если перед растром выводились алфавитно-цифровые литеры, то положение печатающей головки не изменяется. После вывода растра текст начинает печататься непосредственно за растром.

Драйвер позволяет выводить графическую и алфавитно-цифровую информацию не только с помощью программ на языке MACRO-11, но и непосредственно из программ на языках высокого уровня, например Форт-

Таблица 2

Идентифицирующий код	Одно слово	Два слова или массив	Функция
000	XZ	YZ	Установка начала координат в точке ()
001	PR1	BLANK	Установка параметров пункта
002	XMUL	XDIV	Установка масштаба по горизонтали
003	YMUL	YDIV	Установка масштаба по вертикали
004	N	N слов	Ввод вертикальных координат точек, выводимых с инкрементом по горизонтали и линейной интерполяцией
005	N	N слов	Ввод горизонтальных координат точек, выводимых с инкрементом по вертикали и линейной интерполяцией
006	N	2N слов	Ввод координат точек, выводимых с инкрементом и линейной интерполяцией
007	Последовательность байтов с установленным старшим битом		Растровый вывод изображения
040—377	N	2N слов	Ввод координат точек, выводимых в виде алфавитно-цифровых символов

ран IV. При выводе графической информации из программ на языке Фортран IV используется стандартный последовательный вывод в символьном формате (тип A). Программа (рис. 2) выводит на УВВЧ график $Y=\text{sqrt}(X)$, приведенный на рис. 3.

Для вывода алфавитно-цифровой и графической информации на экран телевизора или видеоконтрольного устройства написан драйвер D1 с использованием дисплейного набора модулей («ДИНАМО») в стандарте КАМАК. Драйверы D1 и GP программно совместимы.

За дополнительной информацией обращаться на кафедру теоретических основ теплотехники Ленинградского политехнического института им. М. И. Калинина, Тел. 552-64-07.

УДК 681.3.06

С. Н. Попов

ПРОГРАММА «ИМИТАТОР ДОС» ДЛЯ 8-РАЗРЯДНЫХ ПЭВМ

Наличие в составе ПЭВМ накопителей на гибких магнитных дисках или электронного диска позволяет использовать одну из стандартных ДОС и пакеты программ, работающие под ее управлением. В случае простых ПЭВМ, не имеющих НГМД, оснастить ПЭВМ разнообразным программным обеспечением гораздо сложнее. В какой-то мере решением этой проблемы яв-

```

PROGRAM BPFEX
C///// Подготовка массива
INTEGER*2 MD(160)
DO 10 I=1,160
MD(I)=(I-1)**2
C///// Инициализация вывода на GP
OPEN (UNIT=3,NAME='GP:')
110 FORMAT (X,200A2)
120 FORMAT (X,68(1H-'/'O',2A2/'O',A2,'GRAFIK',A2,'Y=SQRT(X)'A',A2,'Y=SQRT(X)'A',A2)
130 FORMAT (X,3A2,'Y',/'O',/X,/'O1',/X,/'O0',1A2)
140 FORMAT ('1',66(1H-))
C///// Установка интервала между символами - 1
WRITE (3,110) 3+1**400
C///// Установка интервала между строками - 3
WRITE (3,110) 4+3**400
C///// Установка тройной пропечатки и уверенного шрифта.
C///// Печать заголовка
WRITE (3,120) 1+3**400,2+137**400,"16,"17
C///// Установка одинарной пропечатки и нормального шрифта
WRITE (3,110) 1+1**400,2+37**400
C///// Установка сплошной линии
WRITE (3,110) "33+1**400,"177777,0
C///// Установка масштабов по X и по Y - 1/10
WRITE (3,110) "33+2**400,10,1,"33+3**400,10,1
C///// Печать рамки графика в виде массива из 5 точек
WRITE (3,110) "33+6**400,5,0,-34,36,-34,36,-20,0,-20,0,-34
C///// Установка начала координат и печать разметки осей
WRITE (3,130) "33+0**400,1,-25
C///// Установка начала координат и параметров пункта 1,1
WRITE (3,110) "33+0**400,1,-7,"33+1**400,1,1
C///// Печать оси Y в виде массива из 2 точек
WRITE (3,110) "33+6**400,2,0,0,0,10
C///// Установка параметров пункта 20 точек, 20 пропусков
WRITE (3,110) "33+1**400,20,20
C///// Печать осей X, Y в виде массива из 3 точек
WRITE (3,110) "33+6**400,3,0,10,0,0,32,0
C///// Установка параметров пункта 1,1
WRITE (3,110) "33+1**400,1,1
C///// Печать оси X в виде массива из 2 точек
WRITE (3,110) "33+6**400,2,32,0,0,0
C///// Установка тройной пропечатки и масштабов
WRITE (3,110) 1+3**400,"33+2**400,1,80,"33+3**400,1,2;
C///// Печать 4 точек в виде символа "*" (код '32')
WRITE (3,110) "33+52**400,4,(MD(I),1,1=40,160,40)
C///// Установка одинарной пропечатки
WRITE (3,110) 1+1**400
C///// Печать графика X=f(Y)
WRITE (3,110) "33+5**400,160,(MD(I),1=1,160)
C///// Перевод формата
WRITE (3,140)
STOP 'BPFEX'
END
    
```

Рис. 2



Рис. 3

Статья поступила 15 января 1987 г.

ляется описываемая ниже программа «имитатор ДОС».

Для 8-разрядных ПЭВМ наибольшее распространение получили операционные системы CP/M, МикроДОС, ОС-1800. Все эти ДОС реализуют одни и тот же набор системных вызовов. Прикладные программы для осуществления операций по вводу-выводу информации производят вызов операционной системы в соответствии со строго регламентированными правилами.

В этих ДОС подпрограммы обслуживания внешних устройств выделены в отдельный модуль — модуль программ ввода-вывода (МПВВ). Такое построение ДОС дает возможность легко адаптировать их под любую аппаратуру. Это позволило создать большое число пакетов программ самого различного назначения, многие

из которых осуществляют только ввод информации с клавиатуры и вывод сообщений на экран дисплея.

С другой стороны, под управлением ДОС работают трансляторы со многих языков программирования высокого уровня. Применение их в разработке программного обеспечения для ПЭВМ без НГМД сдерживается тем, что полученная в результате их работы программа в объектных кодах содержит в себе подпрограммы ввода-вывода, ориентированные на обращение к ДОС.

Программа «имитатор ДОС» (см. таблицу) позволяет запускать на ПЭВМ без НГМД различные программы, предназначенные для работы с ДОС CP/M, ОС-1800 или МикроДОС. Естественно, что это должны быть программы, осуществляющие при своей работе только ввод-вывод данных с консоли. «Имитатор ДОС» загружается в ОЗУ ПЭВМ с ячейки с адресом 100H. Прикладная программа загружается с адреса 300H. Запуск программы в работу производится с адреса 100H.

Таблица

<pre> 7400 = BASE EQU 7400H ;ПОДПРОГРАММЫ ВВОДА-ВЫВОДА РЕЗИДЕНТНОГО МОНИТОРА MONITOR EQU OF800H ;ЗАПУСК МОНИТОРА CO EQU OF809H ;ВЫВОД НА ЭКРАН CI EQU OF803H ;ВВОД С КЛАВИАТУРЫ CSTS EQU OF812H ;СТАТУС КЛАВИАТУРЫ LO EQU CO ;ВЫВОД НА ПРИНТЕР ; (НЕТ ПРИНТЕРА) </pre>		<pre> 7400 223C75 7403 EB 7404 223E75 7407 EB 7408 60 7409 69 740A 224075 740B 210000 7410 39 7411 224275 7414 310076 7417 F5 7418 79 7419 FE0D 741B D84274 </pre>		<pre> ;СОХРАНИТЬ СОСТОЯНИЕ РЕГИСТРОВ ;ПРИКЛАДНОЙ ПРОГРАММЫ EDOS: SHLD HLL ; HL XCHG ; DE SHLD DEL XCHG MOV H,B MOV L,C SHLD BCL ; BC LXI H,O DAD SPL SHLD SPL ; SP LXI SP,BASE+200H PUSH PSW MOV A,C CPI 13 JNC MRET ;ИГНОРИРОВАТЬ ;ОБРАЩЕНИЯ К ДИСКУ ;СИСТЕМНЫХ ВЫЗОВОВ ДОС MVI H,O MOV L,C DAD H MOV B,H MOV C,L DAD H,TACL DAD B MOV C,H INX H MOV H,M MOV L,C </pre>									
<pre> 0100 0100 C30401 JMP START 0103 20 DB 20H ; В ЭТОЙ ЯЧЕЙКЕ НЕОБХОДИМО ; ПОМЕСТИТЬ РАЗМЕР .COM ФАЙЛА ; В СТРАНИЦАХ (256 БАЙТ) </pre>		<pre> 741E 2600 7420 69 7421 29 7422 44 7423 4B 7424 212D74 7427 09 7428 4E 7429 23 742A 66 742B 69 742C E9 </pre>		<pre> ;ОСНАЧАЛА ПРОИЗВОДИТСЯ ИНИЦИАЛИЗАЦИЯ РАБОЧИХ ;ЯЧЕЕК ИМИТАТОРА НА ПЕРВОЙ СТРАНИЦЕ ОЗУ START: MVI A,0C3H STA 0 LXI H,BIOS SHLD 1 STA 5 LXI H,BDOS SHLD 6 XRA A STA 3 STA 4 ;ОЧИСТИТЬ БЛОКИ УПРАВЛЕНИЯ ФАЙЛОМ LXI H,40H FILLFCB: MVI M,20H INX H MOV A,H CPI 1 JNZ FILLFCB XRA A STA 5CH STA 6CH LXI SP,100H ;ТЕПЕРЬ ИМИТАТОР ДОС ПЕРЕМЕЩАЕТСЯ В ;ТРЕБУЕМОЕ МЕСТО ПАМЯТИ LXI H,1A0H ;НАЧАЛЬНЫЙ АДРЕС ;КОПИИ ИМИТАТОРА 0138 010074 LXI B,BDOS ;НАЧАЛЬНЫЙ АДРЕС ;ИМИТАТОРА БЛОС 0138 7E MOVCPM: MOV A,M 013C 02 STAX B 013D 23 INX H 013E 03 INX B 013F 7C MOV A,H 0140 FE03 CPI 3 ;ПЕРЕСЫЛКА ОТ 1A0H ;ДО 2FFH 0142 C23B01 JNZ MOVCPM ;ТЕПЕРЬ В СВОЕ МЕСТО ПЕРЕМЕЩАЕТСЯ ;ПРОГРАММА ПЕРЕСЫЛКИ .COM ФАЙЛА LXI H,160H LXI B,80H MOVCPM: MOV A,M STAX B INX H INX B MOV A,L OAOH JNZ MONOV JMP 60H ;ЗАПУСК ПРОГРАММЫ ;ПЕРЕМЕЩЕНИЯ ; .COM ФАЙЛА </pre>		<pre> 742D 4774 742F 5974 7431 6474 7433 4774 7435 4774 7437 4774 7439 6B74 743B 4774 743D 4774 743F 9174 7441 A074 7443 E374 7445 E774 </pre>		<pre> ;ТАБЛИЦА АДРЕСОВ ПОДПРОГРАММ ;НЕОБСЛУЖИВАЕМЫХ СИСТЕМНЫХ ВЫЗОВОВ ;ПРОСТО ИГНОРИРУЮТСЯ TACL: DW FVSR ;ФУНКЦИЯ 0 DW FCONIN ;ФУНКЦИЯ 1 DW FCONQ ;ФУНКЦИЯ 2 DW FRI ;ФУНКЦИЯ 3 DW FPO ;ФУНКЦИЯ 4 DW FLO ;ФУНКЦИЯ 5 DW FBCIO ;ФУНКЦИЯ 6 DW FGI0B ;ФУНКЦИЯ 7 DW FGI0V ;ФУНКЦИЯ 8 DW FRST ;ФУНКЦИЯ 9 DW FRCB ;ФУНКЦИЯ 10 DW FCGS ;ФУНКЦИЯ 11 DW FRVN ;ФУНКЦИЯ 12 ;ВОЗВРАТ В ПРИКЛАДНУЮ ПРОГРАММУ ;СЛЕДУЮЩЕЙ ФУНКЦИИ ИГНОРИРУЮТСЯ FSYR: ;ФУНКЦИЯ 0 FRI: ;ФУНКЦИЯ 3 FPO: ;ФУНКЦИЯ 4 FLO: ;ФУНКЦИЯ 5 FGI0B: ;ФУНКЦИЯ 7 FSI0B: ;ФУНКЦИЯ 8 ;ВОССТАНОВИТЬ СОДЕРЖИМОЕ РЕГИСТРОВ MRET: POP PSW MRET1: LHLD SPL SPHL LHLD DEL XCHG LHLD BCL MOV B,H MOV C,L LHLD HLL RET ;----- ; ФУНКЦИЯ 1 ;ВВОД КОДА СИМВОЛА С КЛАВИАТУРЫ FCONIN: POP PSW CALL CI MOV C,A CALL CO JMP MRET1 ;----- ; ФУНКЦИЯ 2 ;ВЫВОД НА ЭКРАН ДИСПЛЕЯ FCONQ: MOV C,E CALL CO JMP MRET </pre>					
<pre> ; "ОБРАЗ" ПРОГРАММЫ ПЕРЕМЕЩЕНИЯ .COM ФАЙЛА ; LDA I03H: LXI H,100H: LXI B,300H: MVI E,03 ; MOV D,A: LDAK B: MOV M,A: INX H: INX B: ; DCX D: MOV A,D: ORA E: JNZ CJK: JMP 100H </pre>		<pre> 7459 F1 745A CD03F8 745D 4F 745E CD09F8 7461 C34874 </pre>		<pre> 7447 F1 7448 2A4275 7449 F9 744C 2A3E75 744F EB 7450 2A4075 7453 44 7454 4D 7455 2A3C75 7458 C9 </pre>		<pre> 0160 0160 3A03012100 DB 3AH,03H,01H,21H,00H,01H 0166 0100031E00 DB 01H,00H,03H,1EH,00H,57H 016C 0A7723031B DB 0AH,77H,23H,03H,1BH,7AH 0172 B3C28C00C3 DB 0B3H,0C2H,8CH,00H,0C3H,03H </pre>		<pre> 7464 4B 7465 CD09F8 7468 C34774 </pre>		<pre> ;ПРОГРАММА "ИМИТАТОР ДОС" ORG BASE </pre>		<pre> 7400 ORG BASE </pre>	

```

; ФУНКЦИЯ 6
; НЕПОСРЕДСТВЕННЫЙ ВВОД-ВЫВОД
; С КОНСОЛИ
;-----
746B F1 FDCIO: POP PSW
746C 7B MOV A,E
746D FEFF CPI OFFH
746E CA7E74 JZ INPUT ;ВВОД СИМВОЛА
7472 FEFE CPI OFEH
7474 C8B74 JZ STATUS ;СТАТУС КЛАВИАТУРЫ
7477 4B MOV C,E
7478 CD09FB CALL CO
747B C34874 JMP MRET1
747E CD12FB INPUT: CALL CSTS
7481 B7 ORA A
7482 CA4874 JZ MRET1 ;ИЕТ СИМВОЛА
7483 CD03FB CALL CI
7488 C34874 JMP MRET1
748B CD12FB STATUS: CALL CSTS
748E C34874 JMP MRET1
;-----

```

```

; ФУНКЦИЯ 9
; ПЕЧАТЬ СТРОКИ ДО СИМВОЛА '*'
; НАЧАЛЬНЫЙ АДРЕС В DE
;-----
7491 EB FPRST: XCHG
7492 7E FPRST1: MOV A,M
7493 23 INX H
7494 4F MOV C,A
7495 FE24 CPI '*'
7497 CA4774 JZ MRET
749A CD09FB CALL CO
749D C39274 JMP FPRST1
;-----
; ФУНКЦИЯ 10
; ВВОД ДАННЫХ В БУФЕР, ПРЕДОСТАВЛЯЕМЫЙ
; ПРИКЛАДНОЙ ПРОГРАММОЙ. DE-НАЧАЛО БУФЕРА
; (DE)- ДЛИНА БУФЕРА. ПО ВОЗВРАТУ (DE+1)
; СОДЕРЖИТ КОЛИЧЕСТВО ЗАПИСАННЫХ В БУФЕР
; СИМВОЛОВ. В ПОДПРОГРАММЕ РЕАЛИЗОВАНО
; ТОЛЬКО РЕДАКТИРОВАНИЕ ПРИ ПОМОЩИ КЛАВИШИ
; *ШАГ НАЗАД* (КОД 08H).
;-----

```

```

74A0 F1 FRCB: POP PSW
74A1 EB XCHG
74A2 36 MOV D,M ;ДЛИНА БУФЕРА
74A3 23 INX H
74A4 23 INX H ;HL=НАЧАЛО БУФЕРА
74A5 0600 FRCB1: CALL CI
74A7 CD03FB MOV C,A
74AA 4F CPI 8H
74AB FE08 JZ BACKS ;*ШАГ НАЗАД*
74AD CAC974 CALL CO
74B0 CD09FB DCR D ;ДЛИНА-1
74B3 15 JNZ SKIP ;ЕЩЕ ЕСТЬ МЕСТО
74B4 C2B974 MVI A,08H ;КЛАВИША "BK"
74B7 3E0D SKIP: MOV M,A
74BA 23 INX H
74BB 04 INR B
74BC FE0D CPI 08H
74BE C2A774 JNZ FRCB1
74BF 2A3E73 LHL DEL
74C4 2D MOV M,B ;ЗАПОМНИТЬ СЧЕТЧИК
74C5 7D INX H
74C6 C34874 JMP MRET1
74C9 78 MOV A,B
74CA B7 ORA A ;ПРОВЕРКА НА
74CB CA7774 JZ FRCB1 ;НАЧАЛО БУФЕРА
74CE 14 INR D ;ДЛИНА+1
74CF 05 DCR B ;СЧЕТЧИК-1
74D0 23 DCX H ;КЛАВИТЕЛЬ-1
74D1 0E08 MVI C,8H ;СТЕРЕТЬ СИМВОЛ
74D3 CD09FB CALL CO ;С ЭКРАНА
74D6 0E20 MVI C,20H
74D8 CD09FB CALL CO
74DB 0E08 MVI C,8H
74DD CD09FB CALL CO
74E0 C3A774 JMP FRCB1
;-----

```

```

; ФУНКЦИЯ 11
; ВЕРНУТЬ СТАТУС КЛАВИАТУРЫ
;-----
74E3 F1 FCS: POP PSW
74E4 C3B774 JMP STATUS
;-----

```

```

; ФУНКЦИЯ 12
; ВЕРНУТЬ В HL НОМЕР ВЕРСИИ ДОС ...
;-----

```

```

; СЧИТАЕМ, ЧТО ДОС - CP/M VERS. 2.2
FRV: POP PSW
74E8 2A4273 LHL SPL
74EB F9 SPHL
74EC 2A3E73 LHL DEL
74EF EB XCHG
74F0 2A4073 LHL BCL
74F3 44 MOV B,M
74F4 4D MOV C,L
74F5 212200 LXI H,22H
74F8 C9 RET
;-----

```

```

; ИМИТАТОР МОДУЛЯ ПОДПРОГРАММ
; ВВОДА - ВЫВОДА (ИПЗВ)
;-----
7500 ORG BASE+100H
7500 C800FB JMP MONITOR ;В МОНИТОР
7503 C800FB BIOS: JMP MONITOR ;В МОНИТОР
7506 C312FB JMP CSTS ;СТАТУС КЛАВИАТУРЫ
7509 C203FB JMP CI ;ВВОД С КЛАВИАТУРЫ
750C C809FB JMP CO ;ВЫВОД НА ЭКРАН
750F C809FB LO ;ВВОД НА ПРИНТЕР
7512 C85573 JMP PUNCH ;ПЕРФОРАТОР
7515 C83773 JMP READER ;ФОТОСЧИТЫВАТЕЛЬ
7518 C83B73 JMP PRET ;ИГНОРИРОВАТЬ
751B C83B73 JMP PRET ;ИГНОРИРОВАТЬ
751E C83B73 JMP PRET ;ИГНОРИРОВАТЬ
7521 C83B73 JMP PRET ;ИГНОРИРОВАТЬ
7524 C83B73 JMP PRET ;ИГНОРИРОВАТЬ
7527 C83B73 JMP PRET ;ИГНОРИРОВАТЬ
752A C83B73 JMP PRET ;ИГНОРИРОВАТЬ
752D C83B73 JMP LSTST ;СТАТУС ПРИНТЕРА
7530 C83B73 JMP PRET ;ИГНОРИРОВАТЬ
;-----

```

```

; ПРИНТЕР ВСЕГДА НЕ ГОТОВ
LSTST: XRA A
RET
;-----
; ПЕРФОРАТОРА НЕТ (МОЖНО ИСПОЛЬЗОВАТЬ
; МАГНИТОФОН).
;-----

```

```

PUNCH: MOV A,C
RET
;-----
; ФОТОСЧИТЫВАТЕЛЯ НЕТ (МОЖНО ИСПОЛЬ-
; ЗОВАТЬ МАГНИТОФОН). 1AH - КОНЕЦ ФАЙЛА.
;-----

```

```

7537 3E1A READER: MVI A,1AH
7539 E67F ANI 7FH
;ПРОСТО ВОСВРАТ
PRET: RET
;-----
; ЯЧЕЙКИ ДЛЯ ВРЕМЕННОГО СОХРАНЕНИЯ
; РЕГИСТРОВ ПРИКЛАДНОЙ ПРОГРАММЫ
;-----

```

```

753C 0000 HLL: DW 0 ;HL
753E 0000 DEL: DW 0 ;DE
7540 0000 BCL: DW 0 ;BC
7542 0000 SPL: DW 0 ;SP
;-----

```

```

7544 END
;-----

```

Программа написана на языке ассемблера микропроцессора КР580ИК80А, занимает около 512 байт и позволяет имитировать функции ДОС по управлению дисплеем и клавиатурой.

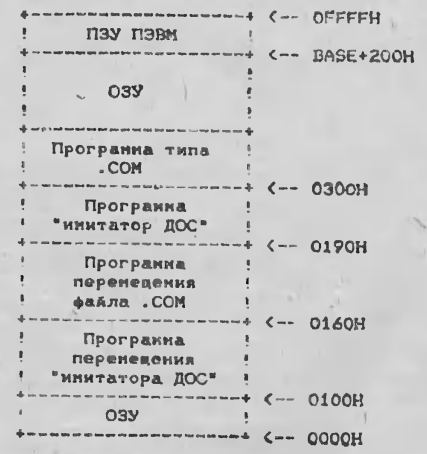


Рис. 1

Перед этим в ячейку памяти с адресом 103H необходимо поместить число страниц в прикладной программе (1 страница равна 256 байт).

На рис. 1 показано, как располагаются компоненты программы после ее загрузки в ОЗУ ПЭВМ. Будучи запущена в работу, программа «имитатор ДОС» производит перемещение этих компонентов в нужные адреса памяти (рис. 2) и запуск прикладной программы с адреса 100H.

-----	<-- OFFF7H
! ПЗУ ПЭВМ	-----
! "инициатор ИПБВ"	<-- BASE+200H
! "инициатор БДДС"	<-- BASE+100H
! ОЗУ	<-- BASE
! Программа типа .COM	-----
-----	<-- 0100H
! Программа перемещения файла .COM	-----
-----	<-- 0080H
! Рабочие ячейки для "инициатора ДДС"	-----
-----	<-- 0000H

Рис. 2.

Приведенные в таблице значения предназначены для ПЭВМ «Микро-80» [1], «Радио-86РК» [2] и «Микроша», однако простая модификация позволяет настроить программу для работы на ПЭВМ других типов. Необходимые для этого комментарии даны в тексте программы.

Телефон для справок: 235-83-82, Москва

ЛИТЕРАТУРА

1. Зеленко Г. В., Панов В. В., Попов С. Н. Радиолюбитель о микропроцессорах и микроЭВМ. Цикл статей//РАДИО. — 1982. — № 9—12, 1983. — № 2—12.
2. Горшков Д. В., Зеленко Г. В., Озеров Ю. В., Попов С. Н. Персональный радиолюбительский компьютер «Радио-86РК»//Радио—1986. — № 4—9.

Статья поступила 30 марта 1987 г.

УДК 681.306.

И. Г. Игнатова, Д. Л. Преснухин

ВЫБОР РЕЖИМОВ АДРЕСАЦИИ ПРИ ПРОГРАММИРОВАНИИ В СИСТЕМЕ КОМАНД МИКРОЭВМ ТИПА «ЭЛЕКТРОНИКА 60»

В большинстве случаев микропроцессорные системы программируются на языке ассемблера [1]. Ряд отечественных микроЭВМ, а также однокристальных микропроцессоров имеет систему команд микроЭВМ «Электроника 60». Одним из важных моментов при

программировании на языке ассемблера являются задания режимов адресации.

Даже хорошее знание программистом архитектуры машин не гарантирует легкости ориентировки во всех режимах адресации языка ассемблера. Квалифицированным специалистам было предложено авторами оценить набор различных сочетаний символов, используемых в языке ассемблера для адресации. Необходимо было вычеркнуть запрещенные сочетания символов. Ошибки допустили 92% опрошенных.

Сложность использования терминологии и символьных обозначений режимов адресации объясняется тем, что названия режимов адресации отражают особенности распознавания отдельных режимов в форматах команд на аппаратном уровне, а не суть самого режима с точки зрения программирования.

В языке ассемблера предусмотрено 12 режимов адресации [2]. По названиям режимов их можно разделить на две половины. Элементы одной половины отличаются от элементов другой только словом «косвенный», которое обозначает, что указателем исполнительного адреса является адрес, определяемый другими словами в названии режима адресации. Например, в режиме «автоинкрементный косвенный» такой указатель определяется по слову «автоинкрементный». Само это слово неявно содержит указания еще на одну косвенную адресацию. Термин «автоинкрементный косвенный» скорее ассоциируется с таким представлением, что это режим косвенной адресации с автоматическим прибавлением единицы к указателю исполнительного адреса. На самом деле это режим двойной косвенной адресации с автоматическим прибавлением единицы к «указателю указателя» исполнительного адреса.

Такого рода сложности в терминологии, а также то, что во всех справочных материалах и пособиях по программированию на языке ассемблера [2, 3] mnemonicская запись режимов адресации как набор символов соотносится только с названием режима, создают трудности в пользовании режимами адресации, особенно начинающим программировать на этом языке.

С точки зрения программиста, цель обращения по адресам (в адресное пространство микроЭВМ входят номера регистров и адреса ячеек ОЗУ) из команд программы может быть либо непосредственно к их содержанию — однократная операция выделения содержимого, либо затем, чтобы через это содержимое обратиться к содержанию других ячеек — многократное выделение содержимого, приводящее к понятию ранга адреса [4] или уровня доступа к операндам [5]. Кроме того, при обращении по адресу у программиста могут возникнуть две ситуации: либо адрес этот надо оставить без изменений, либо его как-то модифицировать.

Исходя из этого предлагается таблица, в которой используемые в языке ассемблера мнемоники режимов адресации размещены в соответствии с возможными уровнями доступа к данным и модификациями адресов.

Таблица

Уровень доступа	Без модификации		С модификацией	
	номера регистра	адреса ОЗУ	номера регистра	адреса ОЗУ
К данному через: исполнительный адрес	RN	MET; @ # MET	—	MET (RN)
адрес исполнительного адреса	(RN); @ RN	@ MET	(RN) +; — (RN)	@ MET (RN)
адрес адреса исполнительного адреса	—	—	@ (RN) +; @ — (RN)	—
К коду адреса*	—	# MET	—	—

* Символ # используется также для задания непосредственного данного из команды, например #300.

БИБЛИОТЕКА СТАНДАРТНЫХ ПРОГРАММ ДЛЯ БИС МИКРОЭВМ СЕРИИ K1816

Библиотека стандартных программ STEP (Standard Extension Programs), содержащая 67 модулей, написанных на языке ассемблера для БИС микроЭВМ серии K1816 [1], может использоваться самостоятельно и в комплексе с системой «Электроника Микросс-048» [2].

По функциональному назначению и объему своих элементов библиотека STEP разнородна и может быть разделена на четыре группы программ: обмена и сдвигов; конвертирования; плавающей арифметики; вычисления специальных функций.

Первая группа — наиболее простые по реализации программы, осуществляющие: сдвиги информации в регистрах R0...R7 нулевого банка с учетом и без учета содержимого бита переноса C; перемещение информации (до 8 байт) из регистров нулевого банка в ОЗУ и обратно; перемещение информации из нулевого банка регистров в первый и обратно (до 4 байт из различных регистров). Кроме того, имеются программы всевозможных проверок содержимого регистра нулевого банка, которые могут использоваться как программы контроля порядка числа в плавающем формате, проверки на равенство нулю мантиссы числа и т. д.

Вторая группа — программы обработки чисел: перевода 16-разрядного целого числа в 32-разрядное число с плавающей точкой и обратно; нормализации 32-разрядного числа с плавающей точкой; нахождения целой части 32-разрядного числа в плавающем формате; перевода градусной меры угла в радианную и обратно (32-разрядные числа с плавающей точкой); упаковки и распаковки 32-разрядных чисел с плавающей точкой и т. д.

Третья группа — программы сложения, вычитания, деления, умножения 32-разрядных чисел с плавающей точкой; сложения, вычитания, деления, умножения чисел в смешанном формате (одно из чисел 32-разрядное с плавающей точкой, другое — 16-разрядное с фиксированной точкой), сложения и вычитания в плавающем формате с двойной точностью; увеличения и уменьшения 32-разрядного числа с плавающей точкой на единицу.

Четвертая группа — программы вычисления элементарных и специальных функций: вычисления квадратных корней, экспоненты, логарифма, гиперболических функций, тригонометрических функций, показательной функции, обратных тригонометрических функций, факториала. Аргумент и результат вычисления специальных функций — 32-разрядное число с плавающей точкой. Исключение составляет лишь программа вычисления факториала, исходным значением для которой является 16-разрядное число с фиксированной точкой.

Во всех группах программ библиотеки в качестве рабочей памяти используются регистры нулевого и первого банков, аккумулятор, флаги F0 и F1, бит переноса C и, кроме того, для четвертой группы до 40 байт ОЗУ.

Иерархическая структура вложений программ STEP имеет шесть уровней. Например, цепочка вызовов содержит программы: вычисления специальных функций (например, экспоненты, плавающей арифметики (например, деления), нормализации чисел, контроля порядка. Программы плавающей арифметики и вычисления специальных функций, как правило, имеют также подуровни вложений.

Основные характеристики программ приведены в таблице.

Таблица поможет выбрать нужную мнемонику режима адресации. Ядром мнемоники является номер РОНа или адрес ОЗУ, т. е. элемент адресного пространства архитектуры, задаваемый средствами языка. Вокруг ядра строится оформление символами, определяющими уровень доступа к данному по адресу, задаваемому ядром, а также возможную модификацию содержимого ядра.

Одновременно адрес ОЗУ и номер регистра могут задаваться только в режиме, определяющем индексную модификацию, причем ядро мнемоники здесь — адрес ОЗУ, а содержимое регистра считается индексом. Номер регистра задается в скобках после адреса ОЗУ. Индексная модификация может применяться с уровнем доступа к данному через его адрес, а также к данному через адрес его адреса. В первом случае ядро мнемоники задается как адрес ОЗУ, после которого в скобках указывается номер регистра — держателя индекса. Во втором случае в оформление ядра добавляется @ — символ использования косвенности адресации.

В мнемониках режимов адресации, где ядро — адрес ОЗУ, косвенность используется только один раз, и в этих режимах она задается символом @. В данном случае модификация может быть только индексная. Средства языка ассемблера позволяют обращаться непосредственно не только к содержимому, но и к адресам ОЗУ без каких-либо модификаций. В мнемонике такого режима к ядру добавляется символ #.

Для доступа к данному через адрес ОЗУ в языке ассемблера возможно использование двух мнемоник: просто задания ядра без оформления типа MET либо сочетания символов @ # MET. Символы @ и # нейтрализуют действие друг друга.

В мнемониках режимов адресации, где ядро — номер регистра, задание косвенности допускается символом @ или скобками, в которые заключаются ядро мнемоники. Для обеспечения уровня доступа к данному через адрес его адреса, т. е. для однократного использования косвенности (когда ядро — номер РОНа) возможны равносильные мнемоники (RN) и @RN.

Одновременное присутствие в мнемонике символов @ и () указывает на двукратное использование косвенности. Это соответствует уровню доступа к данному через адрес адреса исполнительного адреса.

Двукратное использование косвенности в режимах, где ядро мнемоники — номер регистра, возможно только в сочетании с автоинкрементной или автодекрементной модификацией.

Модификации автоинкрементную и автодекрементную возможно также использовать в мнемониках режимов с ядром — номером регистра и с однократным использованием косвенности, т. е. на уровне доступа к данному через адрес указателя исполнительного адреса.

Для доступа к данному через его адрес (в случае, если адрес — это номер РОНа) задается ядро мнемоники без всякого оформления.

Телефон для справок: 534-54-71, Москва

ЛИТЕРАТУРА

1. Обучение специалистов применению и разработке микропроцессоров и микроЭВМ/И. Г. Гусев, В. М. Немчинов, А. Г. Филиппов, И. И. Шагурин//Микропроцессорные средства и системы. — М.: 1984. — № 2. — С. 88—90.
2. Сингер М. Программирование на языке ассемблера и организация машины РДР-11. — М.: Мир, 1984. — 272 с.
3. Малые ЭВМ и их применение /Под ред. Б. Н. Наумова. — М.: Статистика, 1980. — 231 с.
4. Энциклопедия кибернетики в 2 т. Т. I. — Киев: Изд-во АН УССР, 1975.
5. Проектирование специализированных информационно-вычислительных систем/Под ред. Ю. М. Смирнова. — М.: Высшая школа, 1984. — С. 297—323.

Статья поступила 4 февраля 1987 г.

Основные характеристики программ

Номер программы	Наименование программы	Имя программы	Время исполнения, мс	Объем памяти, байт	Абсолютная погрешность	Номера используемых программ
1—11	Сдвиги вправо, влево на 1 разряд 24- и 23-разрядной мантиссы с и без очистки бита переноса	SR, SR0, SL, SLO, SR02, SR1, SL02, SL1, PC, LC, LI	0,4	40	—	—
12	Проверка на равенство 0 24-разрядной мантиссы	PR	0,2	26	—	—
13—17	Обмен содержимым (46) между регистрами одного и двух банков	OBM, IN1B, IN01, IN0B, IN10	0,6	34	—	—
18—22	Перемещение массива (1—8 байт) из ОЗУ в регистры и обратно	OR, RO, MARG, NIN1, ONIN	0,8	98	—	—
23	Контроль порядка	ORD1	0,2	20	—	—
24, 25	Перевод числа из плавающего формата в фиксированный и обратно	FIX, FLOT	8	120	—	7,35
26—30	Упаковка, распаковка, округление, изменение знака числа на противоположный, выбор числа с большим порядком	UPC, RPC, DIL, ZNAC, MX0	0,2	56	—	—
31	Перевод числа в дополнительный код	DORK	2	90	—	—
32	Нахождение целой части числа	INTD	1,4	95	—	—
33—34	Перевод из градусной меры в радианную и обратно	GRAD, RAD	8	40	2 ⁻²³	42
35	Нормализация числа в плавающем формате	NORM	0,4	140	—	12, 23
36, 37	Увеличение, уменьшение числа с плавающей точкой на 1	FIN, DEKR	6	240	2 ⁻²³	35, 12, 23, 31
38—41	Сложение, вычитание чисел с плавающей точкой и в смешанном формате	ADI, SUB, ADF, SUBF	9	300	2 ⁻²³	2,5 12, 13, 23, 25, 31, 35
42—46	Умножение, деление чисел с плавающей точкой и в смешанном формате	FMUN, FDIV, MF, FDIF, PR2	9	380	2 ⁻²³	9—13, 15—17, 23, 25
47—49	Сложение, вычитание чисел с двойной точностью	ADD, AD, SUBD	11	360	2 ⁻²³	—
50	Вычисление факториала	FACT	18	105	—	25, 42, 35
51	Вычисление квадратного корня	SQR	24	380	2 ⁻²²	9, 10, 12, 13, 16, 23, 35, 43
52—56	Вычисление экспоненты, гиперболических функций	EXP, SH, CH, TH, CTH	30	520	2 ⁻¹⁷	18, 19, 24, 35, 38, 42, 43
57—60	Вычисление тригонометрических функций	SIN, COS, TH, CTH	20	490	2 ⁻²²	38, 42, 43
61—64	Вычисление обратных тригонометрических функций	ARS1, ARS2, ATN1, ATN2	20	450	2 ⁻¹⁷	38, 42, 43
65—67	Вычисление логарифма и показательной функции	LN, LOG, FPI	22	540	2 ⁻¹⁷	25, 35, 38, 42, 43

Модульность, структурность программ библиотеки обеспечивают прозрачность логической структуры каждого ее элемента и существенно сокращают объем необходимой для ее размещения памяти.

Для программ плавающей арифметики были приняты алгоритмы из работ [3—5]. Однако эти алгоритмы направлены на использование регистров с достаточно большим числом разрядов, в то время как БИС микроЭВМ серии K1816 имеют лишь 8-разрядные регистры, что привело к необходимости предварительно модифицировать алгоритмы с учетом имеющихся ресурсов и возможностей соответствующего ассемблера.

Для вычисления квадратного корня применялась итерационная формула Герона, для факториала — формула рекурсии, для остальных функций — аппроксимация Паде с коэффициентами 3-го и 4-го порядков и формулы приведения. Абсолютная погрешность вычислений в среднем составила 10⁻⁶.

Упакованный и распакованный форматы чисел с плавающей и фиксированной точкой соответствуют стандартным форматам, принятым в микроЭВМ серии K1816. Исходные данные для всех программ библиотеки записаны в регистры нулевого банка регистров. Исключение составляют программы сложения и вычитания чисел с двойной точностью, в которых данные записаны

в 16 регистров (используются нулевой и первый банки). Кроме того, для части программ исходными данными являются также содержимое бита переноса C и аккумулятора A (программы нормализации чисел с плавающей точкой, обмена информацией между регистрами и ОЗУ). Если исходным данным или результатом является одно 4-байтное число с плавающей точкой, то оно заносится в регистры R4...R7 нулевого банка, если два, то первое — в R0...R3, второе — в R4...R7 нулевого банка. Число в фиксированном формате занимает регистры R2 и R3.

При нормальном завершении программы регистр A (аккумулятор) содержит 0. В случае аварийного завершения программы в A заносится диагностическая информация, при этом исходная информация, как правило, теряется.

В качестве аварийных выходов рассматривались ненормализованные исходные данные, исчезновение и переполнение порядка в процессе вычислений, деление на 0.

Программы отлаживались с помощью комплекса кросспрограмм «Электроника Микросс-048» [2], под управлением ОС ДВК. В состав комплекса входил кросс-ассемблер, кроссредактор связей, программно-логическая модель (отладчик).

АДР.	КОД	МНЕМОНИКА	АДР.	КОД	МНЕМОНИКА	АДР.	КОД	МНЕМОНИКА
0002	00FD	MOV A,R0	0002	00FD	MOV A,R0	0002	00FD	MOV A,R0
ИМЯ	R0	R01	ИМЯ	R0	R01	ИМЯ	R0	R01
R0	0081	0000	R0	0000	0000	R0	0000	0000
R1	0040	0000	R1	0000	0000	R1	0040	0000
R2	0000	0000	R2	0000	0000	R2	0000	0000
R3	0000	0000	R3	0000	0000	R3	0000	0000
R4	0081	0000	R4	0081	0000	R4	00FF	0000
R5	0040	0000	R5	0040	0000	R5	0040	0000
R6	0000	0000	R6	0000	0000	R6	0000	0000
R7	0000	0000	R7	0000	0000	R7	0000	0000
TEST0	=0		TEST0	=0		TEST0	=0	
TEST1	=0		TEST1	=0		TEST1	=0	
FLAG0	=0		FLAG0	=0		FLAG0	=0	
FLAG1	=0		FLAG1	=0		FLAG1	=0	
TIMER	=0000		TIMER	=0000		TIMER	=0000	
FLAG T	=0		FLAG T	=0		FLAG T	=0	
REGISTER A	=0		REGISTER A	=0		REGISTER A	=0	

АДР.	КОД	МНЕМОНИКА	АДР.	КОД	МНЕМОНИКА	АДР.	КОД	МНЕМОНИКА
*****	СТЕК ПУСТОЙ	*****	*****	СТЕК ПУСТОЙ	*****	*****	СТЕК ПУСТОЙ	*****
00A6	0093	RETR	0008	0093	RETR	00AC	0093	RETR
ИМЯ	R0	R01	ИМЯ	R0	R01	ИМЯ	R0	R01
R0	0086	0002	R0	0000	0082	R0	0001	0001
R1	0000	0020	R1	0000	0020	R1	0020	0020
R2	0000	0000	R2	0000	0000	R2	0000	0000
R3	0000	0000	R3	0000	0000	R3	0000	0000
R4	0081	0000	R4	0081	0000	R4	00FF	0000
R5	0040	0000	R5	0040	0000	R5	0040	0000
R6	0000	0000	R6	0000	0000	R6	0000	0000
R7	0000	0000	R7	0000	0000	R7	0000	0000
TEST0	=0		TEST0	=0		TEST0	=0	
TEST1	=0		TEST1	=0		TEST1	=0	
FLAG0	=0		FLAG0	=0		FLAG0	=0	
FLAG1	=0		FLAG1	=0		FLAG1	=0	
TIMER	=0000		TIMER	=0000		TIMER	=0000	
FLAG T	=0		FLAG T	=0		FLAG T	=0	
REGISTER A	=0		REGISTER A	=1		REGISTER A	=2	

Рис. 1.

Рис. 2

Рис. 3

Фрагмент трассировочной таблицы, полученной с помощью программно-логической модели PLM 48, при нормальном и аварийных завершениях программы приведен на рис. 1—3. Дано содержимое регистров нулевого и первого банков, флагов, аккумулятора и т. д. до и после завершения программы деления 32-разрядных чисел с плавающей точкой. В регистрах нулевого банка R0...R3 расположен делитель, в R4...R7 — делимое и частное от деления. В одном из рассмотренных случаев (см. рис. 1) в аккумуляторе содержится 0, что соответствует нормальному завершению программы, в двух других (см. рис. 2 и 3) содержимое аккумулятора отлично от 0, т. е. выполнение программы закончилось

аварийно. В первом случае (см. рис. 2) делитель равен нулю, в аккумуляторе 1, во втором (см. рис. 3) — переполнение разрядной сетки — в аккумуляторе 2.

Рассмотрим пример использования библиотеки в микропроцессорной информационно-управляющей вычислительной системе. Закономерности функционирования объекта управления часто задаются в виде системы дифференциальных уравнений.

Используя значения входной функции $Y(t)$ лишь в предшествующие моменты и второй полином Ньютона, можно записать для интерполирующего полинома 3-й степени формулу численного дифференцирования в виде [6]

$$Y'_k(t) = \frac{1}{T} \left(\frac{11}{6} Y_k - 3Y_{k-1} + 1,5 Y_{k-2} - \frac{1}{3} Y_{k-3} \right),$$

где T — период квантования входных данных; Y_k , Y_{k-1} , Y_{k-2} , Y_{k-3} — входные величины в моменты k , $k-1$, $k-2$, $k-3$.

Программа вычисления $Y'_k(t)$ по приведенной формуле содержит операции: четыре умножения, одно деление, три сложения (вычитания). Командам вызова подпрограмм сложения (ADI), деления (FDIV), умножения (FMUN) предшествует загрузка исходных данных (32-разрядных чисел) в регистры нулевого банка. Все необходимые константы (11/6; 3; 1,5; 1/3) размещаются в регистрах перед выполнением соответствующих программ умножения, т. е. нет необходимости использовать ОЗУ для хранения констант.

Имена подпрограмм должны быть объявлены внешними и занесены в список имен команды EMT основной программы. Вызываются подпрограммы командой CALL (имя подпрограммы).

Для отладки программы можно использовать кросс-систему Микросс-048, установленную на ДВК-2. При этом сборку программы рекомендуется проводить следующим образом. Кросс-систему и монитор установить на системный вход дисковод, а необходимые подпрограммы библиотеки, основную программу и редактор связей L48 — на другой вход. Время расчета одного значения $Y'_k(t)$ приблизительно 4 с. Программа содержит 40 операторов. Без подпрограмм библиотеки она имела бы около 1000 операторов ассемблера, что существенно увеличило бы сроки разработки.

Общий объем библиотеки составляет приблизительно 8 тыс. команд ассемблера. Библиотека является открытой системой, расширяемой на уровне кроссредактора связей.

Библиотека сдана в отраслевой фонд алгоритмов и программ. Пользователь может получить исходные, объектные и загрузочные модули на гибком магнитном диске и полную техническую документацию сопровождения, включающую алгоритмы всех стандартных программ и систему тестов.

Телефон для справок: 434-94-65, Москва

ЛИТЕРАТУРА

1. Кобылинский А. В., Липовецкий Т. П. Однокристальные микроЭВМ серии К1816//Микропроцессорные средства и системы. — 1986. — № 3. — С. 10.
2. Белов А. М., Иванов Е. А., Муренко Л. Л. Комплекс кросспрограмм «Электроника Микросс»// Микропроцессорные средства и системы. — 1986. — № 3. — С. 27.
3. Кнут Д. Искусство программирования для ЭВМ. т. 2. Получисленные алгоритмы. — М.: Мир, 1977. — 724 с.
4. Люк Ю. Специальные математические функции и их аппроксимации. — М.: Мир, 1980. — 509 с.
5. Байков Б. Д., Смолов В. Б. Специализированные процессоры: Итерационные алгоритмы и структуры. — М.: Радио и связь, 1985. — 288 с.
6. Микропроцессоры: Кн. 2. Средства сопряжения. Контролирующие и информационно-управляющие системы/Под ред. Л. Н. Преспухина. — М.: Высшая шк., 1986. — 383 с.

Статья поступила 12 февраля 1987 г.

Опечатка в № 4, 1987 г.

На с. 19 в табл. «Средства и трудозатраты на разработку» в последней строке время разработки и внедрения следует читать — 2 дня.

УСКОРЕНИЕ СЧИТЫВАНИЯ С ДИСКА В ОПЕРАЦИОННОЙ СИСТЕМЕ CP/M-80

Предлагаемая программа TURBO предназначена для пользователей операционной системы CP/M-80-2x [1] (или эквивалентных) и обеспечивает ускорение считывания с дисков одинарной плотности примерно в два раза. Ускорение достигается за счет изменения логики считывания секторов и использования промежуточного буфера. Алгоритм считывания аналогичен процедуре деблокировки секторов, применяемой в CP/M при организации доступа к диску, у которого физический размер сектора превышает логический (128 байт). Запись на диск через промежуточный буфер в данном случае не применяется, поскольку при этом не удастся получить выигрыша во времени (к тому же запись на диск используется значительно реже считывания).

Программа загружается с диска как обычная команда и, перемещаясь, располагается непосредственно перед процессором консольных команд (CCP), определяя его положение в памяти по адресу входа в базовую дисковую операционную систему (BDOS) и изменяя соответственно с ним свои рабочие адреса. Затем программа изменяет адрес входа в BDOS, по которому определяется верхняя граница области программ пользователя (рис. 1). В результате после загрузки программы доступный пользователю объем памяти уменьшается на 4К байт (2К байт — CCP и 2К байт — TURBO вместе с буфером для считывания дорожки). Программа является резидентной и удаляется из памяти только при полной перезагрузке системы. Сразу после загрузки и перемещения TURBO изменяет адреса в таблице переходов базовой системы ввода-вывода (BIOS) и в дальнейшей работе взаимодействует только со стандартными процедурами, вследствие чего является машинно-независимой. Кроме того, TURBO учитывает особенности работы других перемещающихся программ, таких как DDT, SID, XSUB, DESPOOL и других, — все они могут работать совместно с TURBO. Однако, в отличие от XSUB и DESPOOL, TURBO осуществляет

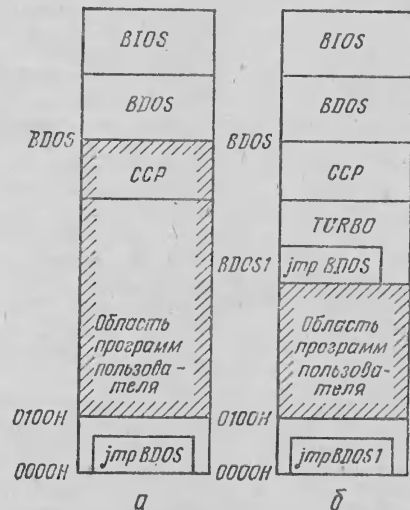


Рис. 1. Размещение операционной системы (а) и программы TURBO (б) в оперативной памяти

реальную перезагрузку системы с последующим восстановлением своих функций.

Операционная система CP/M-80 позволяет работать с дисками любого формата, но исторически сложилось так, что основная масса программного обеспечения для нее поставлялась на 8-дюймовых гибких дисках с одинарной плотностью записи (77 дорожек по 26 128-байтных секторов). При таком формате порядок следования секторов при записи-чтении дорожки используется следующий: 1, 7, 13, ..., 10, 16, 22, с шагом в 6 секторов. Столь большой шаг (возможно, вынужденный для старых компьютеров) в настоящее время приводит к значительному простоя процессора во время записи-чтения последовательных файлов. Изменение формата записи для дисков одинарной плотности не является целесообразным, так как при этом затрудняется обмен программным обеспечением, к тому же оно требует изменений в BIOS и, следовательно, является машинно-зависимым.

В программе TURBO считывание организовано с учетом минимизации размера буфера и времени доступа к произвольному сектору диска. Для этого за один прием считывается только половина дорожки: отдельно проходы по нечетным и по четным секторам. Такой алгоритм хорошо согласуется с приведенным выше логическим порядком следования секторов для дисков одинарной плотности. Кроме того, считывание через один сектор практически реализуемо на любом современном компьютере, тогда как посекторное считывание с использованием только стандартных процедур BIOS на большинстве компьютеров организовать невозможно. Упрощенная блок-схема программы TURBO приведена на рис. 2. Как видно из схемы, процедура считывания сектора начинается с проверки: была ли раньше считана группа секторов, в которую входит данный сектор. Если да, то считывание производится из буфера, при этом используется подпрограмма быстрой пересылки (рис. 3), которая на пересылку одного байта затрачивает 21 такт вместо 39 тактов по стандартной пересылке. Если диск имеет другой формат записи или при считывании в буфер была обнаружена ошибка, то управле-

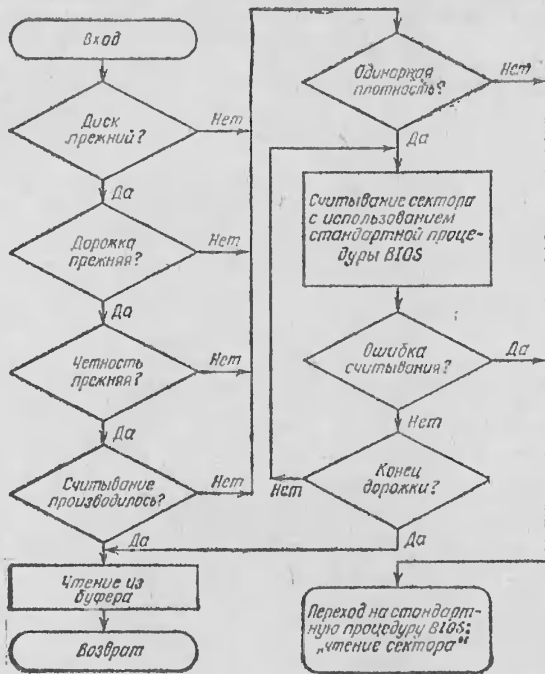


Рис. 2. Блок-схема работы программы

Входные параметры

DE - начальный адрес сектора в буфере
(DMAADR) - начальный адрес перемещения

MVI C, 52 ; инициализация счетчика байтов
; (128/4)
LXI H, 0
DAD SP
SHLD SAVESTK ; сохранить указатель стека
XCHG ; в указателе стека - начальный
; адрес сектора в буфере
SPHL ; в HL - начальный адрес
; перемещения

SECMOV: POP D ;
MOV M, E ;
INX H ;
MOV M, D ;
INX H ;
POP D ; цикл перемещения сектора -
MOV M, E ; 4 байта за один проход
INX H ;
MOV M, D ;
INX H ;
DCR C ;
JNZ SECMOV ;
LHLD SAVESTK
SPHL ; восстановить указатель стека
XRA A ; нормальное окончание (Z=1)
RET

Рис. 3. Подпрограмма быстрой пересылки

ние передается стандартной процедуре BIOS «считывание сектора». Последнее условие позволяет успешно считывать сектор, если в группе секторов имеются сбойные секторы. Дамп программы приведен на рис. 4. При считывании последовательных файлов за один оборот диска производится считывание нечетной (или четной) половиной дорожки (13 секторов), а затем выполняется пересылка секторов, за время которой диск успевает повернуться на 10...18 секторов (в зависимости от быстродействия процессора). Поэтому при считывании следующей группы секторов для компенсации поворота диска к номеру сектора прибавляется смеще-

```

0100 C3 22 01 12 F7 78 02 00 52 45 51 55 49 52 45 53
0110 20 43 50 2F 4D 20 56 45 52 53 49 4F 4E 20 32 2E
0120 32 24 0E 0C CD 05 00 FE 22 D2 34 01 0E 09 1F 08
0130 01 C3 05 00 C3 3B 01 42 44 4F 53 2A 06 00 11 37
0140 01 2E BA EB 01 00 04 1A E6 5F BE C2 56 01 23 13
0150 05 C2 47 01 0E 08 79 87 C2 78 01 0E 09 11 63 0F
0160 C3 05 00 43 41 4E 27 54 20 52 45 4C 4F 43 41 54
0170 45 20 54 55 52 42 4F 24 3A 03 01 F5 2A 05 01 E5
0180 21 04 01 3A 07 00 91 32 07 01 86 57 1E 00 C1 C5
0190 D5 21 00 02 78 B1 CA A1 01 0B 7E 12 13 23 C3 94
01A0 01 D1 C1 D5 E5 3A 07 01 67 7B B1 CA C7 01 0B 7B
01B0 E6 07 C2 BA 01 E3 7E 23 E3 6F 7D 17 6F D2 C3 0E
01C0 1A 84 12 13 C3 A9 01 D1 E1 E9 CD 0F 1C AF 32 DA
01D0 11 3A 4F 30 B7 CC 1B 1C CD 06 16 3A 05 30 FE 01
01E0 C2 01 07 3A 4F 30 B7 CC 21 1C 21 DA 11 34 CD 06
01F0 16 3A 05 30 FE 04 C2 01 07 3A 09 30 FE 2C CA D8
    
```

Рис. 4. (Начало)

```

0200 AF CD 81 F7 22 1A F8 21 6B F9 CD 8C F7 3E 15 CD
0210 81 F7 22 30 F8 21 2B F8 CD 8C F7 3E 18 CD 81 F7
0220 22 1D F8 21 32 F8 CD 8C F7 3E 1B CD 81 F7 22 20
0230 F8 21 4D F8 CD 8C F7 3E 1E CD 81 F7 22 23 F8 21
0240 5D F8 CD 8C F7 3E 21 CD 81 F7 22 9B F8 21 6E F8
0250 CD 8C F7 3E 24 CD 81 F7 22 26 F8 21 A4 F8 CD 8C
0260 F7 3E 27 CD 81 F7 22 29 F8 21 9D F8 CD 8C F7 2A
0270 06 00 22 07 F8 26 F8 22 7D F8 F1 32 BC F8 C3 00
0280 00 2A 01 00 85 BF 23 5E 23 56 EB C9 EB 72 2B 73
0290 C9 2A 2A 2A 2A 20 54 55 52 42 4F 20 42 59 20
02A0 41 4C 45 58 20 47 4F 53 54 45 56 20 28 32 37 2E
02B0 30 37 2E 38 36 29 20 2D 20 46 41 53 54 20 44 49
02C0 53 4B 20 52 45 41 44 49 4E 47 20 28 46 4F 52 20
02D0 53 49 4E 47 4C 45 20 44 45 53 49 54 59 29 20 56
02E0 45 52 53 49 4F 4E 20 2D 20 32 2E 31 20 2A 2A 2A
02F0 2A 2A 00 00 00 00 00 00 00 00 00 00 00 00 00

```

```

0300 00 00 00 00 00 00 C3 00 00 54 55 52 42 4F 20 2D
0310 20 32 37 2E 30 37 2E 38 36 C3 00 00 C3 00 00 C3
0320 00 00 C3 00 00 C3 00 00 C3 00 00 AF 32 73 F9 C3
0330 00 00 21 77 F9 7E B9 CA 3F F8 71 AF 32 73 F9 CD
0340 1C F8 E5 5E 23 56 13 1A 32 74 F9 E1 C9 21 78 F9
0350 7E B9 CA 1F F8 71 AF 32 73 F9 C3 1F F8 21 79 F9
0360 79 AE 0F 71 D2 22 F8 AF 32 73 F9 C3 22 F8 60 69
0370 22 75 F9 21 7A F9 7E B7 C2 9A F8 34 21 06 F8 3A
0380 07 00 BC DA 9A F8 22 06 00 3E C3 32 00 F0 32 00
0390 F8 21 00 00 22 01 F0 22 01 F8 C3 00 00 AF 32 73
03A0 F9 C3 28 F8 21 74 F9 7E FE 07 C2 25 F8 2B 7E B7
03B0 C2 FC F8 CD 29 F9 06 0D 21 72 F9 3E 12 86 FE 1A
03C0 DA C5 F8 D6 1A 77 3A 79 F9 E6 01 EE 01 86 3C 4F
03D0 79 CD 3A F9 C5 CD 21 F9 CD 25 F8 C1 B7 C2 56 F9
03E0 0C 0C CD 4E F9 05 C2 D0 F8 3E 01 32 73 F9 CD 29
03F0 F9 2A 75 F9 EB 3A 79 F9 4F CD 21 F9 CD 37 F9 0E

```

```

0400 20 21 00 00 39 22 7B F9 EB F9 2A 75 F9 D1 73 23
0410 72 23 D1 73 23 72 23 0D C2 0D F9 2A 7B F9 F9 AF
0420 C9 D5 CD 22 F8 C1 C3 9A F8 3A 77 F9 4F CD 1C F8
0430 3A 78 F9 4F C3 1F F8 3A 79 F9 3D 0F 0F 57 1E 00
0440 D2 45 F9 1E 80 E6 3F 57 21 7D F9 19 EB C9 79 FE
0450 1B D8 D6 1A 4F C9 CD 29 F9 3A 79 F9 4F CD 22 F8
0460 2A 75 F9 44 4D CD 9A F8 C3 25 F8 AF 32 7A F9 C3
0470 19 F8 00 00 00 00 00 00 12 48 49 21 24 84 92 12
0480 48 49 21 24 84 92 0A 44 00 00 00 00 00 00 00
0490 00 00 00 00 00 00 00 00 00 00 00 00 02 08 42
04A0 40 21 08 49 02 24 24 22 04 04 92 40 92 08 24 20
04B0 20 80 11 21 08 84 91 12 01 08 00 24 08 91 22 40
04C0 20 20 00 91 21 22 40 1A 1A 1A 1A 1A 1A 1A 1A
04D0 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A
04E0 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A
04F0 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A

```

Рис. 4. Текст программы TURBO

ние, которое должно быть четным, чтобы не изменить четность группы секторов (ячейка с адресом 103H на рис. 4). Значение смещения (12H) подобрано для микроЭВМ CM 1800 [2] (эквивалентная тактовая частота процессора 1,34 МГц), что дает ускорение считывания в 1,8 раза. Для компьютера с другим быстродействием значение константы пользователь может подобрать сам.

Программа TURBO была опробована на микроЭВМ CM 1800 и IF-800 (Япония); на последнем компьютере проверялась совместимость с дисками разных форматов. Программу можно получить, набрав текст, изображенный на рис. 4 (например, используя DDT), и записав его с помощью команды: SAVE 4 TURBO.COM. Можно также обратиться к автору (естественно, со своим диском) в Институт нитроскопии (г. Москва) и получить копии программы и утилитов для оптимизации быстродействия (а для пользователей CM 1800, кроме того, адаптированную автором версию CP/M-80), тел. 235-59-18.

ЛИТЕРАТУРА

1. Hughes L. E. System programming under CP/M-80, Reston: Reston, 1983.
2. Прохоров Н. Л., Ландау И. Я. МикроЭВМ CM 1800 и ее программное обеспечение // Микропроцессорные средства и системы. — 1984. — № 2. — С. 28.

Статья поступила 26 января 1987 г.

УДК 681.3.06

С. Ф. Добриневский

ДРАЙВЕР ГРАФОПОСТРОИТЕЛЯ ЭМ-7042АМ ДЛЯ ОС ДВК

Графопостроитель ЭМ-7042АМ (и командно совместимые с ним устройства) широко используется для вывода графической информации в измерительно-вычислительных комплексах на основе ЭВМ «Электроника 60», «Электроника НЦ» с операционной системой ОС ДВК (Рафос) [1]. Включение в систему соответствующего драйвера позволяет обращаться к графопостроителю системными средствами [2]. Тем самым графопостроитель становится доступным для программ, написанных на любых языках высокого уровня, поддерживаемых системой, причем в полной мере используются его возможности не только как устройства вывода, но и как кодировщика графической информации.

Предполагается, что графопостроитель подключен к ЭВМ с помощью стандартного устройства последовательного обмена. Адреса регистров приемника и передатчика, а также векторы прерываний определяются константами в начале текста драйвера и могут быть изменены в соответствии с имеющейся аппаратурой.

Обмен информацией между графопостроителем и ЭВМ осуществляется по прерываниям. Передача информации на графопостроитель завершается по исчерпанию счетчика слов элемента очереди к драйверу. При приеме ситуация несколько сложнее. Графопостроитель передает в ЭВМ координаты текущей точки и состояние нищущего инструмента (или координаты постоянного окна) в виде четырех целых чисел, после каждого из которых следуют символы возврата каретки и перевода строки. Если принять обычную схему ввода с завершением по концу строки, то окончание обработки текущего элемента очереди и переход к следующему (т. е. к очередному целому числу) не успевают произойти до прихода очередной посылки, в то время как на ЭВМ при работе с графопостроителем ЭМ-7042АМ накладывается требование постоянной готовности к приему информации. Поэтому обработка элемента очереди завершается только после получения четвертой строки, после

Исходный текст драйвера на макроассемблере:

```

TITLE 'ПРОГРАММА ОБСЛУЖИВАНИЯ ДИСКОВОДА НГМД-7012'
;ПОРТЫ ДИСКОВОДА
RADISK EQU 40H
RBDISK EQU 41H
PCDISK EQU 42H
PRDISK EQU 43H
;КОДЫ ФУНКЦИЙ ДИСКОВОДА
RBFUN EQU 0
RDBFUN EQU 2
WRITF EQU 4
READF EQU 6
RROSF EQU 10
;МАСКИ СИГНАЛОВ НГМД
RVU EQU 10H
MERR EQU 20H
MZAV EQU 40H
MZR EQU 80H
;КОДЫ ИМПУЛЬСОВ УПРАВЛЕНИЯ ДИСКОВОДОМ
MPUSK EQU 1
PARAL EQU 3
USTDSK EQU 5

```

```

ORG 5000H

MUI A, 0AH ;ИНИЦИАЛИЗАЦИЯ ИНТЕРФЕЙСА НГМД
OUT PRDISK

```

```

READ: ;ПОДПРОГРАММА ЧТЕНИЯ СЕКТОРА В ОЗУ
MUI C, READF
CALL SETFUNC ;УСТАНОВКА КОДА КОМАНД ЧТЕНИЯ СЕКТОРА
CALL WIZAV ;ОЖИДАНИЕ ГОТОВНОСТИ НГМД ПРИНЯТЬ КОМАНДУ
CALL RWSEC ;ЧТЕНИЕ СЕКТОРА
CALL RDBUF ;ЧТЕНИЕ ИЗ БУФЕРА В ОЗУ
RET

```

```

WRITE: ;ПОДПРОГРАММА ЗАПИСИ СЕКТОРА ИЗ ОЗУ
CALL WIZAV ;ЗАПИСЬ ИЗ ОЗУ В БУФЕР
MUI C, WRITF
CALL SETFUNC ;УСТАНОВКА КОДА КОМАНД ЗАПИСИ СЕКТОРА
CALL RWSEC
RET

```

```

RUSC: ;ПОДПРОГРАММА ЧТЕНИЯ/ЗАПИСИ СЕКТОРА
MVI B, 10 ;СЧЕТЧИК ПОВТОРЕНИЙ ПРИ ОШИБКЕ
;ПО КОНТРОЛЬНОМУ КОДУ
;СЧЕТЧИК ПОВТОРЕНИЙ ПРИ ОШИБКЕ ПОИСКА
RUS0: LDA IOF ;ЗАГРУЗКА ОПРЕДЕЛЕННОЙ РАВНЕ КОМАНД
OUT RADISK ;ЗАПИСЬ В РЕГИСТР ДИСКОВОДА
CALL WREG ;УСТАНОВКА СИГНАЛА 'ПУСК'
CALL WTZR ;ОЖИДАНИЕ СИГНАЛА 'ЗАПРОС ПЕРЕДАЧИ'
LDA IOS ;ЗАСЫЛКА АДРЕСА СЕКТОРА
OUT RADISK
CALL WREG
CALL WTZR
LDA IOT ;ЗАСЫЛКА НОМЕРА ДОРОЖКИ
OUT RADISK
CALL WREG
CALL WIZAV
CALL INERR ;ПРОС ОБЫСКИ
RZ ;НЕТ ОШИБКИ-ВОЗВРАТ
CALL RROS ;ЧТЕНИЕ РОС НГМД
RRC ;АНАЛИЗ ТИПА ОШИБКИ (0-ОШ.УСТАНОВКИ,
; 1-ОШ.КОНТР.КОДА)
CALL RUS3 ;НАЧАЛЬНАЯ УСТАНОВКА ДИСКОВОДА
MVI MUST
DCR C
JNZ RUS0 ;ПОВТОРЕНИЕ ОПЕРАЦИИ ПО ОШИБКЕ
RUSC: JMP ERDISK ;ВЫХОД ПО ОШИБКЕ
DCR B
JNZ RUS0 ;ПОВТОРЕНИЕ ПРИ ОШ. ПО КОДУ
JMP ERDISK

```

```

RDBUF: ;ПОДПРОГРАММА ЧТЕНИЯ БУФЕРА НГМД В ОЗУ
MUI C, RDBFUN ;УСТАНОВКА КОДА КОМАНД
CALL SETFUNC
RDB0: MUI C, 10 ;СЧЕТЧИК ПОВТОРЕНИЙ ПРИ ОШИБКЕ
MVI H, IOF ;АДРЕС ОЗУ
LDA IOF
OUT RADISK ;ЗАПИСЬ КОМАНД
CALL WREG ;УСТАНОВКА СИГНАЛА 'ПУСК'
RDB1: CALL INUVU
JZ RDB1 ;ОЖИДАНИЕ СИГНАЛА 'ВВОД'
RDB2: CALL INZR ;ПРИ СНЯТИИ СИГНАЛА 'ЗАПРОС ПЕРЕДАЧИ'
CALL CNZ ;ЧИТАЕТСЯ СЛЕДУЮЩЕЙ БАЙТ
CALL INZAV ;ЕСЛИ НЕТ СИГНАЛА 'ЗАВЕРШЕНО'
CALL JZ ;ПРОДОЛЖАЕТСЯ ПРИЕМ ДАННЫХ
CALL INERR ;ЕСЛИ НЕТ ОШИБКИ - ВОЗВРАТ
RZ
DCR C
JNZ RDB0 ;ПОВТОРЕНИЕ ОПЕРАЦИИ ПРИ ОШИБКЕ
RDB1: JMP ERDISK ;ВЫХОД ПО ОШИБКЕ
RDB2: IN RADISK ;ЧТЕНИЕ БАЙТА ИЗ РЕГИСТРА ДИСКА
MVI A
MOV R0, A ;ЗАПИСЬ В ОЗУ
INX R0 ;ПЕРЕХОД НА СЛЕДУЮЩИЙ АДРЕС
CALL WREG ;ПОДАЧА ИМПУЛЬСА 'ПУСК'
RET

```

```

RDBUF: ;ПОДПРОГРАММА ЗАПОЛНЕНИЯ БУФЕРА НГМД ИЗ ОЗУ
MUI C, WREG
CALL SETFUNC ;УСТАНОВКА КОДА КОМАНД ЧТЕНИЯ БУФЕРА
MVI C, 10 ;СЧЕТЧИК ПОВТОРЕНИЙ ПО ОШИБКЕ

```

```

.PORC
PLNEXT: TST PLFBLK
;BEG WCNT(R4)
;TST PDONE
;BPL @#PLOC5
;CRET
;IF EQ #MSGKT
;IFT MOVB @#BUFF(R4),OUTB
;INC INC BUFF(R4)
;IFF JSR PC,@#GTBYT
;MOV MOV (SP)+,OUTB
;ENDC
PLOT: INC WCNT(R4)
;BIC #17700,OUTB
;MOVB #4,WHIT ;ВЫХОД БАЙТА
;CMP #111
;BNE PLNEXT
;MOVB #1,IDENT
;BR PLNEXT
;BR PLDONE
;BR PLERR
;MOVB #R4
;ADD #R5
;MOVB #R5,BUFF
;RTS PC
PTINT: INTEN ;,PIC ;ПЕРЕРЫВАНИЕ ПО ПРИЕМУ
;BIC #100,@#PLICS
;MOVB #4,PL18R,INBUF
;BIC #17700,INBUF
;CMP INBUF,#23
;BEQ PLWAIT
;CMP INBUF,#21
;BEQ PIORET
;CMPB #15,INBUF
;BNE NOLF
;DEC WAIT
;TST IDENT ;ПОДСЧЕТ ЧИСЛА СТРОК
;BEQ NOLF
;CLR WAIT
;CLR IDENT
NOLF: MOVB INBUF,@#BUP
;CMP INPCNT,#100
;BGE SAME
;INC INPCNT
;INC BUFF
SAME: MOV INPCNT,RECV0
;TST WAIT
;BNE PTIRET
;TST READ
;BEQ PIORET
;JMP INPFIN ;ВВОД ЗАКОНЧЕН
PLWAIT: BIC #100,@#PLOC5
;BR PTIRET
;JMP ORET
;JMP IRET
;PLERR: BIS #NDERR,@#CSW-Q,BLNK(R4)
;PLDONE: BIC #100,@#PLICS
;BIC #100,@#PLOC5
;CLR PLFBLK
;CLR PLFBLK+2
;DRFIN PL
TABLE: .WORD PLIVEC
; .WORD PTINT-
; .WORD 340
; .WORD PLOVEC
; .WORD PLINT-
; .WORD 200
; .WORD 0
IDENT: .WORD 0 ;ОБРАБОТКА КОМАНДЫ "I"
READ: .WORD 0 ;ФЛАГ ЧТЕНИЕ-ЗАПИСЬ
WAIT: .WORD 0 ;СЧЕТЧИК СТРОК
OUTB: .WORD 0 ;ВЫХОД БАЙТА
INBUF: .WORD 0 ;ВВОД БАЙТА
INPCNT: .WORD 0 ;СЧЕТЧИК СИМВОЛОВ
RECV0: .WORD 0 ;ДЛИНА ПОСЫЛКИ
BUFF: .WORD 0 ;УКАЗАТЕЛЬ БУФЕРА
GOT: .BLKB 40 ;ВХОДНОЙ БУФЕР
PLFBLK: .WORD 0,0,0
;DREND PL
;END

```

чего вся информация передается монитору. Как исключение, обрабатывается команда идентификации, в ответ на которую графопостроитель выдает только одну текстовую строку «EM7042AM».

Драйвер транслируется и компонуется обычными средствами ОС ДВК, затем переносится на системный диск и устанавливается

```

.MACRO PL
.LINK PL
.COPY/SYS PL.SAV SY:PL.SYS
.INSTALL PL

```

После выполнения этих операций драйвер PL будет динамически загружаться при обращении к графопостроителю из программы пользователя. Командой монитора LOAD можно загрузить драйвер до запуска программы. Драйвер PL занимает в оперативной памяти ЭВМ около 600 байт.

В заключение демонстрируются простейшие операции обращения к графопостроителю. Приведенная программа на языке Паскаль выполняет перемещение инструмента в точку с абсолютными координатами 6000, 3000, идентификацию устройства и считывание текущих координат инструмента:

```
PROGRAM PLTEST (INPUT,OUTPUT,PLOTTER,CODER);
VAR
  PLOTTER,CODER:TEXT;
  CH:CHAR;
  POINTER,X,Y,STATUS,INSTRUMENT:INTEGER;
  MESSAGE:ARRAY[1..ALFALEN] OF CHAR;
BEGIN
  REWRITE(PLOTTER,'PL:'); RESET(CODER,'PL:');
  WRITE(PLOTTER,'A6000,3000'); BREAK(PLOTTER);
  READLN(CODER,MESSAGE);
  WRITE(PLOTTER,'!'); BREAK(PLOTTER);
  READLN(CODER,X,Y,STATUS,INSTRUMENT);
  CLOSE(CODER); CLOSE(PLOTTER);

  WRITELN(MESSAGE,X,Y,STATUS,INSTRUMENT);
END.
```

Драйвер PL позволяет использовать все «интерактивные» возможности графопостроителя, в том числе и команды с задержанным ответом (когда, например, при кодировании необходимо сначала подвести инструмент к заданной точке, а затем ввести координаты в ЭВМ). Драйвер обрабатывает только одну «интерактивную» команду. Это значит, что после выдачи команды, которая требует ответа от графопостроителя, поступающая информация должна быть прочитана до начала следующего вывода на графопостроитель. Такое ограничение представляется вполне логичным при организации диалогового взаимодействия программ пользователя с ЭМ-7042АМ, поскольку по «инициативе» графопостроителя никакая информация в ЭВМ не передается.

Некоторые дополнительные удобства вытекают из возможности легко переадресовывать входные и выходные потоки при обращении к графопостроителю через драйвер. Так, команда монитора

..ASSIGN TT: PL:

позволяет получить отладочную распечатку команд на терминале, а назначение

..ASSIGN DK: PL:

переадресовывает вывод на диск, в результате чего, например, приведенная выше программа PLTEST создаст файл DK:O—UTPIC.GRP, содержащий команды графопостроителя. Таким образом, готовые изображения можно сохранять в виде текстовых файлов и выводить на графопостроитель командой .COPY DK:OUTPIC.GRP PL:

Отметим также, что могут быть созданы драйверы других графических устройств, эмулирующие команды ЭМ-7042АМ. В частности, автором разработан драйвер GD графического дисплея для комплексов ДВК-3, совместный по входному языку и размерам рисунка с драйвером PL.

Телефон для справок: 39-47-52, Минск

ЛИТЕРАТУРА

1. Кокорин В. С., Кридинер Л. С., Попов А. А., Хохлов М. М. Тенденция развития диалоговых вычислительных комплексов//Микропроцессорные средства и системы.— 1986.— № 4.— С. 11.
2. Валикова Л. И., Вигдорчик Г. В., Воробьев А. Ю., Лукин А. А. Операционная система СМ ЭВМ РАФОС.— М.: Финансы и статистика, 1984.— С. 7, 171—186.

Статья поступила 24 марта 1987 г.

УДК 681.3.06

Н. А. Мешков

СТУДЕНЧЕСКИЙ НАУЧНО-ПРОИЗВОДСТВЕННЫЙ ОТРЯД МАТЕМАТИКОВ-ПРОГРАММИСТОВ «ПРОМЕТЕЙ»*

В Московском институте электронного машиностроения с ноября 1985 г. функционирует студенческий научно-производственный отряд математиков-программистов «Прометей». Отряд специализируется на разработке программного обеспечения (ПО) персональных компьютеров и пропаганде компьютерной грамотности. В соответствии с договором о научно-техническом сотрудничестве с Институтом проблем информатики АН СССР бойцы отряда подготовили более 20 учебных, деловых, инструментальных и игровых программ для лабораторного практикума по школьному курсу «Основы информатики и вычислительной техники» на базе комплекта учебной вычислительной техники «Ямаха». Лучшие программы переданы для опытной эксплуатации в ИИИ АН СССР, ГИВЦ Министерства просвещения СССР, в педагогические вузы и университеты страны. Министерство просвещения СССР рекомендовало учебную программу «Азбука» к тиражированию и распространению в составе стандартного ПО комплектов учебной вычислительной техники. За создание пакета учебных, деловых, инструментальных и игровых программ для ПЭВМ студенческий научно-производственный отряд «Прометей» награжден 1-й премией Московского городского конкурса молодых ученых и специалистов по учебным и игровым программам.

Структура пакета программ содержит пять разделов. Учебные программы иллюстрируют перспективы использования средств вычислительной техники в народном хозяйстве; формируют навыки общения с компьютером; моделируют физический эксперимент; способствуют закреплению профессиональных знаний. Деловые программы — это система ведения компьютерной документации и электронные таблицы. Инструментальные программы способствуют повышению производительности труда при использовании графических средств персональных компьютеров и оптимизируют рабочие версии программ. Игровые программы — это интеллектуальные и развлекательные игры. В состав пакета входит и системное программное обеспечение.

Приведем характеристики некоторых из программ. После названий программ указаны их авторы. Разработчик всех программ — студенческий научно-производственный отряд математиков-программистов МИЭМ «Прометей».

Программа «Азбука» В. А. Каймина, О. А. Кудиновой, И. В. Яшиной (тел. 297-59-33, Москва) предназначена для обучения работе с клавиатурой персональных компьютеров. Может быть использована при изучении основ информатики и вычислительной техники в средней общеобразовательной школе, ПТУ, вузах и в системе компьютерного всеобуча. Представление информации в наглядной графической форме и оперативная помощь учащемуся способствуют формированию навыков общения с компьютером и интенсификации учебного процесса.

Программа «Физика-8» А. С. Тимошенко (МИЭМ) и В. А. Львовского (НИИ общей и педагогической психологии АПН СССР — тел. 297-59-33, Москва) предназначена для изучения темы «Импульс. Закон сохра-

* Статья представляет собой переработанный для печати текст сообщения, сделанного на семинаре журнала «МП» 15 мая 1987 г.

нения импульса» из школьного курса физики (восьмой класс). Учащимся предоставлена возможность выполнить работу физика-экспериментатора: провести «компьютерный» эксперимент, моделирующий реальную ситуацию, выдвинуть и проверить гипотезы, проанализировать полученные результаты и сформулировать соответствующие выводы. Активная творческая деятельность способствует интенсификации учебного процесса.

Программа «Микробаза данных» М. П. Федосова и Д. А. Бирюкова (тел. 297-59-33, Москва) предназначена для обучения работе с простейшей базой данных и электронными таблицами. Она наглядно иллюстрирует возможности применения персональных компьютеров в деловой сфере. Может быть использована при изучении основ информатики и вычислительной техники в средней общеобразовательной школе и ПТУ, а также в качестве «электронной» записной книжки.

Система ведения компьютерных архивов «Сводка» А. Н. Бунто и Г. Б. Дукаревича (тел. 297-20-52, доб. 2-54, Москва) предназначена для обработки данных, представленных в табличной форме. Может быть использована при изучении основ информатики и вычислительной техники в высших учебных заведениях и в системе компьютерного всеобуча. Использование реляционной модели данных и графических средств персональных компьютеров позволяет облегчить общение с системой и интенсифицировать учебный процесс.

Программа «Компрессор» Д. Л. Морозовского (тел. 297-59-33, Москва) предназначена для оптимизации рабочих версий программ. Она может быть использована в качестве инструментального средства разработки прикладного программного обеспечения на Бейсике. Основной эффект достигается за счет уменьшения объема программ, повышения быстродействия и сокращения времени передачи программ по каналам связи.

Игровая программа «Лабиринт» Д. Л. Морозовского (тел. 297-59-33, Москва) предназначена для организации интеллектуального досуга. Может быть использована при изучении основ информатики и вычислительной техники в средней общеобразовательной школе, ПТУ и в системе компьютерного всеобуча. Программа реализует игровую ситуацию «Путешествие в лабиринте». Способствует развитию памяти, логического мышления и пространственного воображения.

Статья поступила 26 мая 1987 г.

УДК 681.32

В. П. Емелин, В. В. Новоселов, Д. В. Пузатков

КОМФОРТ ДЛЯ МИКРОПРОГРАММИСТА

Основные достоинства микропрограммируемых БИС (комплектов К1804, К1802 и др.) — это высокое быстродействие, интерпретация произвольных систем команд при любых форматах данных, а также возможность реализации прикладных алгоритмов на микропрограммном уровне для достижения экстремальных показателей быстродействия (производительности) микропроцессорных систем. Известный «минус» — трудоемкость разработки и отладки оригинальных устройств по сравнению с типовыми конфигурациями микроЭВМ на основе однокристальных микропроцессоров.

Однако не столько сложность аппарата микропрограммного управления, сколько отсутствие массовых удобных инструментальных средств затрудняет работу микропрограммиста, не даст резко повысить ее производительность.

Удобные инструментальные средства обладают компактностью, простотой и надежностью при интенсивной эксплуатации инструментального комплекса в течение достаточно длительного периода отладки микропрограмм;

легкостью освоения входного языка и естественностью предоставления информации пользователю в диалоговом режиме;

невысокой стоимостью и реальной возможностью приобретения отладочного комплекса в кратчайшие сроки; универсальностью, т. е. отсутствием выраженной ориентации его на узкую номенклатуру БИС одной серии;

возможностью отладки устройств в их реальном конструктивном исполнении и на рабочих частотах;

автоматизированной документацией результатов отладки.

Перечисленные свойства в их подавляющем большинстве отсутствуют в системах программного моделирования отлаживаемых микропроцессорных устройств на ЭВМ общего назначения. Кроме того, самостоятельное расширение библиотеки программных (адекватных и безошибочных!) моделей БИС при постоянном расширении номенклатуры последних — задача, не вызывающая оптимизма у тех, кто ограничен в штатах, средствах и сроках разработки конечного продукта. Гарантия же централизованной разработки унифицированного ряда программных моделей и своевременной поставки их пользователям пока проблематична. Наконец, программное моделирование может служить лишь промежуточной ступенью перехода к отладке натурных макетов микропроцессорных устройств (при отсутствии качественных инструментальных средств проблемы отладки остаются).

Таким образом, моделирующие программы (см., например, [1]) — это средство повышения эффективности учебного процесса (и средство прекрасное!), тогда как аппаратно-программные инструментальные комплексы — орудие труда инженера-разработчика микропрограммируемых процессоров.

По существу реализуемых функций все комплексы для отладки микропрограмм [2] схожи. Базовая микроЭВМ с дополнительным оборудованием (эмулятором программной памяти, памятью трассы для логического анализа, источником питания и др.), развитой периферией и специальным ПО (микроассемблером, динамическим отладчиком и др.) образуют персональное рабочее место микропрограммиста.

Из-за выполнения подобным комплексом функций быстродействующего эмулятора микропрограммной памяти и логического анализатора сложно самостоятельно его изготовить в кратчайший срок (а для многих организаций вопрос сегодня стоит именно так). В этой ситуации для проверки отлаживаемого макета на рабочей частоте естественно пользоваться серийным логическим анализатором, а отладку смысловой правильности микропрограмм выполнять в облегченном (псевдостатическом) режиме с помощью несложных средств, описываемых ниже.

Жизнеспособность и эффективность такого сочетания подтверждает опыт отладки 32-разрядного одноплатного функционального расширителя для микроЭВМ СМ1800 (50 микросхем серий К1804, К531, К555, К155, К589, К556, К580 в структуре Е2) со сложным микропрограммным обеспечением (арифметические, прямые и обратные тригонометрические функции в форматах с плавающей точкой и с фиксированной точкой).

Предлагаемые сервисные аппаратные и программные средства дополняют базовую микроЭВМ СМ1800 или комплекс АРМ2-05 на ее основе (см. рисунок). Несложными модификациями можно достичь сопряжения и с персональными компьютерами, совместимыми с СМ1800 на уровне системного интерфейса и ПО.



Структурная схема инструментального комплекса для микропрограммиста.

Вся аппаратная часть состоит из простейшей платы сопряжения (конструктив Е2) базовой микроЭВМ с отлаживаемым макетом микропроцессорного устройства. На плате находятся: буферный регистр микрокоманды (не более 96 разрядов), приемники (не более 64 разрядов) и логика сопряжения с шиной И41.

Буферный регистр необходим для побайтной передачи информации из базовой микроЭВМ на плату сопряжения по шине И41. С буферного регистра микрокоманда целиком поступает через разъем и многожильный кабель (0,5 м) на отлаживаемое устройство пользователя, непосредственно (с помощью контактных приспособлений) на панельку РС-24—8 (свободные при этом от микросхем ППЗУ микропрограммной памяти).

В обратном направлении, т. е. от отлаживаемого устройства к плате сопряжения, аналогично передается адрес следующей микрокоманды и трасса сигналов с контролируемых точек макета. Управляют синхронизацией макета пользователя также через плату сопряжения.

Ограничений на конструктив макета нет, так как он связан с платой сопряжения по кабелю. Поскольку система не предназначена для работы на высоких частотах, подобная кабельная связь (позволяющая пользователю выбрать произвольный конструктив) предпочтительнее специализированных отладочных плат, выполненных в конструктиве базовой системы и снабженных краевыми разъемами для передачи сигналов в быстродействующих отладочных комплексах [2]. Предлагаясь плата сопряжения расположена в каркасе микроЭВМ СИ1800 (с лицевой стороны), а отлаживаемый микропроцессорный макет — на столе рядом с дисплеем.

На начальном этапе выявления ошибок монтажа любая из точек макета может быть проконтролирована с помощью осциллографа или тестера. В дальнейшем при отладке логики выполнения микропрограмм необходимость в этих приборах отпадает: сигналы с трассируемых точек макета отображаются на экране дисплея в цифровом виде и (по желанию пользователя) могут сопровождаться мнемониками.

Программное обеспечение отладочного комплекса включает в себя готовые и специально разработанные компоненты:

популярный микропрограммный ассемблер AMDASM [2...4], настраиваемый на произвольный формат микрокоманд;

программу перестановки разрядов и полей в микрокомандах для приведения логического формата к физическому размещению разрядов в конкретных ППЗУ;

программу доопределения безразличных значений в полях микрокоманды и подготовки файлов для программирования ППЗУ;

экранный редактор текстов для первоначальной записи и последующих модификаций текстов микропрограмм на уровне мнемоник и на уровне объектных кодов;

отладочный монитор, организующий в диалоговом режиме (на русском языке) взаимодействие с пользователем при выполнении микропрограмм (транслированных ранее с применением перечисленных выше средств) на натурном макете микропроцессорного устройства. Диалоговый отладочный монитор (35К байт) написан на Паскале.

Все программные средства вместе с операционной системой ОС1800 или CP/M, а также подробное руководство по отладочному комплексу размещены на одной дискете. На второй дискете (дискетод В) удобно хранить все микропрограммы на входном языке микроассемблера и в кодах.

Сеанс отладки начинается с диалога «настройки» монитора на работу с конкретным макетом. Необходимо задать число и вид синхропоследовательностей, моменты съема трассы контролируемых точек и смены информации в буфере микрокоманд.

С платы сопряжения на макет пользователя можно подавать от одной до восьми синхропоследовательностей произвольной скважности и с произвольным фазовым сдвигом. Поскольку синхропоследовательности формируются программно, т. е. относительно медленно, то режим функционирования отлаживаемого макета — не динамический, а по сути псевдостатический.

Трассы контролируемых точек можно снимать в произвольные (по желанию пользователя) моменты временной диаграммы такта микропроцессорного устройства. Число запоминаемых шагов ограничено лишь размером свободного пространства памяти на дискете. Максимальная ширина трассы — 64 разряда, включая адрес следующей микрокоманды (не более 12 разрядов). Разумеется, чем шире наблюдаемая трасса, тем эффективнее отладка. В этом смысле трассировка лишь разрядов адреса (пусть даже на рабочей частоте) мало полезна для выводов о правильности функционирования операционных блоков микропроцессора, хотя и применяется в некоторых отладочных комплексах.

Для удобства визуального восприятия и анализа 64-разрядных трасс пользователь может разбить их на любое число полей (групп разрядов) и снабдить мнемониками. Соответствующие мнемоники назначаются полям в начале сеанса при «настройке» монитора.

Благодаря простоте изготовления аппаратной части (платы сопряжения) и широкому спектру функций ПО данный отладочный комплекс полезен практически всем разработчикам систем на базе микропрограммируемых микропроцессоров.

Адрес для справок: 107022, Ленинград, ул. Профессора Попова, д. 5, ЛЭТИ, кафедра вычислительной техники, тел. 234-25-03.

ЛИТЕРАТУРА

1. Разработка и отладка микропрограммного обеспечения цифровых систем на основе секционированных микропроцессоров/А. Г. Алексеев, А. В. Галоненко, А. Д. Иванников, И. Д. Курилов//Микропроцессорные средства и системы.—1986.—№ 1.—С. 37—43.
2. Микропроцессоры: системы программирования и отладки/Под ред. В. А. Мясникова, М. Б. Игнатова.—М.: Энергоатомиздат, 1985.
3. Мик Дж, Брик Дж. Проектирование микропроцессорных устройств с разрядно-модульной организацией. В 2-х книгах/Пер. с англ.—М.: Мир, 1984.
4. Мета-ассемблер МЕАСС для микропроцессорных систем с наращиваемой разрядностью/Г. П. Мозговой, С. С. Семенова, Е. И. Семин, О. В. Трещалин//Микропроцессорные средства и системы.—1986.—№ 3.—С. 20—22.

Статья поступила 5 апреля 1987 г.

К. А. Сизов

ПРОГРАММА ОБРАБОТКИ ТЕКСТОВ С ПЕРЕМЕННОЙ ШИРИНОЙ ЗНАКОФОРМИРУЮЩЕЙ МАТРИЦЫ

В программное обеспечение (ПО) любой персональной ЭВМ (ПЭВМ) входит программа обработки текстов. Преимущества системы ПЭВМ — принтер очевидны: возможность редактирования, выравнивания кромок текста, оперативное исправление ошибок без помарок — в общем, возможность «семь раз отмерить», прежде чем загубить очередной лист бумаги. Но не все, у кого есть ПЭВМ с принтером и программой обработки текстов, ими пользуются из-за неудовлетворительного качества печати, вызванного точечной структурой букв матричного принтера.

Проблема во многом решается благодаря печатающим устройствам с режимом печати NLQ (Near Letter Quality), что означает «близкий к качеству литеры». Такие принтеры используют для построения изображений большее число точек, снижая дискретность изображения.

Когда требуется создать иной вид шрифта или печатать на каком-либо национальном языке, большинство ПЭВМ позволяют в стандартной матрице изобразить на своем дисплее желаемые буквы, а выдавать их на принтер не могут, так как у большинства печатающих устройств есть внутренний знакоформиователь, а существующие отдельные программируемые модели принтеров дефицитны и дороги.

В таком случае изображение, сформированное на экране, копируется на бумагу, но тогда об NLQ не может быть и речи, да и читаемость текста на экране дисплея оставляет желать лучшего.

Эту проблему решает способ печати со знакоформирующей матрицей переменной ширины.

Для полиграфического текста (наиболее удобочитаемого) характерны разная ширина букв и различное соотношение ширины и высоты знаков (в отличие от компьютера, где все символы формируются, как правило, в фиксированной матрице 8×8). Например, если строчная буква «н» имеет минимальную ширину, то такие буквы, как «ж» или «ю» в полиграфическом тексте в 2,5 раза шире буквы «н» (расстояние между соседними буквами в слове также различны).

При матрице из 8 точек по горизонтали вписывание, например, букв «Ш» русской или «т» латинской приводит к несимметричности расположения окаймляющих букв пустых бит и к различному расстоянию между символами в строке. Это ухудшает читаемость текста.

Хотя все буквы текста, напечатанного на пишущей машинке, имеют одинаковую ширину, текст читается легче, так как нет дискретизации на точки. Существуют принтеры (например, фирмы DEC), использующие для формирования букв матрицы со значительно большим числом точек по горизонтали и по вертикали. Форма символов, содержащихся в их знакоформиователях, очень схожа с формой литер пишущей машинки. Используются матрицы 16×24: дискретизация в этом случае практически незаметна, а форма букв прямоугольная и поэтому несколько ближе к натуральной. Заметим, однако, что записанные в ПЗУ их знакоформиователей буквы кириллицы сформированы в матрице 24×24 (форма символов на листе — квадратная).

Во многих ПЭВМ используется матричный графический дисплей с разрешением 320×200 точек или кратным ему. Буквы в этом случае имеют квадратную форму либо прямоугольную с соотношением сторон 1:2.

И то, и другое неоптимально для зрительного восприятия. Кроме того, при таких формах отображения в строке размещается либо 40, либо 80 символов, а машинописный лист (формат А4) содержит 60—64 позиции. В случае 40-позиционного дисплея площадь его экрана недостаточна, а в случае 80-позиционного дисплея она используется неполностью (60 позиций из 80, к тому же печать на 80 позиций требует более высокой разрешающей способности дисплея).

Все эти соображения привели к программному получению текста, хорошо читаемого на экране и на бумаге без улучшения разрешающей способности дисплея и принтера. Изображение формировалось на дисплее, а принтер с помощью специальной подпрограммы копировал его на бумагу. Дисплей и принтер имели по 320 точек в строке. Программа позволяет формировать, редактировать и выдавать на печать тексты на любом языке с практически любым шрифтом. Хорошая читаемость текста достигается представлением символов (букв), приближенным к полиграфическому. Изображения букв можно корректировать в процессе работы над текстом или в уже написанном тексте.

Вывод символов с переменной шириной матрицы на принтер и на экран рассматривался как одна задача, несмотря на то что разрешающие способности этих устройств могут весьма различаться.

При перемещении курсора по тексту он должен постоянно изменять свою ширину в соответствии с шириной символа, под которым он находится. Двигаясь вертикально по тексту, курсор будет изменять и свою видимую горизонтальную координату, хотя номер его горизонтальной позиции будет сохраняться постоянным.

В процессе редактирования при вводе каждого нового символа на место старого должна изменяться и длина строки в соответствии с шириной вводимого символа. В данной программе эта задача была решена таким образом, что после ввода каждого символа изображение всей текущей строки заменяется на новое. Эффект мигания строки в процессе набора текста снижен специальной быстродействующей подпрограммой вывода символа на экран.

Для выравнивания правых кромок строк (текста, столбцов, таблиц и др.) в программу введен псевдосимвол переменного пробела. С его помощью можно установить в каждом конкретном месте ширину пробела с точностью до одного бита. Структура переменного пробела такова: символ переменного пробела «0» плюс следующая за ним длина пробела в битах (рис. 1). При нажатии соответствующей функциональной клавиши правая кромка выравнивается автоматически.

Алгоритм подпрограммы выравнивания правой кромки (рис. 2) — рекурсивный. Он изменяет размер строки заменой всех ее пробелов на переменные пробелы. Подпрограмма поочередно увеличивает длину каждого пробела на один бит и после каждой такой операции вычисляет длину строки и сравнивает ее с необходимым числом точек по горизонтали. По достижении строкой необходимой длины выравнивание заканчивается.

Режим выравнивания отменяется заменой всех переменных пробелов на фиксированный символ пробела (ширина — 5 бит).

Строки программы имеют различное число символов, поэтому под каждую строку отводится 80-байтовая зона памяти (средняя длина строки в обыкновенном тексте — 50—60 символов).

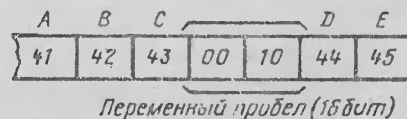


Рис. 1. Структура переменного пробела

СЕМИНАР «МП»

С момента своего основания журнал «МП» проводит регулярные встречи авторского актива с читателями. Периодичность их 2—3 раза в месяц. Предмет обсуждения: последние разработки промышленности или схмотехнические находки талантливых умельцев, отдельные программные решения или конкретный аппаратно-программный комплекс, вопросы применения или новые технические принципы. Неизменной остается лишь форма общения с аудиторией — авторы не столько рассказывают, сколько показывают свои результаты. На сцене обычно установлена аппаратура самого различного назначения: контроллер для управ-

ления цеховыми испытательными стендами или станками с ЧПУ, «домашняя ЭВМ», или профессиональный персональный компьютер, автомобильный «радиос-центр», звукоинтегратор для создания в диалоге с микроЭВМ музыкальных произведений или промышленные средства отладки встраиваемых микропроцессоров. Словом, это должен быть полезный в цехе, лаборатории или дома «компьютеризованный прибор» или система. На каждом очередном заседании выносятся до десяти конкретных разработок... Подробнее о целях, формах и задачах семинаров см. материалы в «МП» (1986, № 2 и № 3, а также в № 2, 3 и 6 1987 г.).



В январе—июне 1988 г. во Дворце культуры завода им. Владимира Ильича будет проведен учебный цикл семинаров

МикроЭВМ и микропроцессоры: основы применения

5 января — Обзор основных классов отечественных микропроцессоров и микроЭВМ

19 января — Аппаратная структура и программное обеспечение типового микропроцессорного комплекса и устройств связи с объектом автоматизации

2 февраля — Периферийное оборудование микроЭВМ: дисплеи, графопостроители, гибкие диски, электронный «квазидиск», ЦМД-память, клавиатура, печатающие устройства

1 марта — Аппаратно-программная реализация многомашиных комплексов и локальных вычислительных сетей

15 марта — Операционные системы микроЭВМ

5 апреля — Постоянные запоминающие устройства (ПЗУ) и программируемые логические матрицы (ПЛМ): физические основы, методы и техника программирования и отладки

3 мая — МикроЭВМ в обучении: школьная и вузовская информатика, профориентация, повышение квалификации

17 мая — Экономика электронизации

7 июня — Компьютер — новый инструмент творчества: компьютерные игры, ЭВМ и музыка

28 июня — Персональные компьютеры

Ведет семинары — Г. Р. Громов
Лекции-семинары проводятся по вторникам в 12 часов.

Справки о семинарах можно получить по телефону 923-00-19 у методиста Политехнического музея Ермолаевой Татьяны Юрьевны. Абонементы будут продаваться в кассе Политехнического музея (подъезд № 9, тел. 923-69-16).

Авторы и читатели журнала «МП», желающие выступить с сообщением, могут согласовать порядок участия в семинаре по телефону 208-73-23 с научным секретарем семинаров Матвеевым Сергеем Семеновичем.

УЧЕБНЫЙ ВИДЕОФИЛЬМ «МИКРОПРОЦЕССОРЫ» (вводный курс)

УДК 681.325.5+681.326—181.4

А. В. Александров, М. Г. Весноватов, В. В. Павлов

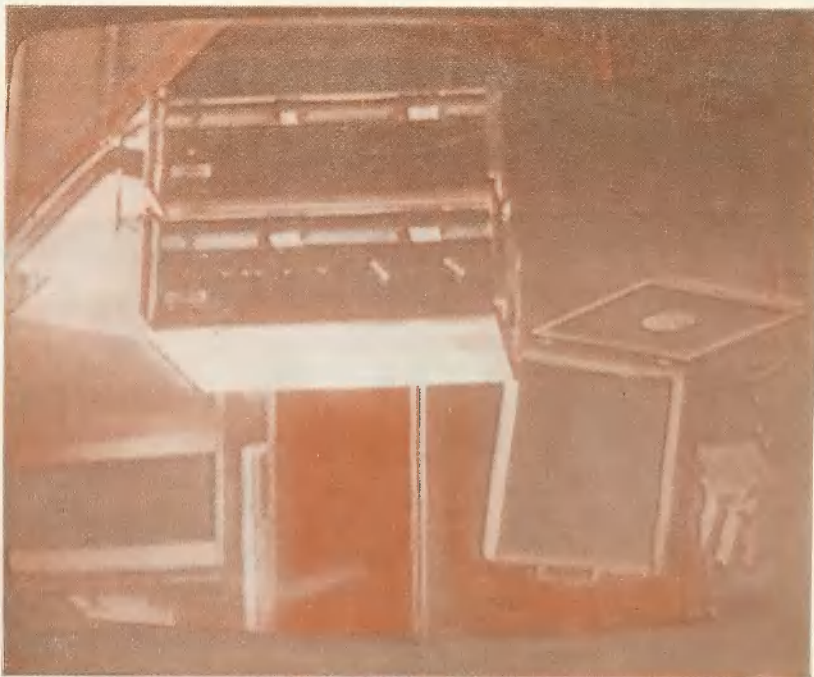
Цель курса — развить созданную базу знаний для понимания микропроцессорной техники (МПТ), содействовать активному использованию микропроцессоров (МП) в машинах и оборудовании, обеспечить широкое внедрение МПТ во всех отраслях народного хозяйства.

Видеокурс предназначен для специалистов с широким диапазоном по уровню подготовки и выполняемым обязанностям. Он будет полезен для инженеров-разработчиков, технологов, а также администраторов различных уровней.

Структура видеокурса. Видеокурс состоит из 10 лекций продолжительностью от 25 до 45 минут.

Лекция 1-я «**История развития вычислительной техники**» характеризует путь, пройденный вычислительными устройствами от электронных ламп, транзисторов до больших и сверхбольших интегральных схем. Вы поймете, что такое микропроцессор и каковы его преимущества для создания систем управления приборами, машинами, оборудованием.

Лекция 2-я «**Аппаратные средства микропроцессорных систем**» посвящена архитектуре построения микропроцессорного комплекта. Вы узнаете о логических элементах, на основе которых строится схемотехника данных для микропроцессора, об основах двоичной арифметики.



Микропроцессорная система диагностики автотракторных двигателей



Работу отладочного устройства «Электроника 580» объясняет инженер-консультант Ф. Х. Белялетдинова

В лекции 3 «Методы программирования микропроцессоров (машинный код)» подробно рассматривается программирование микропроцессоров в машинных кодах (системы счисления и перевод из десятичной системы в шестнадцатеричную), демонстрируется перевод десятичного числа в шестнадцатеричное. Представлены отдельные команды и группы команд для микропроцессора КР 580.

В лекции 4 «Методы программирования микропроцессоров (ассемблер)» рассказано об основных принципах программирования на языке ассемблера, о преимуществах написания программ на ассемблере, о формате предложений языка, приводятся примеры мнемонических кодов операций, символических имен для адресов операндов и символических имен для адресов команд. Кроме того, в ней определены основные функции и возможности программы, транслирующей программу на ассемблере в машинные коды.

В лекции 5 «Разработка микропроцессорных систем» подробно рассматривается схема разработки с двумя параллельными ветвями — схемотехнической и программной, аппаратные и программные средства, необходимые на каждом этапе.

В 6-й лекции «Разработка и выполнение программ в реальном масштабе времени» на примере микропроцессора серии КР580 рассмотрена реализация двух программ. Работа программы, реализующей счетчик случайных событий, показана на уровне операторов ассемблера. Принцип работы программы, поддерживающей диалоговый режим «оператор—ЭВМ», описан на уровне блок-схемы. В заключение названы проблемы разработки и отладки устройств реального времени.

Лекция 7 «Средства и методы отладки микропроцессорных систем» разделена на три

части. В первой части демонстрируется отладочное устройство — «Электроника-48», приводятся его параметры, возможности. Во второй — подробно рассматривается отладочное устройство — «Электроника-580». В третьей — излагается концепция отладки микропроцессорных систем управления с помощью кросс-систем в сочетании со схемными эмуляторами.

Лекция 8-я «Архитектура микропроцессоров» посвящена архитектуре и периферийным устройствам микропроцессора. Приведена блок-схема микропроцессорной системы, показаны связи с внешними устройствами, схемные решения на базе популярного микропроцессора серии КР580, рассмотрены способы сопряжения с аналого-цифровыми и цифроаналоговыми преобразователями сигнала.

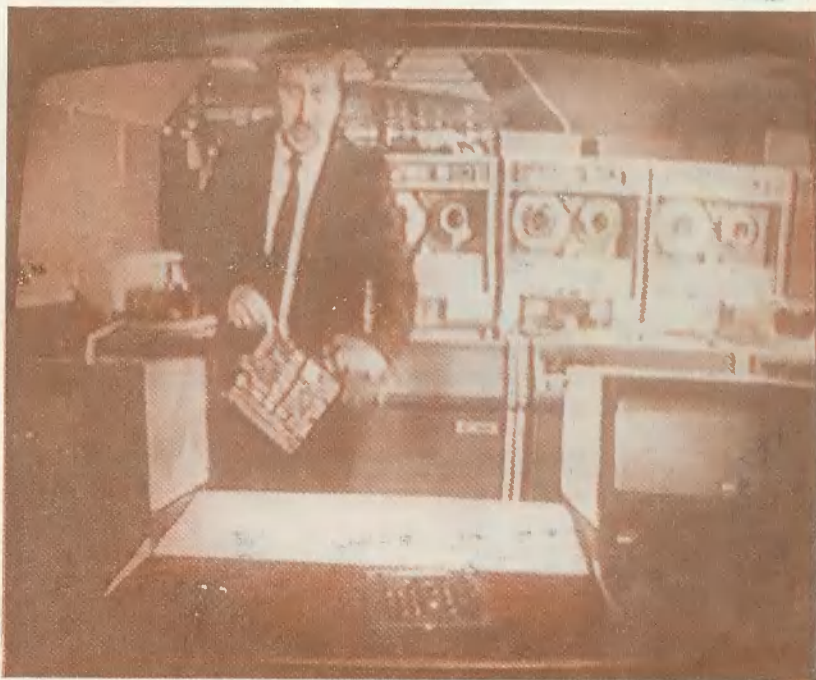
В 9-й лекции «Применение микропроцессоров в народном хозяйстве» показан ряд

МП-систем, работающих в медицине, сельском хозяйстве, промышленности и дается понятие об основах применения микропроцессоров в народном хозяйстве: рассказывается о распределенных системах и для примера рассматриваются аппаратные и программные средства управления инкубатором.

В последней 10-й лекции «Тенденции развития микропроцессорной техники» представлена вся номенклатура серийно-выпускаемых БИС, а также БИС однокристальных микроЭВМ, дается обзор тенденций развития МП.

Цикл учебных лекций «Микропроцессоры» поможет вам понять проблемы внедрения МП и сделать первый шаг в деле самостоятельной разработки МП-средств автоматизации.

Телефон для справок об условиях поставки: 468-81-75, Москва.



Функциональные возможности процессорной платы микроЭВМ ДВК-2М эквивалентны возможностям нескольких десятков плат центрального процессора мини-ЭВМ СМ-4

СИСТЕМА МИКРОЭВМ ВТ 20 IV НА 4 РАБОЧИХ МЕСТА

Аппаратное построение. Все дисплеи равнозначны. Центральный процессор системы находится в отдельном корпусе. Блок сопряжения диска имеет микропроцессорное управление; наряду с физическим управлением диском, он обеспечивает и функции управления файлами. Исполнение интерфейса таково, что к блоку сопряжения диска могут иметь доступ несколько процессоров. Каждое рабочее место располагает отдельным микропроцессором и полным объемом памяти. Фактически в системе имеются четыре самостоятельные микроЭВМ, использующие общее ВЗУ.

Дисковый процессор расположен на двух платах. Набор его инструкций — 158. Рабочая частота 4 МГц; память содержит 64К байт ЗУПВ (RAM) и 6К байт ПЗУ (ROM). ПЗУ можно расширить до 16К байт, но в соответствии с емкостью встроенного ПЗУ уменьшается используемая область ЗУПВ.

Управляющие платы процессора. Каждому рабочему месту соответствует одна управляющая плата процессора, работающая на микропроцессоре.

Построение памяти аналогично построению памяти процессора дисков с тем отличием, что здесь встроен только 1К байт ПЗУ. Каждая отдельная плата — помимо процессора и памяти — содержит и последовательный блок сопряжения, имеющий интерфейс модема V.24 и «токовой петли 20 мА/20В». К этим блокам сопряжения можно подключить и рабочие места с дисплеем.

Периферийные устройства традиционные. Накопители на магнитных дисках: накопитель на сменных дисках (2×2,5М байт); АЦПУ — ленточное со скоростью печати 300 строк/мин. Рабочие места с дисплеями типа VDN 52500.

Программное обеспечение. Различные функции системы программного обеспечения микроЭВМ разделены между отдельными аппаратными блоками. На процессоре дисков выполняется прогон средств управления общими элементами МО.

Видеотон. Завод вычислительной техники. 1021, Будапешт, 11.



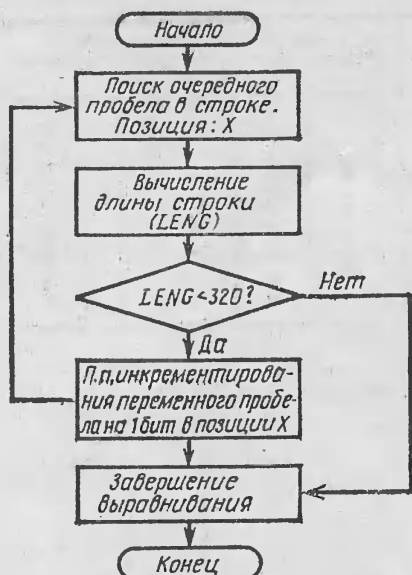


Рис. 2. Алгоритм подпрограммы выравнивания правой кромки

Подпрограмма выдачи символов на дисплей также имеет алгоритм, непохожий на обычные алгоритмы печати символов в ПЭВМ. Структура многих дисплеев ПЭВМ такова, что экранная зона ОЗУ состоит из лежащих одна под другой строчек байтов (биты которых формируют изображение). В таком же виде в большинстве ПЭВМ представляются и изображения букв знакоформирователя (рис. 3,а). При различной ширине символов такое их представление неудобно. Так как высота матрицы фиксирована и равна 8 бит, то наиболее целесообразна форма в виде последовательности байтов, составляющих столбцы изображения символа (рис. 3,б). При вертикальной структуре символа легко менять ширину знаков.

Изображение символа из вертикальной матрицы знакоформирователя в горизонтальные байты экранной памяти переносит специальная подпрограмма. Ее алгоритм основан на вращении через перенос битов в байтах матрицы знакоформирователя и дисплея и в целом напоминает алгоритм перемещения кубиков в венгерском кубе.

Подпрограмма выдачи строки на дисплей считывает код очередного символа из буфера строки, вызывает подпрограмму печати символа и подсчитывает длину строки. Подпрограмма выполняется до тех пор, пока длина строки (в битах) не достигнет горизонтального размера экрана.

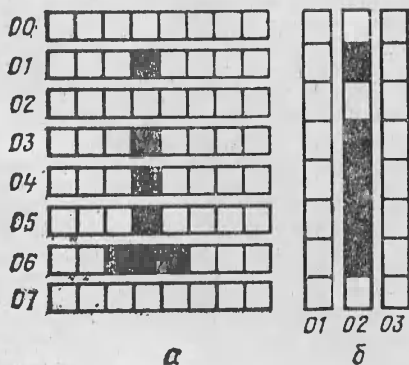


Рис. 3. Изображения букв в памяти знакоформирователя

Особенность программы — можно оперативно менять изображения символов, гарнитуру шрифта и алфавиты.

Специальная графическая подпрограмма увеличивает коррегируемый символ на экране в 8 раз. С помощью клавиш можно изменить изображение символа, нарисовать новое, изменить код символа или поменять символы местами.

Для смены гарнитуры шрифта или замены его другим необходимо загрузить соответствующий бинарный файл с диска или ленты.

Воспринимаемость текста улучшится также в сочетании с переменной шириной знакоформирующей матрицы и увеличением числа точек знакоформирователя, так как использование в данной программе матрицы, имеющей 8 точек по вертикали, недостаточно для высокого качества печати.

Для дальнейшего улучшения читаемости текста на бумаге можно использовать у принтера свое ЗУ знакогенератора. Предлагается оперативно изменять его содержимое в соответствии с изменениями знакогенератора дисплея; ЗУ принтера в этом случае можно выполнить на МНОП-структурах либо в виде обычного ОЗУ, загружаемого при старте программы.

Блоки программы можно включить в состав уже имеющихся более мощных редакторов текстов. Программы, использующие для представления информации матрицу знакоформирователя переменной ширины, могут расширить круг пользователей ПЭВМ в союзных республиках.

Адрес для справок — 127567, Москва, ул. Белозерская, д. 116, кв. 198.

Статья поступила 25 февраля 1987 г.

УДК 681.3.06

Л. Д. Хацкевич. Л. Л. Койлис

СУБД ДЛЯ ПЕРСОНАЛЬНОЙ ЭВМ «ЭЛЕКТРОНИКА 85»

Создание СУБД «Парус» (см. рисунок) преследовало следующие цели: организацию взаимодействий с конечными пользователями, не являющимися специалистами в области программирования и вычислительной техники; сокращение затрат на разработку прикладного программного обеспечения при внедрении автоматизированных информационных систем; повышение достоверности и сокращение стоимости обработки информации. Для реализации указанных целей необходимо решить следующие задачи:

обеспечить семантическую целостность базы при модификации концептуальной схемы, при вводе и удалении данных;

создать диалоговый непроцедурный язык взаимодействий, близкий к естественному языку, сочетающий мощностные и простоту;

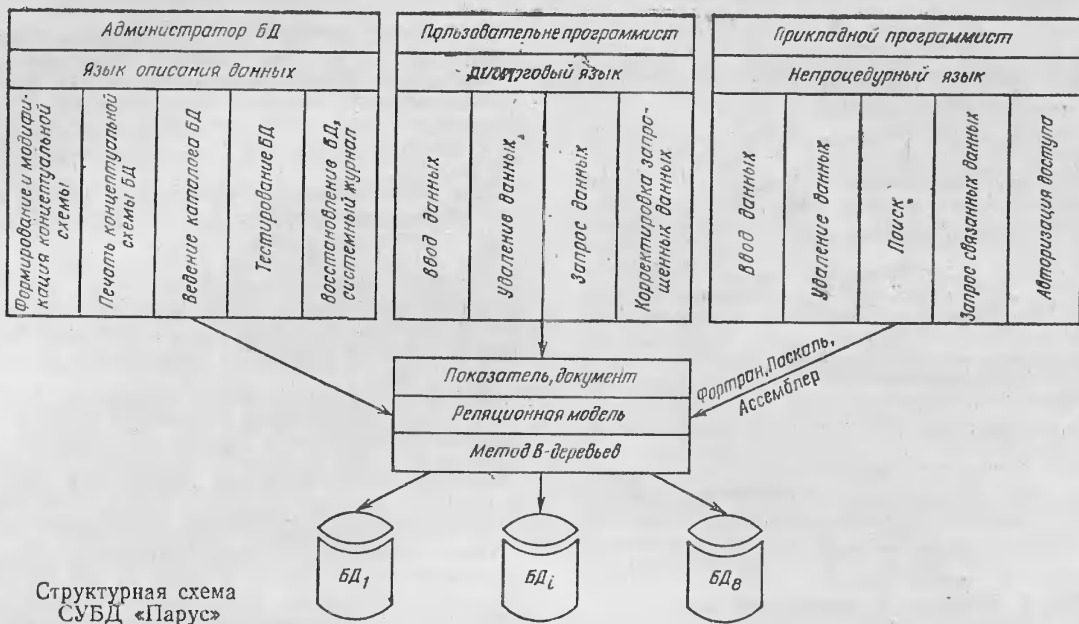
обработать широкий круг произвольных и каталогизированных запросов на поиск данных с обеспечением полноты и связности выдаваемых пользователю ответов на запросы;

провести автоматизированную реструктуризацию БД с автоматической реорганизацией физической структуры БД;

обеспечить тестирование и просмотр физической БД; динамически построить сценарии диалога с пользователем, исключить необходимость проектирования прикладных программ для осуществления синтаксического и семантического контролей, для помещения, удаления или поиска данных в БД;

вести физическую БД без использования программ для сжатия БД и «сборки мусора»;

создать развитый интерфейс прикладного программиста для расширения функциональных возможностей СУБД при адаптации и построении различных информационных систем.



Структурная схема СУБД «Парус»

В основу описания данных на уровне концептуальной схемы БД положена реляционная модель, расширенная с точки зрения семантики данных и адекватности схемы БД информационной модели предметной области.

СУБД «Парус» реализована в операционной среде ФОДОС.

Дальнейшее функциональное развитие СУБД «Парус» для ПЭВМ «Электроника 85» связана с переходом в ОС РВ и предоставлением пользователям возможностей: задания и выполнения формул расчета показателей в процессе обработки произвольных запросов, корректировки запрошенных данных,

подключения блоков пользователей при вводе данных в БД.

Опыт построения автоматизированных информационных систем на основе СУБД «Парус» показывает высокую адаптируемость системы к различным предметным областям, снижение затрат на разработку и эксплуатацию систем, создание необходимых условий конечным пользователям для работы с БД.

Адрес для справок: 394026, Воронеж, ул. Дружников, 10, тел. 52-78-93. УЧФ РО ВНИПИучет, Койлису Л. Л.

Сообщение поступило 10 февраля 1987 г.

УДК 681.3.06

А. Н. Степанов, В. А. Жуков

КОМПЛЕКС АППАРАТНО-ПРОГРАММНЫХ СРЕДСТВ ОТЛАДКИ МИКРОПРОГРАММ ДЛЯ РАЗРЯДНО-МОДУЛЬНЫХ МИКРОПРОЦЕССОРОВ

К специальным аппаратно-программным средствам относятся микроассемблеры, отладчики, эмуляторы, компиляторы языков микропрограммирования, а также аппаратные средства для имитации внешней среды, ввода микропрограмм в микропроцессорный блок (МБ) и т. д. Среди возможных путей автоматизации отладки микропрограмм — использование технологической ЭВМ (ТЭВМ), состыкованной с МБ через аппаратный интерфейсный блок (АИБ). На ТЭВМ выполняется программа-отладчик, загружающая микрокоманды через АИБ в МБ и организующая выполнение микропрограмм в МБ (прокрутку). Генерация микропрограммы также возлагается на ТЭВМ или кросс-систему

и поддерживается микроассемблером или настраиваемым компилятором. По такой схеме построена известная система автоматизации микропрограммирования МЕТАМИКРО [1].

Обычно в состав АИБ таких систем включается быстрое запоминающее устройство, заменяющее ПЗУ микропрограмм и позволяющее приблизиться к реальной скорости выполнения микропрограммы. Однако построение быстрого ОЗУ с временем выборки 40...60 нс объемом 2...4К слов по 64 и более разрядов (что требуется для микропрограммного ПЗУ МБ на разрядно-модульных секциях) обходится довольно дорого и вызывает большие затруднения у пользователей системы автоматизации микропрограммирования.

При прокрутке микропрограммы на пониженной скорости применение ОЗУ теряет смысл, так как обнаружить ошибки, связанные с временными соотношениями оно не позволяет, а для выявления алгоритмических ошибок достаточно сохранять образ микропрограммы в ОЗУ ТЭВМ и загружать микрокоманды из этого ОЗУ через дополнительный регистр в МБ. Таким образом, АИБ упрощается до регистра микрокоманды.

В разработанном комплексе аппаратно-программных средств автоматизации процесса прокрутки микропрограммы со скоростью, задаваемой ТЭВМ, блок АИБ отличается простотой и универсальностью. Комплекс ори-

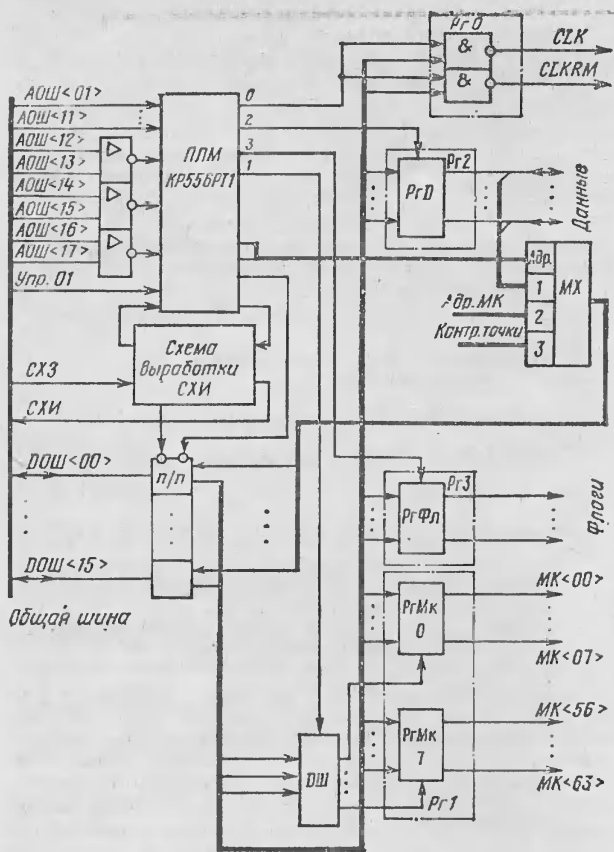


Рис. 1. Структурная схема аппаратного интерфейсного блока (АИБ)

ентрирован на работу с МБ на основе разрядно-модульных секций серии К589, К1802, К1804, где выполнение микрокоманды начинается после ее занесения по фронту тактового импульса в регистр микрокоманды. Адрес следующей микрокоманды вырабатывается секвенсером, входящим в состав МБ. Таким образом, для выполнения микрокоманды необходимо загрузить регистр микрокоманды и прочитать адрес следующей микрокоманды. Для этой цели в состав АИБ включен регистр распаковки поступающего из ТЭВМ по частям слова микрокоманды и мультиплексор для сбора и передачи в ТЭВМ необходимой информации.

Структурная схема АИБ приведена на рис. 1. Блок содержит четыре доступных со стороны ТЭВМ виртуальных регистра: регистр 0 управления доступен только для записи; регистр 1 — для записи микрокоманды и чтения адреса следующей микрокоманды; регистр 2 данных доступен для записи и чтения; регистр 3 — для записи флагов и чтения информации в контрольных точках.

Запись кода в регистр 0 вызывает его стробирование управляющим сигналом, передачу кода на тактовый вход МБ (СЛК) и на отдельный тактовый вход регистра микрокоманды МБ (СЛКRM). Эти входы активизируются не одновременно, но в штатном режиме работы МБ с микропрограммой в ПЗУ они объединяются. Такое раздвоение тактового сигнала полезно при отладке, так как подача СЛКRM вызывает только ввод новой микрокоманды без записи результатов ее исполнения. Подача двух сигналов одновременно вызывает запись в регистры микропроцессора и внутреннего СОЗУ МБ.

Содержимое регистров считывается с помощью мультиплексора, собирающего информацию об адресе микрокоманды, данных на внутренней шине МБ и данных в произвольно выбранных контрольных точках МБ. Управление записью и чтением выполняется с помощью программируемой логической матрицы. Слово микрокоманды выдается из ТЭВМ побайтно с указанием номера байта, который поступает на дешифратор, управляющий записью в регистр 1. Данный АИБ ориентирован на работу совместно с ТЭВМ типа СМ 1300 и другими (с интерфейсом «общая шина»), поэтому содержит узлы связи с шиной и выработки стробирующего сигнала «синхронизация источника». Конкретный АИБ был рассчитан на работу с 64-разрядными микрокомандами при 11-разрядном адресе.

Основу системы МЕТАМИКРО составляют программа-генератор трансляторов и настраиваемый транслятор с языка символьного микропрограммирования. Эти компоненты определяют язык и осуществляют сборку микропрограммы, которая затем прокручивается на МБ с помощью АИБ с подменным ОЗУ и программы отладочного монитора.

Возможности, которые дают пользователю генератор трансляторов и настраиваемый транслятор, перекрываются микропрограммным ассемблером AMDASM [2] (определение языка и его использование производятся в двух фазах одной и той же программы), позволяющим строить достаточно компактные программы. Ассемблер AMDASM включен в комплекс отладки как средство написания микропрограмм вместе с микропрограммным отладчиком для прокрутки микропрограмм на МБ с помощью простого АИБ и программой прожига ПЗУ серий КР556, К573 и К155 для размещения отлаженной микропрограммы. Программные компоненты обмениваются между собой файлами данных (рис. 2).

Рассмотрим подробнее отладчик микропрограмм. Он оперирует с образом микропрограммы, имеющей до

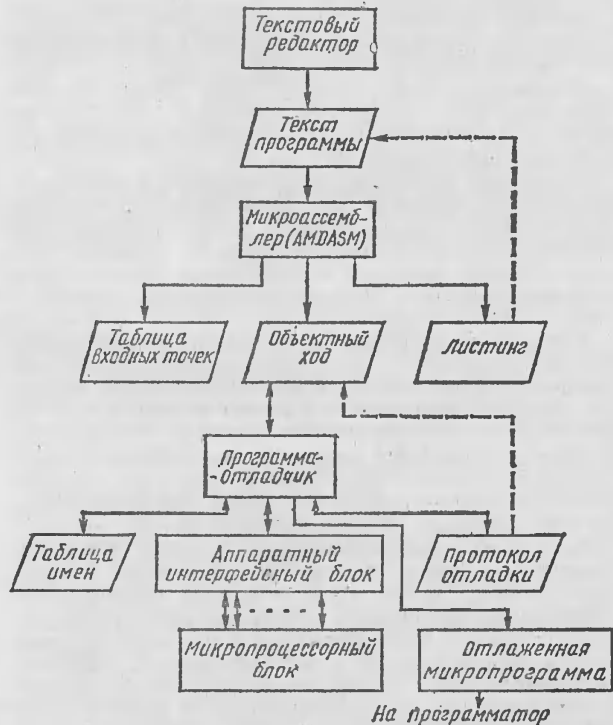


Рис. 2. Схема взаимодействия программных компонент комплекса при написании и отладке микропрограммы

2048 слов разрядностью до 64 бит, интерпретируя язык отладки. Этот язык включает команды:

1. Чтение-модификация микрокоманды
(адрес₀)[, (число строк₀)] [(имя поля)] / [(строка)]
(завершитель)
(завершитель) ::= LF | CR | ^

Позволяет прочитать одну или несколько микрокоманд или отдельных их полей, а также изменить последнюю прочитанную микрокоманду или ее поле. Для перехода к следующему большему или меньшему адресу вводится завершитель LF или ^.

2. Разбиение слова микрокоманды на поля
(начальная позиция₀), (длина₀): (имя поля) CR,
Допустимо введение до 128 именованных полей.
3. Чтение-модификация системной информации
□ (ключ) / (новое значение) CR

Дает возможность читать и исправлять основание системы счисления (C) для микрокоманд и данных; точку останова (B); длину слова микрокоманды (W).

4. Чтение системной информации
□ (ключ) / CR

Позволяет прочитать таблицу полей (T), содержимое регистров (R, Q) и СОЗУ (S) МБ, данные (D) на внутренней шине МБ, текущий адрес (A), код в контрольных точках (K), а также всю системную информацию (I).

5. Запись флагов
□ F: (строка) CR

6. Управление
(адрес₀); GCR — запуск микропрограммы с указанием адреса

[(целое₀)]; PCR — запуск с точки останова, (целое₀) — указывает число циклов;
(целое₀); SCR — установка шагового режима, (целое₀) — величина шага;
(целое₀); TCR | (целое₀); DCR — установка режима краткой и расширенной трассировки; (целое₀) — число трассируемых микрокоманд;
; ECR — выход из отладчика;
(строка) — последовательность цифр в установленной системе счисления.

Пользователь может просматривать микропрограмму последовательно по адресам, начиная с любого адреса в любом направлении как по словам в целом, так и по отдельным полям.

Выполнение микропрограммы возможно в шаговом и циклическом режимах. При этом задается величина шага и число проходов через точку останова. При выполнении микропрограммы отладчик загружает регистр I, затем регистр O в АИБ, читая адрес следующей микрокоманды, который эквивалентен индексу в массиве образа микропрограммы.

В распоряжении пользователя краткая и полная трассировка выполняемой микрокоманды, позволяющая индексировать адрес, данные и код в контрольных точках. Для чтения содержимого внутренних регистров и СОЗУ МБ пользователю необходимо написать микроподпрограммы, учитывающие конкретное размещение адресов регистров в микрослове. Микроподпрограммы располагаются за пределами массива образа микропрограммы. Пример протокола отладки приведен на рис. 3.

Все программные компоненты, включая AMDASM, написаны на языке Паскаль и выполняются в среде ОС РАФОС. Транслятор с AMDASM благодаря использованию виртуальной памяти для хранения форматов свободен от ограничений, свойственных некоторым другим реализациям [3] и обладает более высокой скоростью трансляции. Программа-отладчик может воспринимать объектные файлы, созданные как AMDASM, так и самим отладчиком. В последнем случае сохраняется заданная в предыдущем сеансе отладки таблица имен полей. Пользователь может изменять объем сохраняемой микропрограммы и доопределять неиспользованные

```

ВХОДНОЙ ФАЙЛ MIC.OBJ/A/O
ВЫХОДНОЙ ФАЙЛ MIC.DMV/1024
ОБЪЕКТНЫЙ ФАЙЛ MIC.DAT
*#M/ 64
*1,4:A1
*5,4:B1
*1,16:EMIT
*230,5.A1/
  230/ 0000
  231/ 0001
  232/ 1000
*#C/ В Н
*230.EMIT/
  230/ 000F 00FF
*1;S
*1;T
*230;G
  СТАР.АДРЕС= 230 НОВ.АДРЕС= 231
*#K/ 0010001000110011
*#P
  СТАР.АДРЕС= 231 НОВ.АДРЕС= 232
*#P
  СТАР.АДРЕС= 232 НОВ.АДРЕС= 233
*#S
*#B/ 4000 230
*20;D
*230;G
*#P
  СТАР.АДРЕС= 230 НОВ.АДРЕС= 231
  ДАННЫЕ= 00AA К.ТОЧКИ= 0010001000100011
  СТАР.АДРЕС= 231 НОВ.АДРЕС= 232
  ДАННЫЕ= 009A К.ТОЧКИ= 0010001000110011
  СТАР.АДРЕС= 232 НОВ.АДРЕС= 233
  ДАННЫЕ= 008A К.ТОЧКИ= 0010001000110011
  СТАР.АДРЕС= 233 НОВ.АДРЕС= 234
  ДАННЫЕ= 0078 К.ТОЧКИ= 0010001000110000
  СТАР.АДРЕС= 234 НОВ.АДРЕС= 230
  ДАННЫЕ= 00A0 К.ТОЧКИ= 0010001000111011
*#T/
  A1 = 1 4 B1 = 5
  EMIT = 1 16
ВСЕГО ПОЛЕЙ 3
*#I/
W = 64
C = H
ТОЧКА ОСТАНОВА= 230
АДРЕС= 230
N ТРАССИРОВКИ= 20
ОБЪЕМ= 1024
ДАННЫЕ= 00A0
КОНТР.ТОЧКИ= 0010001000111011
*#E
  
```

Рис. 3. Пример протокола отладки

слова с помощью соответствующих ключей. Программа-отладчик создает как выходной файл для последующего сеанса отладки, так и объектный файл для программатора ПЗУ.

Для прожига ПЗУ использовался одноплатный универсальный программатор, имеющий выход на интерфейс «общая шина» и позволяющий программировать все отечественные и некоторые зарубежные типы ПЗУ, ПЛМ и ПМЛ. Настройка программатора на конкретный тип ПЗУ или ПЛМ осуществляется как программно, так и аппаратно — сменой соответствующей «персональной платы», вставляемой в дополнительный разъем программатора.

Использование рассмотренных аппаратно-программных средств для микропрограммирования дисплейного процессора графического терминала на секциях КМ1804ВС1 и КМ1804ВУ1 показало, что ошибок, связанных с временем выполнения, можно полностью избежать при правильном расчете временных соотношений в структуре МБ. Выявление и устранение алгоритмических ошибок и ошибок кодирования выполняется с помощью отладчика. Транслятор применялся для первоначального написания микропрограммы и внесения в нее крупных изменений, накопившихся в результате отладки.

Телефон для справок: 135-42-59, Москва

1. Настраиваемый инструментальный комплекс для разработки систем на секционированных микропроцессорах/О. И. Семенов, Л. А. Гриншпан, Е. М. Злотник и др.//Управляющие системы и машины. — 1984. — № 2. — С. 36—39.
2. Микропроцессоры: системы программирования и отладки/Под ред. В. А. Мясникова и М. Б. Игнатьева. — М.: Энергоатомиздат, 1985.
3. Мета-ассемблер МЕАСС для микропроцессорных систем с наращиваемой разрядностью/Г. П. Мозговой, С. С. Семенова, Е. И. Семин и др.//Микропроцессорные средства и системы. — 1986. — № 3. — С. 20—22.

Статья поступила 23 октября 1986 г.

УДК 681.3.06

С. В. Припоров

ЭКРАННЫЙ ОТЛАДЧИК ДЛЯ МИКРОЭВМ СМ1800

Особенности отладки программного обеспечения

В процессе создания программного обеспечения (ПО) систем с МП наиболее трудоемким этапом является отладка ПО [1], при которой часто приходится выполнять много рутинной работы.

Имеется несколько способов облегчить процесс отладки программ. Наиболее часто разработчики используют аппаратные, программные или комбинированные отладчики. В состав ПО микроЭВМ СМ1800 входят следующие программные средства: монитор-отладчик MONID операционной системы СПО [2]; монитор-отладчик DEBUG операционной системы ДОС1800 [3]; отладчик DDT операционной системы ОС1800 [4]; отладчик DBUG операционной системы микроДОС [5].

Пульт контроля и управления (ПКУ) в ЭВМ можно использовать как аппаратное средство отладки. Перечисленные программные отладочные средства ориентированы на работу по следующему принципу. Каким-либо действием пользователем выполняются при помощи набора директив на клавиатуре дисплея. Директива во время набора одновременно отображается на экране. Когда пользователь ведет отладку, например программой DDT, дисплей работает в режиме «плывущего» экрана. Поэтому со временем часть информации на экране теряется.

В случае ошибки директиву необходимо повторить. На это уходит много времени. Если директивы длинные, то их повторный набор влечет существенные неудобства и является дополнительным фактором возникновения ошибок. Например, команда запуска программы с указанием двух точек останова в отладчике DDT или DEBUG выглядит так:

GXXXX, YYYYY, ZZZZ <BK>

где XXXX — адрес запуска программы; YYYYY — адрес первой точки останова; ZZZZ — адрес второй точки останова. Кроме набора самой директивы необходимо набрать и признак ее конца (обычно символ возврата каретки <BK>).

Работу с отладчиком в режиме набора директив целесообразно сравнить с работой пользователя при наборе текстовой информации. Тексты обычно набираются с помощью программы редактора текстов, имеющегося в составе любой инструментальной ОС [6]. Текст изменяется в пределах одной строки. Для перехода на другую строку набирается соответствующая директива. При этом на экране нет самого текста. Его можно

увидеть только в наборе еще одной директивы. Такой способ редактирования неудобен, особенно при наличии в тексте рисунков [7].

На это было обращено внимание при разработке отладчика. Развитие аппаратных средств вычислительной техники и возможностей ПО привело к созданию так называемых экранных редакторов текстовой информации.

Пользователь при работе с таким редактором получает доступ к информации через окно отображения текста на экране дисплея. В окне располагается либо весь текст, либо его часть (страница). Внутри окна редактируемая информация адресуется при помощи курсора дисплея. Используя функциональную клавиатуру, можно перемещать курсор по тексту. Удобство такого редактора очевидно. Текст располагается на экране как на листе бумаги. Редактирование заключается не в наборе директив, а в наборе самого текста. Такой подход к представлению информации и ее редактированию упрощает процедуру изменения текста, сокращает время работы, уменьшает число вносимых ошибок.

При реализации настоящего отладчика был использован принцип организации диалога, а не отношение к отладке как к процессу редактирования кодов, хотя побочным продуктом такого подхода является возможность использовать отладчик для того, чтобы редактировать двоичный код программ в экранном режиме.

Внешний интерфейс отладчика

Во время работы с отладчиком пользователю предоставляется доступ к отлаживаемой программе при помощи окон отображения. В этих окнах располагаются элементы для просмотра или редактирования (изменения).

Курсор дисплея всегда указывает на элемент, интересующий программиста. К элементам относятся: данные в памяти, команды программы, операнды команд, регистры процессора, точки останова, маркеры. Окно отображения можно перемещать по памяти путем набора команд управления и некоторыми другими командами. Имеется пять окон отображения (рис. 1), которые независимо друг от друга отслеживаются отладчиком по мере их использования.

Первое окно — область памяти микроЭВМ. Окно содержит снимок памяти размером 256 байт (страница) и имеет вид квадратной таблицы в 16 строк по 16 цифр в строке. Каждая пара цифр отображает байт памяти. Слева указаны их физические адреса. Редактирование байта возможно при помощи набора шестнадцатеричных цифр. Изменяется байт, на который ука-

VDT V2.1																	
CODE:DB00																	
MEMR:FB00																	
MEMORY:1000																	
FILENAME:F1:TEST.LDC																	

P5M:0000 BC:..== DKNO 4 ==..PC:0000 SP:DE00 LEVL:01																	
ISTART:1000																	
ISTOP:1011																	

	0	1	2	3	4	B	C	D	E	F	I					
I	1000:21	00	40	00	00	00	00	00	00	00	00	00	I..Q	I	LXI R,4000	
I	1010:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	1020:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	1030:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	1040:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	1050:00	00	00	00	00	00	00	00	00	00	00	00	I	== DKNO R ==	I	NOP	
I	1060:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	1070:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	1080:00	00	00	00	00	00	00	00	00	00	00	00	I	31 32 33 34	I	1234I =DKNO Z =	
I	1090:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	10A0:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	10B0:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	10C0:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	10D0:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	10E0:00	00	00	00	00	00	00	00	00	00	00	00	I	I	NOP	
I	10F0:00	42	43	00	00	00	00	00	00	00	00	00	I	..BC	I	NOP

Рис. 1. Схема расположения окон на экране дисплея

MEMORY:1000
FILENAME:F1:TEST.LOC

```

PSW:0000 BC:0000 DE:0000 HL:0000 M:00 PC:0000 SP:0000 LEVEL:01
  0 1 2 3 4 5 6 7 8 9 A B C D E F
1000:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1000:1XL H:44000
1010:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1003:
1020:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1030:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1040:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1050:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1060:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1070:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1080:00 00 00 00 00 00 00 00 00 00 00 00 00 31 32 33 34 1234
1090:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10A0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10B0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10C0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10D0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10E0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10F0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  
```

Рис. 2. Индикация точек останова

зывает курсор. Перемещать курсор, а значит и менять просматриваемый элемент, можно при помощи функциональной клавиатуры. Если курсор направляется пользователем за границу страницы памяти, находящаяся в данный момент в окне (на экране дисплея), отладчик автоматически переходит на отображение следующей страницы в этом же окне.

Второе окно отображает эквивалент страницы памяти в формате кодов ASCII. Если содержимому памяти не соответствует графический символ, он обозначается точкой. Во втором окне, по необходимости, можно просмотреть точки останова на текущий момент отладки (рис. 2), а также оттранслировать ассемблерные команды в коды микропроцессора (рис. 3).

Третье окно отображает память микроЭВМ в формате программы на языке ассемблера (деассемблирование).

Четвертое окно организовано для индикации и изменения содержимого регистров процессора.

Пятое окно используется для загрузки и сохранения программ, а также групповой интерпретации (выполнение отлаживаемой программы по семь команд после набора одной директивы) команд отлаживаемой программы.

Все директивы работы с отладчиком сведены до набора одного или двух символов. Их можно разделить на директивы: доступа в память ЭВМ и управления экраном дисплея; трансляции / деассемблирования команд микропроцессора (МП); сохранения программ на гибких дисках в формате загрузочного модуля файловой системы DOS1800; загрузки программ с гибких

MEMORY:1000
FILENAME:F1:TEST.LOC

```

PSW:0000 BC:0000 DE:0000 HL:1012 M:40 PC:1000 SP:DE00 LEVEL:01
  0 1 2 3 4 5 6 7 8 9 A B C D E F
1000:21 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 H0#0040
1010:00 03 40 00 00 00 00 00 00 00 00 00 00 00 00 00 H1#0003
1020:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 H2#0000
1030:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 H3#0000
1040:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 H4#0000
1050:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 H5#0000
1060:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 H6#0000
1070:00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 H7#0003
1080:00 00 00 00 00 00 00 00 00 00 00 00 31 32 33 34 H8#0000 1234
1090:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 H9#0000
10A0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10B0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10C0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10D0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10E0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10F0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  
```

Рис. 3. Режим трансляции

дисков; доступа к регистрам процессора; запуска и трассировки программ; определения точек останова.

Отладчик работает в командном и экранном режимах. В командном режиме выполняются все функции, связанные с файловыми операциями по загрузке и сохранению отлаживаемых программ. В экранном — обеспечиваются все остальные функции. В этом режиме за каждой функцией закреплена клавиша на клавиатуре дисплея. Ниже перечислены клавиши, за которыми закреплены функции доступа в память микроЭВМ и управления экраном дисплея. Курсор при выборе этих функций перемещается:

- — на следующий байт;
- ← — на предыдущий байт;
- ↑ — на 16 байт назад;
- ↓ — на 16 байт вперед;
- U/S/N — на следующую страницу;
- U/S/P — на предыдущую страницу;
- V — вперед на 16 страниц;
- Λ — назад на 16 страниц;
-] — на страницу памяти по адресу последней деассемблированной команды;
- [— на страницу по адресу в регистре счетчика команд МП;
- U — на страницу памяти по косвенному адресу. Адрес находится в слове, на младший байт которого указывает курсор в текущий момент отладки. По этой команде запоминается адрес байта в странице, откуда произведен переход (адрес возврата). Возможно 10 таких переходов с сохранением адресов возврата;
- U/S/U — по последнему запомненному директивой U адресу.

Следующие директивы позволяют перейти на страницу памяти по адресам, находящимся в паре регистров BC, DE, HL, SP, PC, M:

- :B — переход на страницу по адресу BC;
- :D — переход на страницу по адресу DE;
- :H — переход на страницу по адресу HL;
- :M — переход по адресу в слове, на которое указывает пара HL;
- :S — переход по адресу в регистре SP;
- :P — переход по адресу в регистре PC;
- :: — возврат на страницу памяти, откуда была выполнена одна из предыдущих директив перехода.

Управление экраном заключается в наличии возможности сбросить экран, очистить отдельные его области в случае, если отлаживаемая программа работает с дисплеем.

К директивам деассемблирования относятся следующие:

- T — деассемблировать команду. При этом курсор терминала перейдет на следующую команду. Деассемблирование производится в окне 3;
- U/S/T — деассемблирование группы из 16 команд в окне 3. Курсор остается на месте. Запоминается адрес начала группы деассемблированных команд и адрес конца группы. Эти адреса имеют символические обозначения BEGIN, STOP и выводятся на экран в окне 3;
- R — деассемблирование группы с маркера STOP. При этом маркер BEGIN становится равным маркеру STOP. Таким образом, обеспечивается страничное деассемблирование с продвижением по памяти ЭВМ. Курсор дисплея указывает на байт с адресом, где была введена первая директива U/S/T.

Команды транслируются в окне 2. В режиме трансляции команд в окне 1 автоматически выводятся коды оттранслированной команды.

Отлаженные программы на гибком диске сохраняются по директиве S. Каждая директива создает файл с загрузочным модулем в формате файловой системы DOS1800. Программа сохраняется сегментами с указаниями адресов начала сохраняемого сегмента в памяти, конца сегмента, загрузки. Адрес загрузки определяет место в памяти, куда будет записываться сохраняемый сегмент при последующем записании программы в память с гибкого диска. Число сохраняемых сегментов в одном файле ограничивается только наличием свободного пространства на диске. Если при сохранении программы было указано имя существующего загрузочного модуля, то возможно добавление к этому модулю сегментов, удаление имеющегося файла и создание нового загрузочного модуля под тем же именем. Операция добавления сегментов позволяет в крайнем случае производить коррекцию загрузочных модулей.

Программы загружаются по директиве L, которая позволяет занести программу в память с гибкого диска со смещением относительно необходимого адреса загрузки. В этом случае программист выбирает необходимое смещение.

Директивы запуска программ (J, UC/J, X, G, UC/G) позволяют организовать отладочные прогоны в трех режимах. Первый режим — с передачей управления отлаживаемой программе и возврат в отладчик через команду RSTS. При этом останавливается 10 точек останова, определяемые до запуска программы. При многократном запуске указывать каждый раз точку останова нет необходимости. Второй режим — при помощи интерпретации каждой команды. Третий режим — запуск указанной программы в качестве подпрограммы, вызываемой из отладчика. В этом режиме число точек останова не ограничивается.

Доступ к регистрам процессора (A, B, C, D, E, H, L, M, PC, SP) обеспечивается директивами UC/A, UC/B, UC/C, UC/D, UC/E, UC/H, UC/L, UC/M, P, S соответственно. Для автоматического отображения регистров при выполнении любой директивы запуска программы или по директиве UC/R всегда используется окно 4.

Кроме перечисленных имеются функции пересылки данных, заполнения памяти константами. Отладчик имеет защиту собственных данных от неспреднамеренной порчи со стороны работающего программиста.

Сравнительный пример

Фрагменты работы программиста при отладке одной и той же программы различными отладчиками. Пример не дает полную картину процесса отладки в экранном режиме, которая может быть получена при непосредственной работе за дисплеем. Тем не менее программистам он позволяет получить некоторое представление о работе отладчика.

Первый фрагмент относится к работе с отладчиком операционной системы ОС1800. Второй — показывает возможности экранного отладчика. Пусть имеется оттранслированная подпрограмма:

== DOS-MACROASSEMBLER-80 V.4.1 == GET PAGE

LOC	OBJ	LINE	SOURCE STATEMENT
4000		1	ORG 4000H
		2	
		3	EXTRN MOVS ;ОБЪЯВИТЬ ВНЕШНЮЮ П/К
		4	PUBLIC GET ;ОБЪЯВИТЬ ТОЧКУ ВХОДА
		5	
		6	ВХОД: DE- АДРЕС БЛОКА ОПЕРАНДОВ
		7	HL- АДРЕС МАССИВА РЕЗУЛЬТАТА
		8	
4000	E5	9	GET: PUSH H ;СОХРАНИТЬ 'HL'
		10	
4001	210800	11	LXI H,8 ;ВЫЧИСЛИТЬ АДРЕС
4004	19	12	DAD D ;1-ГО ОПЕРАНДА
		13	
4005	4E	14	MOV C,M ;ЗАГРУЗИТЬ ПЕРВЫЙ
4006	23	15	INX H ;ОПЕРАНДА
4007	46	16	MOV B,M
		17	
4008	23	18	INX H ;ЗАГРУЗИТЬ ВТОРОЙ
4009	7E	19	MOV A,M ;ОПЕРАНДА
		20	
400A	E1	21	POP H ;ВОСТАНОВИТЬ 'HL'
400B	CD000	E	CALL MOVS ;ПЕРЕСЫЛКА
400E	C9	23	RET
		24	
		25	END

ASSEMBLY COMPLETE, NO ERRORS

Программа MOVS пересылает массив с адреса в паре BC по адресу в паре HL, длина массива находится в регистре A. Все регистры сохраняются. Необходимо выполнить следующие действия по отладке:

просмотреть область памяти по адресу в регистрах DE при входе в программу;

запустить программу с адреса GET с точкой останова по адресу команды CALL MOVS;

проверить правильность выбранного адреса в регистре BC;

проверить значение регистра A;

просмотреть данные по адресу в HL;

запустить программу с адреса команды CALL MOVS с точкой останова на команде RET;

проверить содержимое регистров;

проверить содержимое памяти по адресу в HL.

Пусть адрес метки GET 4000H, команды CALL MOVS 400BH, команды RET 400EH. Будем также считать, что программы уже загружены в память.

Директивы отладчика DDT выглядят так:

- X (BK) — отобразить регистры;
- DMMMM, AAAA (BK) — отобразить область памяти (MMMM — содержимое регистра DE, AAAA — конечный результат), адрес области для отображения;
- G4000, 400B (BK) — загрузить н/п с точки останова;
- X (BK) — отобразить регистры;
- DMMMM, AAAA (BK) — отобразить область памяти по адресу в регистрах BC;
- G400B, 400E (BK) — загрузить программу MOVS;
- X (BK) — отобразить регистры;
- DMMMM, AAAA (BK) — отобразить область по адресу HL.

Всего требуется набрать 62 символа. Директивы экранного отладчика выглядят так:

- : D — отобразить область памяти по адресу (страница 256 байт);
- :: — вернуться на программу по адресу 4000H;
- XXXXXXXX — выполнить девять команд покомандно до команды CALL MOVS. После каждой команды регистры автоматически отображаются в области 3
- : H — просмотреть данные по адресу HL;
- : : — возвратить на адрес команды CALL MOVS;
- XJX — выполнить подпрограмму CALL MOVS;
- : H — просмотреть данные по адресу HL.

Во втором случае требуется ввести 22 символа. Регистры процессора отображаются автоматически после выполнения любого действия по запуску программ. На этом примере можно убедиться, что работа с клавиатурой в режиме экранной отладки более эффективна. Необходимо также заметить, что при работе с экранным отладчиком программист ни разу не набирал физических адресов памяти.

Опыт эксплуатации отладчика

Двухгодичная эксплуатация экранного отладчика показала его высокую эффективность. Удобное представление информации для пользователя в экранном режиме сокращает число символов, вводимых им с клавиатуры дисплея в процессе диалога, уменьшает число ошибок, допущенных при отладке программ.

Число символов, набираемых при работе с отладчиком, примерно на 25 % меньше, чем при работе с отладчиком DDT [4].

Наличие такого средства, как командное выполнение программы с отображением счетчика команд процессора посредством курсора терминала, дает значительную наглядность и удобство в работе.

Отладчик оказался особенно полезным в процессе обучения системе команд МП КР580ИК80 и программированию на языке ассемблера. Возможно его применение в качестве учебной программы по изучению МП КР580ИК80; при этом нет необходимости использовать дополнительную аппаратуру.

Отладчик работает с дисплеями ВТА-2000-15, ВТА-2000-30, VDT52100, MERA в режиме фиксированного формата. Возможна адаптация на другие типы дисплеев. Отладчик позволяет полностью отказаться от отладочного ввода и вывода в программах. Имеется вариант программы отладки, работающий в ППЗУ объемом 4,5К байт.

Телефон для справок: 7-54-75, Дзержинск, Горьк. обл.

ЛИТЕРАТУРА

1. Фокс Дж. Программное обеспечение и его разработка. — М.: Мир, 1985.
2. СПО. Монитор. Руководство программиста. 4.072.127ИЭ.
3. DOS1800. Описание применения. 2/Б/. 00054-01 31 01.
4. ОС1800. Отладчик. Руководство программиста. 2/Б/.00035-01 33 05.
5. МикроDOS. Руководство оператора. — М.: МЦНТИ, 1985.
6. ОС1800. Редактор текстовой информации. Руководство программиста. 2/Б/.00035-01 33 02.
7. Лебедев Г. В. Разработка интерактивных программ на основе принципа непосредственного редактирования информации/Микропроцессорные средства и системы. — 1986, — № 1. — С. 44.

Статья поступила 6 августа 1986 г.

УДК 681.3.06

В. И. Бровко

КОМПЛЕКС ДЛЯ ОТЛАДКИ ИЗДЕЛИЙ НА ОСНОВЕ ОДНОКРИСТАЛЬНОЙ ЭВМ K1816BE51

Комплекс предназначен для отладки в реальном масштабе времени аппаратной части и программного обеспечения микроконтроллеров на основе однокристальных ЭВМ K1816BE51 (18051).

Память данных на кристалле ЭВМ и внешняя память не используются для записи и хранения программ из-за отсутствия набора команд и сигналов записи (если, конечно, не предусмотрены дополнительные аппаратно-программные средства для объединения или переклещивания ОЗУ между полями памяти программ и данных), а также возможности выбора кода программ из памяти данных. Поэтому в микроконтроллерах на основе этой ЭВМ фиксированная программа размещается в ПЗУ (здесь же могут находиться и всевозможные таблицы), а переменные — в памяти данных с эффективными способами адресации каждой из них.

Совместно с процессором на кристалле K1816BE51 находятся ПЗУ 4К байта, ОЗУ 128 байт (в том числе 4 банка по 8 регистров общего назначения), регистры специальных функций, два таймера-счетчика, двухканальный высокоскоростной (до 1 Мбит/с) последовательный порт, 2-уровневая структура приоритетного прерывания, 32 двунаправленные линии ввода-вывода, объединенные в четыре порта. Все эти средства имеют мощную программную поддержку, важнейшей особенностью которой является набор инструкций так называемого булевого процессора, позволяющего выполнять над битами внутренней памяти данных и регистров специальных функций (аккумулятора, портов ввода-вывода и т. д.) те же операции, что и над байтами: логические операции, операции пересылки, установки, сброса, инвертирования, ветвления программы в зависимости от значения коакретного бита. Кроме того, ЭВМ имеет команды поддержки двоичной и двоично-десятичной арифметики, в том числе умножение и деление.

Из 111 команд, использующих 11 типов адресаций, 75 выполняется за 1 мкс, умножение и деление за 4 мкс, а остальные — за 2 мкс. Большинство команд (95) одно- и двухбайтовые. Возможности ЭВМ позволяют создавать на их основе компактные, высоконадежные микросистемы и локальные сети для решения задач управления и сбора информации. При реализации алгоритмов управления, использующих собственный ресурс ЭВМ, объем программ и время их вы-

полнения в несколько раз меньше, чем при использовании микропроцессоров КР580ИК80, К1810ВМ86, микроЭВМ «Электроника 60» или ДВК. При расширении системы дополнительной памятью данных и портами ввода-вывода эффективность ее работы снижается. Работа с внешней памятью программ не сказывается на быстродействии, но при этом становится невозможным использование портов P0 и P2 для операций ввода-вывода.

Создание и доводка систем, работающих в реальном масштабе времени, сопряжена с рядом препятствий. Прежде всего, трудно гарантировать стопроцентную правильность функционирования программного обеспечения и аппаратуры при отладке с помощью тест-программ, аппаратных имитаторов, а тем более кросс-отладчиков. Дело в том, что для правильной работы подобных систем важно не только значение опеределенного разряда входного порта (например, линии, подключенной к датчику или концевыку) или выходного управляющего, но также их длительность и относительное временное расположение. Это не всегда удается правильно определить на стадии создания алгоритма. Кроме того, на практике возникают такие ситуации, когда система алгоритмически функционирует правильно, но в определенные моменты происходят сбои. Для их устранения требуется не только внесение конструктивных изменений или дополнений в аппаратную часть, но и введение в нужных местах программы эмпирически определяемых временных задержек (например, с целью завершения переходных процессов в электрических цепях и в механических узлах). Все это требует работы с реальным объектом, причем часто для выявления всех погрешностей нужна наработка на аварийную или сбойную ситуацию, не говоря уже о том, что требуется многократное перепрограммирование ПЗУ. Предлагается вниманию комплекс, решающий подобный круг задач.

Программное обеспечение для проектируемых устройств разрабатывается на универсальной микроЭВМ типа CM 1800 (CM 1810) с помощью развитых средств редактирования под управлением операционной системы, совместимой с ISIS-II (CP/M или MS-DOS). Компиляция исходной программы и настройка на абсолютные адреса осуществляются с помощью кроссассемблера АСМ-51 или кросскомпилятора языка высокого уровня (например, ПЛ/М-51 или СИ-51). Подготовленный таким образом объектный файл исходной программы по соответствующей директиве отладчика со стороны CM 1800 перегружается в ОЗУ программ отладочного модуля, где и производится отладка программы в реальном масштабе времени на реальном объекте или имитаторе процесса под управлением оператора с клавиатуры CM 1800 или с пользовательского пульта (рис. 1). Подключение к объекту управления осуществляется через системную шину и дополнительный разъем. Для расширения отладочного модуля можно использовать практически без переделок изделия, имеющие типоразмеры двойной европлаты с системной шиной И-41 (MULTIBUS-I), изготовленные для других типов 8-разрядных микропроцессоров.

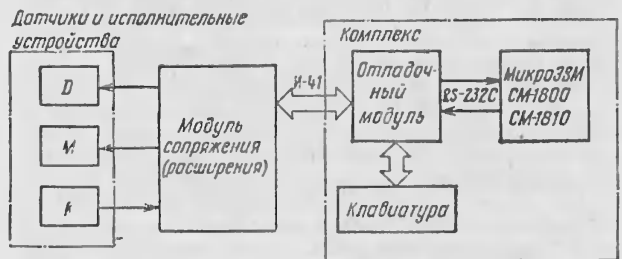


Рис. 1. Схема подключения комплекса к исполнительным устройствам

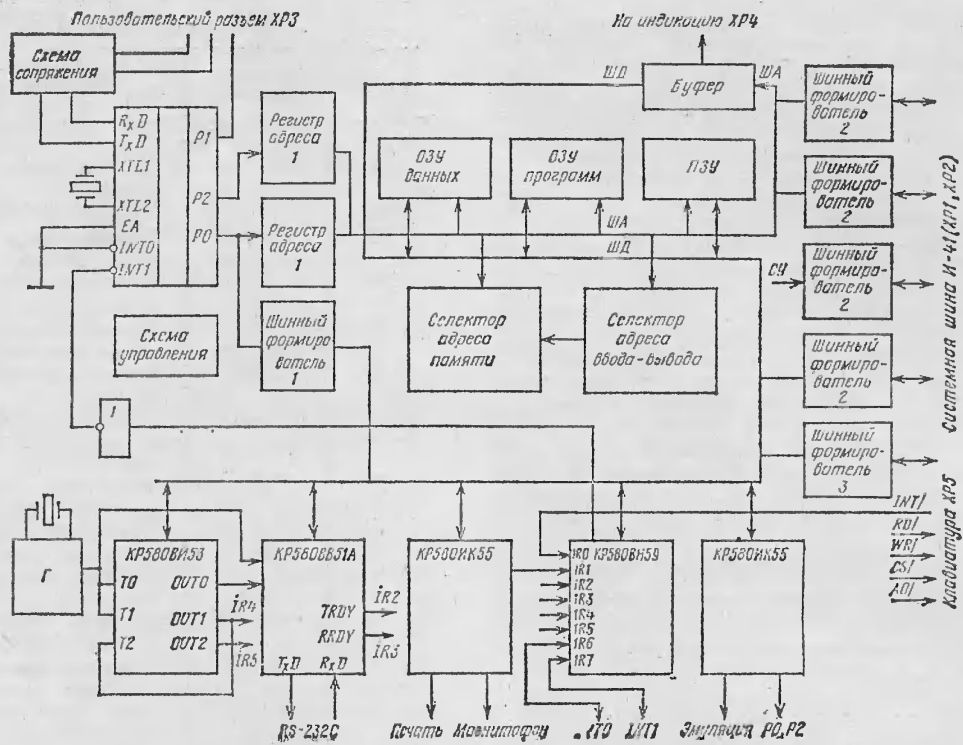


Рис. 2. Структурная схема отладочного модуля

Отладочный модуль (рис. 2) состоит из: центрального процессора (ОЭВМ К1816ВЕ51); энергозависимых ОЗУ программ и ОЗУ данных по 4К байта (К537РУЗ); ПЗУ 4К байт с монитором и служебными подпрограммами (К573РФ2) (зарезервированы два места под ПЗУ пользователя К573РФ2 или К573РФ4); двух схем параллельного интерфейса КР580ИК55 — одна осуществляет связь с печатающим устройством типа УВВПЧ-30-004 и магнитофоном для записи (воспроизведения) и управления лентопротяжным механизмом («Маяк-232»), другая предназначена для моделирования портов Р0 и Р2, которые в отладочном модуле задействованы под организацию магистралей данных и адреса; БИС последовательного обмена КР580ВВ51А, необходимой для связи с ОЭВМ и для решения задач пользователя при работе с одной из факультативных клавиатур; БИС программируемого таймера КР580ВИ53, нулевого канала, который задает скорость обмена по последовательному каналу (два других используются в качестве системных таймеров); контроллера приоритетных прерываний КР580ВН59; схемы сопряжения последовательного канала. Последняя предназначена для подключения отладочного модуля к локальной сети типа Bitbus, основанной на протоколе SDLC фирмы IBM. В данном варианте предусмотрена возможность для работы в двух режимах: синхронном и самосинхронизацией. При этом сигналы как информационные, так и синхронизации, представляют собой дифференциальные пары с электрическими характеристиками, основанными на стандарте RS-485. Основные характеристики отладочного модуля приведены ниже.

Кроме того, в состав комплекса входят две факультативные клавиатуры (рис. 3). Одна выполнена с использованием микросхем малой степени интеграции и

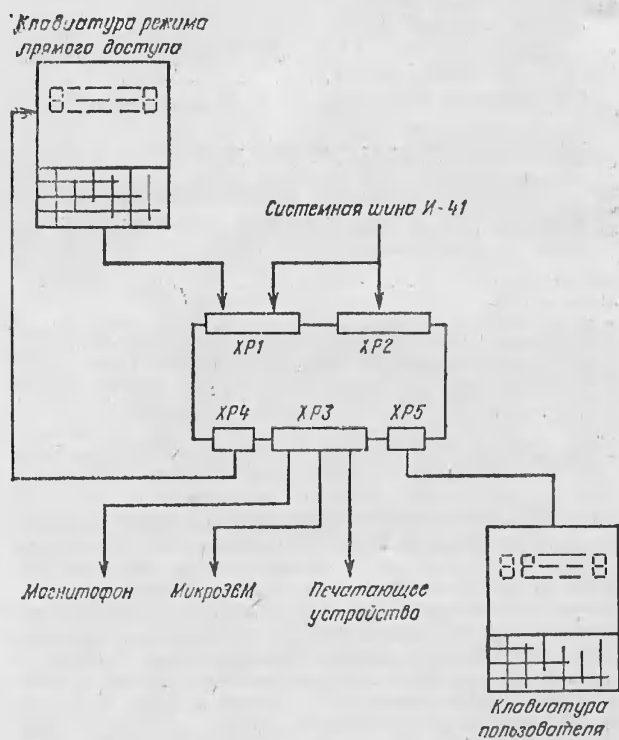


Рис. 3. Схема подключения устройств к отладочному модулю

работой в режиме прямого доступа ко всему полю внешней памяти, в том числе ко всем периферийным БИС. В этом случае на отладочный модуль подается сигнал управления, который переводит регистры адреса 1, шинный формирователь 1 в третье состояние, формируется сигнал сброса для K1816BE51. Шинные формирователи 2 при этом переключаются для приема с системной шины адреса, данных и сигналов управления (см. рис. 2). Индикация осуществляется через буферную схему, подключенную к разъему ХР4.

Другая клавиатура (пользователя), выполненная на базе КР580ВВ79, работает под управлением резидентного монитора ОС51.1. Шинные формирователи 1 служат буферами адреса и данных при выходе на системную шину в работе с пользовательским модулем сопряжения или расширения. Связь с клавиатурой осуществляется через шинный формирователь 3, подключенный к разъему ХР5. Сигнал CS для КР580ВВ79 формируется на отладочном модуле, что делает клавиатуру в определенной степени менее универсальной, но при этом длина кабеля сопряжения умножается вдвое, так как не требуются адресные линии А1...А15 и отсутствует дополнительный селектор адреса на плате клавиатуры.

Основные характеристики отладочного модуля

Разрядность данных, бит	8
Быстродействие, млн. операций/с	1
Объем памяти данных:	
внутренней, байт	128
внешней, Кбайт	4 (64)*
Объем памяти программ:	
внутренней, Кбайт	4**
внешней, Кбайт	12 (64)*
Число регистров:	
общего назначения	32
специальных функций	20
Число двунаправленных линий ввода-вывода	32
Последовательные каналы:	
пользовательский дуплексный	
максимальная скорость передачи, Мбит/с	1
формат данных, бит	8,9
число режимов работы	4
системный (КР580ВВ31А)	RS-232C
Таймеры-счетчики:	
пользовательские	
число	2
разрядность	16
число режимов работы	4
системные (КР580ВВ53)	
Прерывания:	
пользовательские	
число источников	5
число уровней приоритета	2
системные (КР580ВВ59)	
Число инструкций процессора	111
Число режимов адресации	11
Программное обеспечение:	
языки программирования	АСМ-51, ПЛ/M-51
монитор	ОС51.1

- * Максимально возможный объем. В режиме К1816ВЕ31 (EA=0) внутренняя память программ отключена, внешняя память программ адресуется 0000H...FFFFH (64К байт).
- ** В режиме К1816ВЕ51 (EA=1) ОЭВМ работает также и с внутренней памятью программ, лежащей в адресном пространстве 0000H...OFFFH (4К байта). Внешняя память программ в этом случае адресуется 1000H...FFFFH (60К байт). Максимально возможный объем внешней памяти данных в обоих случаях составляет 64К байт (0000H...FFFFH).

В ОЭВМ К1816ВЕ51 имеется несколько адресных пространств с разными механизмами доступа, командами обращения и сигналами управления: программная память на кристалле, память данных на кристалле, регистры специальных функций, внешняя память программ, внешняя память данных. В мониторе реализованы свои индивидуальные команды для работы с каждой из этих областей (см. таблицу). Кроме того, в отладочном комплексе ОЗУ данных и ОЗУ программ могут находиться по одним и тем же адресам (в данном случае оба расположены в поле 0000H...OFFFH), причем реализована возможность записи в обе области аппаратно-программным способом (в мониторе

Команды монитора ОС51.1

Клавиша	Мнемоника команд	Выполняемая функция
SD	SD (SD,)	Модификация ОЗУ программ (данных)
SC	SC	Индикация памяти программ
SI	SI	Модификация внутренней памяти данных
RF	RF	Модификация регистров специальных функций
S	S	Выполнение программы в шаговом режиме
G	G1	Начать выполнение программы по адресу 1
	G1 — G2	Начать выполнение программы по адресу 1 с остановом по адресу 2
F	F (F,)	Заполнить константой ОЗУ программ (данных)
C	C (C,)	Переместить массив в памяти программ (данных)
L		Работа с последовательным каналом
	LI (LI,)	Принять массив и разместить в памяти программ (данных)
	LO (LO,)	Передать массив из памяти программ (данных)
P	PL (PL,)	Распечатать массив из памяти программ (данных)
	PI (PI,)	Считать массив с магнитофона в ОЗУ программ (данных)
	PO (PO,)	Записать массив из ОЗУ программ (данных) на магнитофон
INI	—	Прерывание выполняемой программы
—	D (D,)	Вывести на экран в шестнадцатеричном коде

для соответствующих команд имеются по две модификации).

Внешняя память программ (рис. 4) состоит из ОЗУ, системного и пользовательского ПЗУ. Ячейки с адреса-

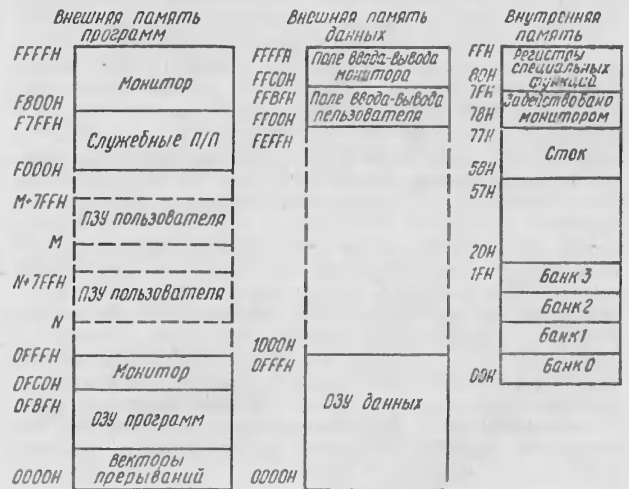


Рис. 4. Карта распределения памяти отладочного модуля

ми ООН ... 42Н зарезервированы для векторов прерываний с записью значений в эти ячейки монитором при начальной инициализации. Верхние 64 байта ОЗУ программ (в данном случае OF СОН ... OFFFH), место положения которых определяется в зависимости от объема памяти, задействованы монитором при работе. Причем при отсутствии ОЗУ программ монитор использует для этой цели те же ячейки ОЗУ данных. ПЗУ пользователя объемом 4К байта (K573PФ2) или 16К байт (K573PФ4) может располагаться в любом месте адресного пространства, ограниченного IOOOHEFFFH. Системное ПЗУ с монитором ОС51.1 и со служебными подпрограммами находится по адресам FOOOH ... FFFFH (4К байта). Остальная часть пространства памяти программ может быть использована разработчиком при расширении отладочного модуля.

Внешняя память данных состоит также из нескольких физически отличных областей: ОЗУ данных (OOOON ... OFFFH), областей портов ввода-вывода системы (FFCOH ... FFFFH) и адресации портов пользователя (FFOO ... FFBFH), находящихся на плате сопряжения (расширения). Обращение к этим портам осуществляется так же, как и ко внешнему ОЗУ данных при помощи команд косвенной адресации через указатель DPTR или регистр R0 (R1) одного из банков с предварительной установкой страницы в регистре специальных функций P2. Область адресации портов пользователя не фиксирована по объему и расположению в поле памяти. Но с точки зрения оптимальности написания программ желательно, чтобы адреса системных и пользовательских портов находились в одной странице. Тогда при обращении к ним не требуется перезагрузка указателя страницы регистра P2, если, конечно, он не использовался для адресации памяти.

Внутренняя память данных разбита на поле стека (SP-57H — начальная инициализация указателя стека), поле (78H ... 7FH), зарезервированное для работы монитора при выполнении директив и двадцать регистров специальных функций (аккумулятора, портов ввода-вывода, PSW и т. д.), находящихся в адресном пространстве 80H ... FFH.

Механизм функционирования монитора ОС51.1 при работе с пользовательской клавиатурой основан на опросе и анализе состояния периферийных схем. Выполнение директив при задании их ведущей микроЭВМ CM 1800 базируется на обработке соответствующих векторов прерываний микросхемы KP580BH59, на вход которой поступают запросы на прерывание от исполнительных устройств со временем реакции ~12 мкс. Ограничений на использование какого-либо регистра специальных функций нет, а из 128 байт внутренней памяти данных только ячейки с адресами 78H ... 7FH необходимы при реализации директив S и G1-G2 плюс дополнительно до 10 байт стека.

Среди команд монитора ОС51.1 (см. таблицу), достаточно типичных для подобного рода программ, есть команды, предназначенные для организации обмена с использованном системной БИС KP580BV51A (последовательный канал, расположенный на кристалле ЭВМ, отдал в распоряжение пользователя). Для их выполнения задаются начальный и конечный адреса области ОЗУ программ или данных. Затем осуществляется пересылка массива информации со скоростью 9600 Бод и индикацией каждого значения адреса и байта информации. Существенно, что команды просмотра или модификаций любой области памяти работают с инкрементированном или декрементированном адресе.

На клавиатуре пользователя имеется клавиша INT, которая подключена ко входу IRO контроллера прерываний KP580BH59, имеющего наивысший приоритет. При замыкании ее происходит прерывание выполняемой программы с индикацией на дисплее клавиатуры адреса и кода команды, которая будет выполнена по возвращении из процедуры обработки прерывания, и предо-

ставляется возможность модификации внутреннего содержимого ЭВМ.

Несколько слов об ограничениях, накладываемых комплексом на использование ресурса ЭВМ K1816BE51. Прежде всего, это необходимость в эмуляции портов P0 и P2, задействованных под магистраль данных и адреса отладочного модуля, при отладке систем, работающих с собственным ресурсом, т. е. наращивания внешней памятью данных или программ. В этом случае инструкции работы с битами или байтами регистров специальных функций P0 и P2 нужно заменить на макрокоманды эмуляции этих портов. Поэтому здесь с известной степенью условности можно говорить об отладке в реальном масштабе времени. Для систем, работающих с внешней памятью данных или программ, ограничения состоят только в эмуляции прерываний INT0 и INT1, которые задействованы соответственно для реализации директивы S и приема запроса прерывания от контроллера KP580BH59. Если при отладке программ с пульта директива S (step) не используется, то возможно переключение входов прерывания микроЭВМ для использования разработчиком непосредственно по назначению.

Применение ЭВМ K1816BE51 особенно эффективно прежде всего в управляющих системах для отслеживания внешних асинхронных событий и сбора информации для привязки процесса управления к реальному времени и построения многоканальных ЦАП и АЦП. Считывание значений АЦП с частотой порядка 200 кГц можно организовать с использованием только внутренней памяти данных.

Реализация сложных булевых функций на K1816BE51 с использованием логических инструкций работы с битами дает выигрыш в объеме программы и времени выполнения в несколько раз.

Телефон для справок: 435-13-33, г. Киев
Статья поступила 5 мая 1987 г.

УДК 681.326—181.4

Е. И. Альперович, В. Г. Белый, С. И. Большинский

ДВУХМАШИННЫЙ ОТЛАДОЧНЫЙ КОМПЛЕКС ДЛЯ РАЗРАБОТКИ УСТРОЙСТВ НА БАЗЕ БИС СЕРИИ KP580

При разработке программного обеспечения (ПО) для встраиваемых систем контроля на базе МПК БИС KP580 в качестве инструментальной и отладочной можно использовать микроЭВМ «Электроника K1-10». Однако ограничения, налагаемые перфоленточной операционной системой МОС1 (слабые возможности периферийных устройств, отсутствие средств работы с библиотеками), значительно снижают эффективность процесса. Авторами предлагается двухмашинный комплекс, объединяющий микроЭВМ «Электроника K1-10» и ДВК-2М.

Линия связи между двумя микроЭВМ (рис. 1) — это интерфейс для асинхронного последовательного обмена данными в двух направлениях. Она подключена к ДВК-2М через устройство последовательного обмена (УПО) 15ВВВ-60/9600—003. Адреса внешних устройств задаются: регистр состояния ввода (176560), входной буфер (176562), регистр состояния вывода (176564), выходной буфер (176566).

К микроЭВМ «Электроника K1-10» линия связи подключается через устройство сопряжения FS-ПЛ или

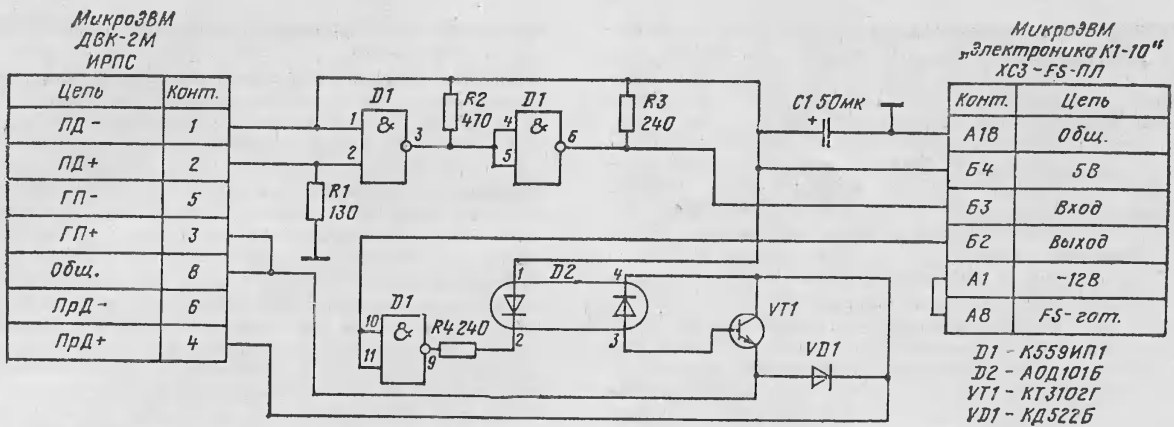


Рис. 1. Схема сопряжения микроЭВМ «Электроника К1-10» и комплекса ДВК-2М

ввода-вывода (УВВ) микроЭВМ «Электроника К1-10». Скорость передачи данных — 9600 Бод. Схему сопряжения через «Токовую петлю 20 мА» (см. рис. 1) можно использовать и для связи микроЭВМ «Электроника К1-10» с дисплеем, имеющим интерфейс «Стык С1».

Предлагаемое решение расширяет возможности микроЭВМ «Электроника К1-10»: позволяет использовать в качестве системной консоли дисплей вместо электрической пишущей машинки (ЭПМ) «Консул 260.1» (авторы использовали алфавитно-цифровой дисплей 15ИЭ-00—013) и делает доступными для микроЭВМ периферийные устройства комплекса ДВК-2М.

Двухмашинный комплекс в совокупности со средствами редактирования и кросскомпиляции для микропроцессоров серии КР580 под управлением операционной системы (ОС) РАФОС значительно повышает эффективность разработки и отладки ПО. Полученный на ДВК-2М под управлением ОС РАФОС объектный код хранится на гибком диске в виде файла. Для отладки объектный код по линии связи передается в ОЗУ микроЭВМ «Электроника К1-10», а откорректированный при отладке объектный код — на диск.

Разработанное ПО линии связи включает в себя доработанный базовый монитор и драйвер устройства, определенный пользователем (для микроЭВМ «Электроника К1-10»); драйвер линии связи (для комплекса ДВК-2М).

Доработка базового монитора обусловлена особенностями системы ввода-вывода микроЭВМ «Электроника К1-10». По правилам МОС1 при включении в систему ввода-вывода нестандартного устройства в специальной области ОЗУ необходимо сформировать команду перехода на драйвер этого устройства и соответственно изменить слово ввода-вывода (СВВ). В модифицированном варианте инициатора монитора заложена возможность подключения устройства ввода-вывода и пульта, определенных пользователем. Изменение стандартной конфигурации системы ввода-вывода зафиксировано в СВВ. Это избавляет пользователя от необходимости корректировать его с пульта.

Для нормального завершения работы комплекса ДВК-2М при приеме информации, драйвер линии связи должен получить сигнал признака конца передачи. Поскольку МОС1 допускает расширение базовой системы команд монитора, для формирования и передачи в линию связи признака конца введена команда «У».

Один и тот же алгоритм описывает работу драйвера устройства и пульта (определенных пользователем), реализует прием и передачу последовательной 8-битовой посылки по правилам МОС1.

Программное обеспечение для микроЭВМ «Электроника К1-10» (размещено в ОЗУ либо в ПЗУ микроЭВМ) написано на ассемблере (рис. 2, 3).

```

-----ЗАГОЛОВОК-----
КРОССАССЕМБЛЕР С80805РАФОС
; ИНИТ- МОДИФИКАЦИЯ БАЗОВОЙ ПРОГРАММЫ МОНИТОР МОС1. НАЗНАЧАЕТ ЛОЖЧЕСКО-
; ЯННУ УСТРОЙСТВ ВВОДА-УСТРОЙСТВО ВВОДА, ОПРЕДЕЛЕННЫЕ ПОЛЬЗОВАТЕЛЕМ; ЛОЖ-
; ЧЕСКОМУ УСТРОЙСТВУ ВЫХОДА-УСТРОЙСТВО ВЫХОДА, ОПРЕДЕЛЕННОЕ ПОЛЬЗОВАТЕЛЕМ;
; ЭЛЕКТРОННОМУ ПУЛЬТУ-ЭПМ "КОНСУЛ 260.1", ЛИБО ПУЛЬТ, ОПРЕДЕЛЕННЫЙ ПОЛЬЗОВА-
; ТЕЛЕМ В ЗАВИСИМОСТИ ОТ ГОТОВНОСТИ ЭПМ.
-----ПРОГРАММА-----

INIT:  OUT    I      ;СБРОС ААРЕСНОЙ ШИНЫ
       LXI    SP,100H ;ОПРЕДЕЛЕНИЕ ВЕРХНЕЙ ГРАНИЦЫ ОЗУ
       CML   @F003H
       MVI   A,H
       STA   B7      ;ЗАПИСЬ СТАРШЕГО БИТА ГРАНИЦ ОЗУ В 7 ЯЧЕЙКУ
       LXI   D,@F027H;ФОРМИРОВАНИЕ В ОЗУ ШАБЛОН ПРОГРАММЫ
       D    D        ;ВЫСОКА ИЗ МОС
       MOV   M,A
       INR  JZ      RES
       INX  D
       JMP  LOOP1
RES:   MVI   MRA
       MOV   M,A
       INR  L
       JNZ  LOOP1
       MVI   L,@05H
       SPHL ;ОРГАНИЗАЦИЯ СИСТЕМНОГО СТЕКА
       LXI  H,100H
       PUSH H
       MOV  L,H
       PUSH H
       PUSH H
       PUSH H
       IN   ANH      ;ЕСТЬ ГОТОВНОСТЬ КОНСУЛА?
       ANI  B2
       JNZ  H
       MVI  A,@6AH  ;ЖЕЛ ПУЛЬТ ПОЛЬЗОВАТЕЛЯ
       JRP  SUV
M:     MVI   A,@2BH  ;ПУЛЬТ-КОНСУЛ
       STA  Z        ;ФОРМИРОВАНИЕ СВВ
       MVI  A,@03H  ;ФОРМИРОВАНИЕ ВЕКТОРА ПЕРЕМЫКАНИЯ
       STA  S
       LXI  D,@F46H
       SHLD J        ;ФОРМИРОВАНИЕ КОМАНД ПЕРЕХОДА НА ДРАЙВЕРЫ ВНЕШНИХ УСТРОЙСТВ
;ОПРЕДЕЛЕННЫХ ПОЛЬЗОВАТЕЛЕМ
       LHLD B        ;ДРАЙВЕР УСТРОЙСТВА ВВОДА
       MVI  L,@56H
       LXI  D,@D0W
       CML  @DR0W
       MVI  L,@53H  ;ДРАЙВЕР ВЫХОДА НА ПУЛЬТ
       LXI  D,@D0LT
       CALL @DRDR
R:     MVI  L,@50H  ;ДРАЙВЕР ВВОДА С ПУЛЬТА
       LXI  D,@D0LT
       CALL @DRDR
       MVI  L,@SCH
       D,@DR0W
       CALL @DRDR
       LXI  D,@D156H;ФОРМИРОВАНИЕ АДРЕСА ДИСПЕТЧЕРА РАСШИРЕНИЯ
       CALL @D15P
DISP: MVI   @F041H ;ПЕРЕХОД НА ДИСПЕТЧЕР МОС
       MOV  C,@B  ;ОБРАБОТКА КОМАНД "У"
       CPI  C3H   ;ПРИНЯТ КОД "У"?
       JZ   @F0
       JMP  @F03DH;НЕТ ПЕРЕХОДА НА ОБРАБОТКУ ОШЕБОК МОС
EOF:   MVI   C,@1AH;ПЕРЕАТЬ ПРИЗНАК КОНЦА ДАННЫХ
       CALL @D1AH
       MVI  C,@1AH
       CALL @DR0W
       JMP  @F041H  ;ПЕРЕХОД НА ДИСПЕТЧЕР МОС

;ПОДПРОГРАММА ФОРМИРОВАНИЯ АДРЕСА ПЕРЕХОДА
;В РЭН АДРЕС ОЗУ.
ADRDR: MVI   @,@03H
       MOV  M,A
       L   M,A
ADR15P:MOV   M,E
       INR L
       MOV M,D
       RET

```

Рис. 2. Текст модифицированного инициатора монитора

нию связи шестнадцатеричный файл А.ОВЖ из области ОЗУ микроЭВМ «Электроника К1-10» на гибкий диск комплекса ДВК-2М. Введенная с пульта микроЭВМ «Электроника К1-10» команда «У» завершит сеанс передачи данных. По соответствующим командам МОС1 и РАФОС можно передать данные в обратном направлении.

Двухмашинный комплекс ориентирован на пользователя, знакомого с ОС МОС1 и РАФОС (специального обучения не требуется). Он повысил эффективность разработки и отладки ПО для устройств на базе БИС серии КР580, показав при этом высокую надежность комплекса.

Адрес для справок: 340048, г. Донецк, ул. Розы Люксембург, 72; Донецкий физико-технический ин-т АН УССР

Статья поступила 15 марта 1987 г.

УДК 681.325.5—181.4

В. Я. Ефремов

СИГНАТУРНЫЙ АНАЛИЗАТОР

Просмотр сигнатур схож с отысканием неисправностей в аналоговых устройствах (усилителях, телевизорах и т. п.). На принципиальных схемах этих устройств в характерных точках приводятся осциллограммы напряжений сигналов и числовые значения напряжений. Сравнивая эти осциллограммы с изображением, наблюдаемым на экране осциллографа, и делают заключение о работе испытываемого узла.

Для МП-системы контроль работы, основанный на сопоставлении осциллограмм, не может быть использован, поскольку все двоичные последовательности на экране осциллографа практически неразличимы. К тому же в МП-системах нет однозначного соответствия между характеристиками устройства и его конкретными узлами.

Подобно логическим анализаторам сигнатурные анализаторы [1] регистрируют потоки цифровой двоичной информации. Но в отличие от логических анализаторов, которые только наглядно представляют информационные каналы, сигнатурные анализаторы обрабатывают «длинные» потоки двоичной информации, «сжимая» их с высокой достоверностью. Получаемые «короткие» форматы — кодовые эталоны — именуются сигнатурами. Содержание сигнатур носит формальный характер и наличие определенной сигнатуры в некоторой точке схемы свидетельствует о конкретном распределении битов информации в потоке данных, регистрируемом в течение заданного интервала времени.

Сигнатурный анализ — сопоставление реальной сигнатуры (в конкретной точке устройства), отображенной на дисплее сигнатурного анализатора, с эталонной сигнатурой этой точки (указанной на схеме или в таблице).

Сигнатурный анализатор (рис. 1), простой в изготовлении и не нуждающийся в настройке, выполнен авторами на микросхемах серии К155 и содержит три печатные платы. Размер и вес его значительно меньше, чем у аналогичных приборов: отечественного (типа 817 [2]) и зарубежного (типа 5004А [3]).

Рассмотрим, как формируется сигнатура [4]. Аппаратурно «сжимает» информацию шестнадцатеричный регистр сдвига и пятиходовый сумматор (по модулю 2). На четыре входа этого сумматора подаются сигналы обратной связи от регистра сдвига, а на пятый — регистрируемый поток двоичной информации. Выход сумматора по модулю 2 подключен на вход регистра сдвига,

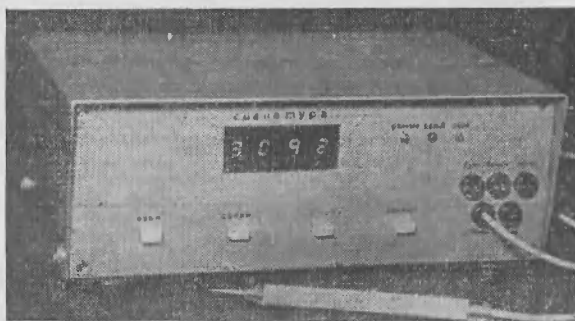


Рис. 1. Сигнатурный анализатор

Математическая операция, осуществляемая такой схемой, соответствует последовательному делению входного потока информации на характеристические полиномы четырех вложенных регистров, образуемых разрядом обратной связи [5]. Сдвиг информации, поступающей в регистр, осуществляют синхронимпульсы тестируемой МП-системы. Оттуда же в сигнатурный анализатор поступают еще два сигнала «Пуск» и «Стоп», формирующие интервал времени поступления информации на регистр сдвига. Поступивший сигнал называется тест-последовательностью. Количество бит его информации в конечном итоге будет представлено 4-разрядной сигнатурой. Это и есть «сжатие» информации в сигнатурном анализаторе [6].

Пример [1] — формирование сигнатуры для 20-битной последовательности 11111000001111111111 (рис. 2). Вначале (такты 0...7) регистр сдвига работает, как обычно. На такте 7 первая единица исходной последовательности доходит до первого отвода обратной связи. Эта единица по цепи обратной связи передается на вход сумматора по модулю 2 и складывается с очередным битом входной последовательности, имеющим нулевое значение. Вследствие этого в такте 8 в первый разряд регистра сдвига будет занесена 1, а не

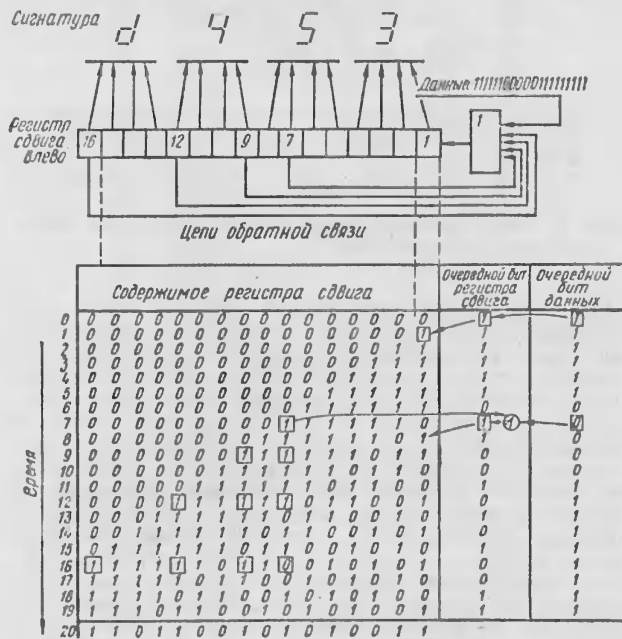


Рис. 2. Образование сигнатуры на 16-разрядном регистре сдвига

элемент D12 на сдвигающий вход 16-разрядного регистра D15. Сигналы тест-последовательности поступают на информационный вход этого регистра, предварительно пройдя через активный входной шуп D4, элементы D6, D9 и J—K триггер D11. Эти сигналы тактированы с синхрипульсами МП-системы, и поэтому осциллограммы импульсов на входе регистра D15 (рис. 4, з) несколько отличаются от исходной (рис. 4, е).

Наличие в МП-системе элементов с тремя состояниями несколько усложняет входное согласующее устройство D4. Это устройство (активный шуп при контроле трехстабильных элементов) передает на вход элемента D11 третье состояние, как то же самое («Лог. 0» или «Лог. 1»), которое имел последний действующий бит тест-последовательности. После окончания измерительного интервала, когда на сдвиге регистра образовалась сигнатура, запускается схема формирователя D5, которая вырабатывает импульс «Сравнение сигнатур» (1 мкс). За время действия этого импульса сравниваются коды восьми младших разрядов двух чисел: шестнадцатеричного числа — сигнатуры (образовалась в сдвиге регистра D15) и сигнатуры, полученной в предыдущем цикле (хранится в регистре D18). Сравнивают их 16-входовый элемент D19 и 8-входовая логическая схема «ИЛИ» D20. Если коды не совпадают, на выходе элемента D20 появляется «Лог. 1» (поступает на вход элемента D22 одновременно с импульсом «Сравнение сигнатур»). В результате этого на выходе схемы D22 возникает отрицательный импульс, запуска-

ется ждущий мультивибратор D23 и вспыхивает светодиод HL3.

Схема контроля равенства сигнатур, полученных в двух последовательных циклах, обнаруживает динамические сбои в работе МП-системы. Сразу после импульса «Сравнение сигнатур» запускается формирователь D8, вырабатывающий импульс «Занесение в промежуточную память». Под воздействием этого импульса (1 мкс) содержимое сдвигового регистра D15 пересылается в регистр хранения сигнатуры D18, содержимое которого благодаря дешифратору D21 высвечивается на 4-разрядном 7-сегментном индикаторе НГ1.

По спаду импульса на выходе формирователя D8 элементами D10 и D13 образуются сигналы установки регистра D15 в состояние «Лог. 0». Время установки не превышает 0,5 мкс, т. е. через 2,5 мкс сдвиговой регистр сигнатурного анализатора готов к приему очередной тест-последовательности и формированию новой сигнатуры.

Для увеличения гибкости прибора активные перепады (фронт или спад) каждого из сигналов, образующих измерительный интервал, могут выбираться переключателями SB1...SB3. Тест-последовательность контролируется светодиодом HL2, а поступление синхросигналов на сдвиговой регистр — светодиодом HL1. При правильном подключении сигнатурного анализатора к МП-системе и запуске тест-программы светодиоды HL1 «Сдвиг» и HL2 «Данные» должны светиться (а не мигать!).

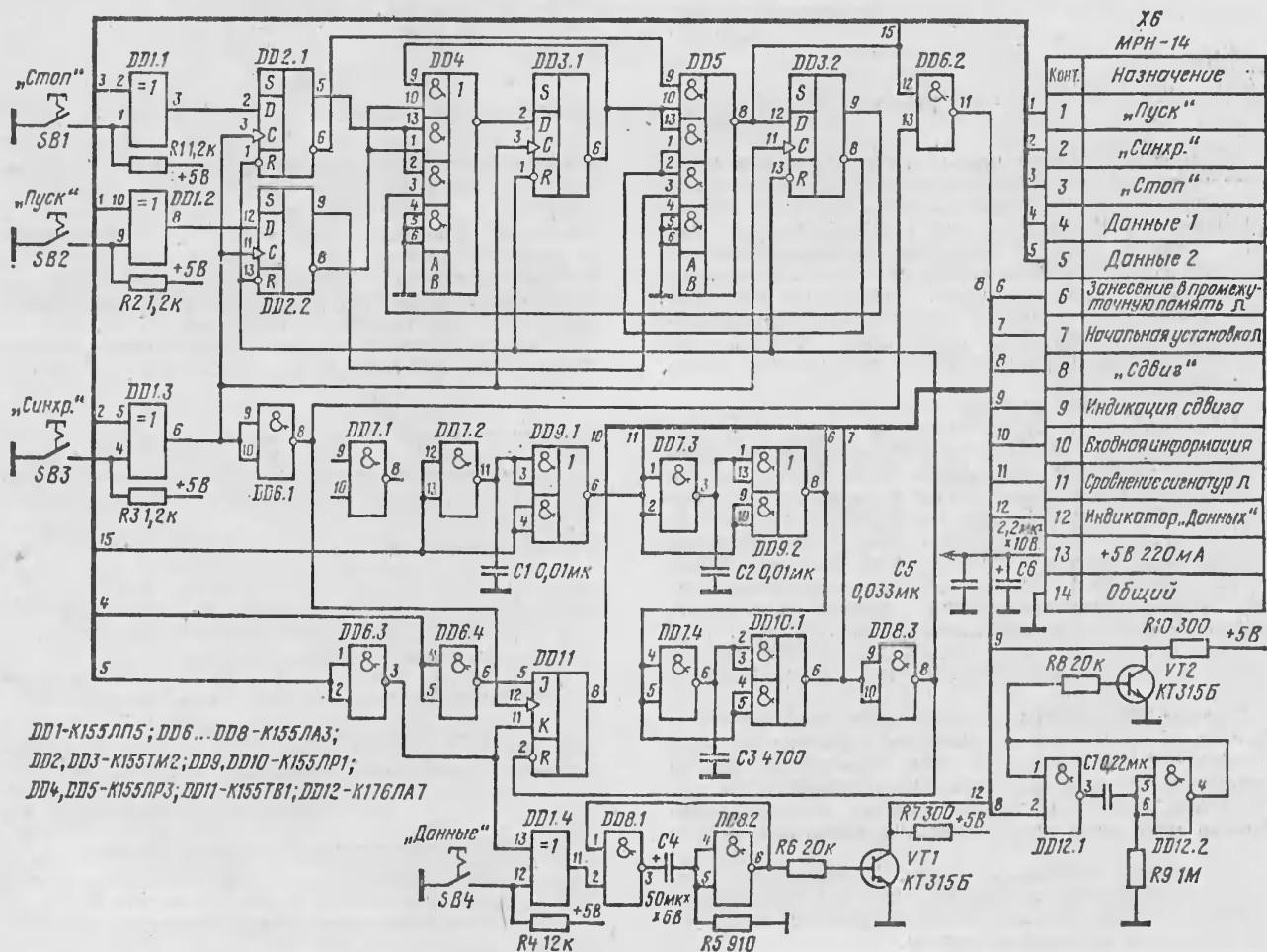


Рис. 5. Входной шуп сигнатурного анализатора

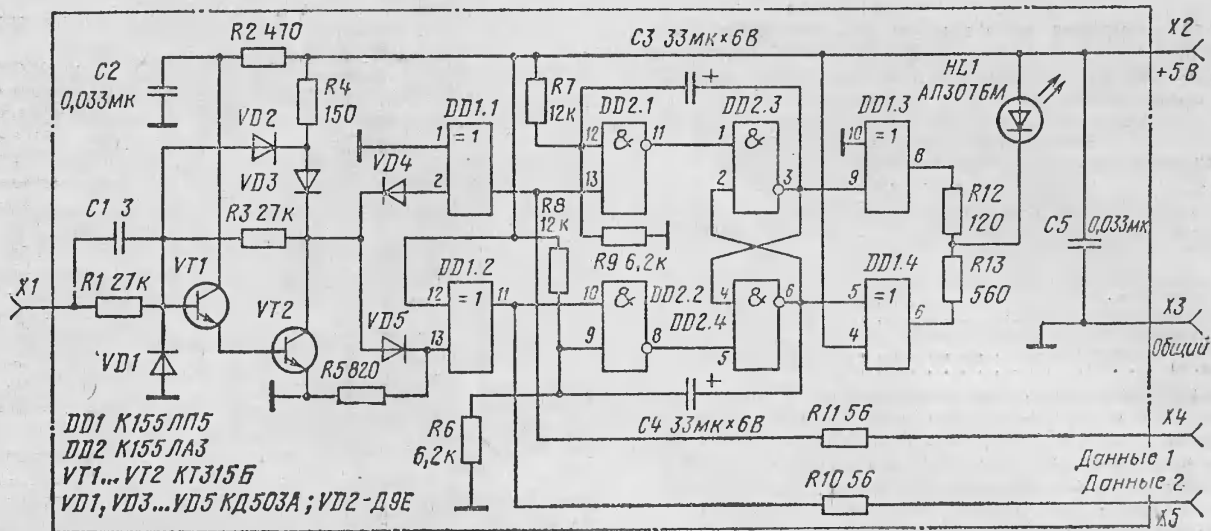


Рис. 6. Входная плата сигнатурного анализатора

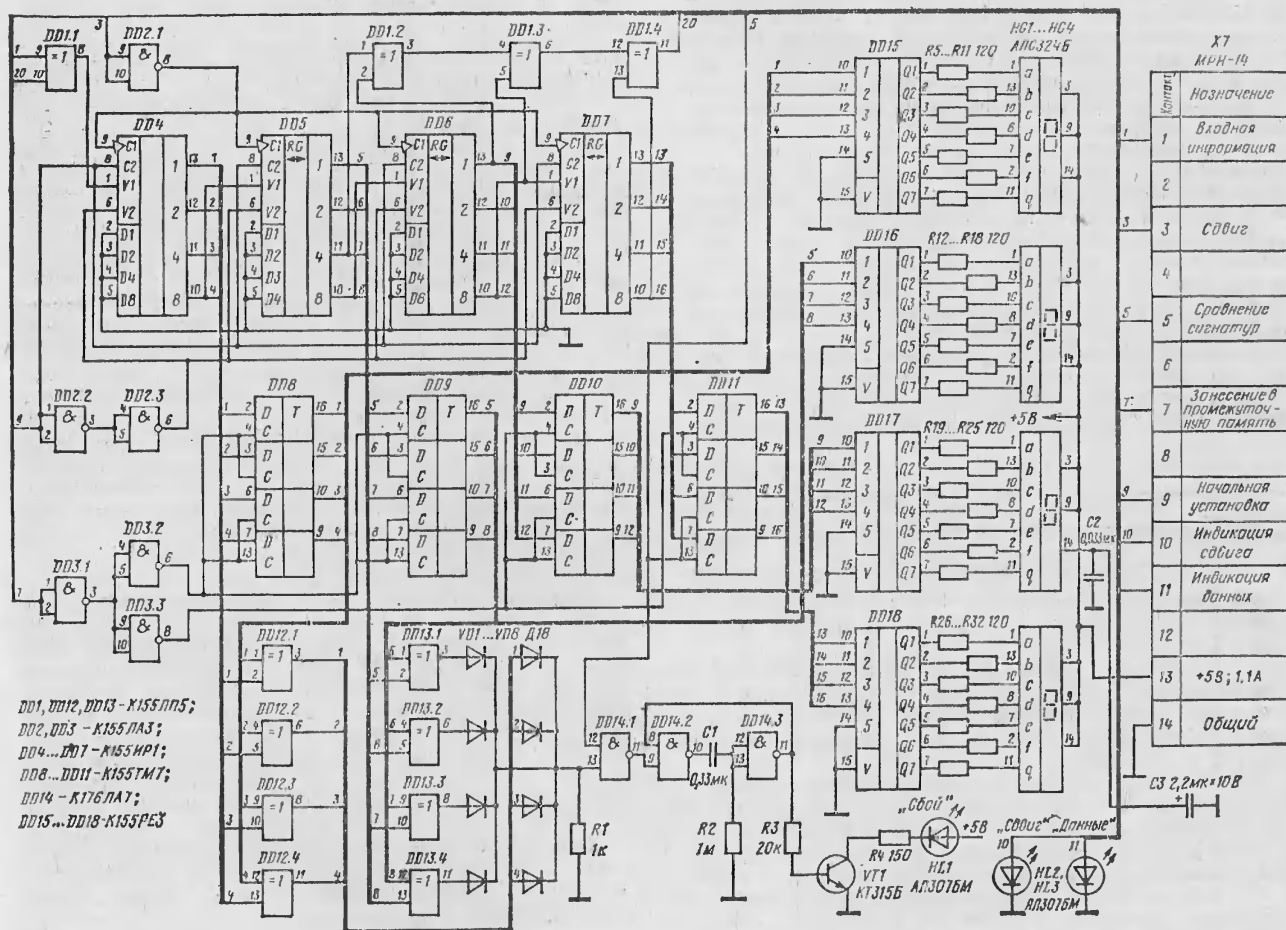


Рис. 7. Плата формирования сигнатур

Если снимаемая сигнатура в каждом цикле тест-программы имеет одно и то же значение, индикатор HL3 «Сбой» не светится. Его мигание говорит о наличии динамических ошибок в контрольной точке.

Основные технические характеристики сигнатурного сигнализатора

Максимальная частота тактовых импульсов, МГц	10
Уровни входных сигналов	ТТЛ
Нагрузка на сигналы с выходов объекта контроля: «Пуск», «Стоп», «Синхр.»	1 вход ИС К155-ЛП5
Входное сопротивление канала «Данные», КОМ	27
Минимальное время между циклами формирования сигнатур, мкс	2,5
Напряжение питания, В	+5±5%
Ток, потребляемый по цепи питания, А	1,3
Габаритные размеры, мм	260×170×90
Масса, кг	3,2

Сигнатурный анализатор реализован из 28 микросхем серии К155 и двух микросхем типа К176ЛА7 на двух печатных платах (есть еще плата стабилизатора +5 В).

На входной плате сигнатурного анализатора (рис. 6) собрано устройство формирования измерительного интервала (на микросхемах DD2... DD6). Его работа, исключая внесенные сигнатурным анализатором искажений в тест-последовательность, подробно описана [5].

Переключатели SB1...SB3 позволяют выбрать желательный активный уровень входных сигналов, которые через элементы DD1.1...DD1.3 будут формировать измерительный интервал.

Три последовательно включенных устройства задержки (на логических элементах DD7.2...DD9.1, DD7.3...DD9.2, DD7.4...DD10.1) вырабатывают сигналы управления работой платы формирования сигнатур. Первые два каскада задерживают сигнал запуска на 2 мкс (по 1 мкс каждый), последний — на 0,5 мкс. Устройство задержки запускаются по спаду импульса измерительного интервала. Схема активного щупа сигнатурного анализатора приведена на рис. 5.

Плата формирования сигнатур (рис. 7) содержит 16-разрядный сдвиговый регистр (микросхемы DD4...DD7), промежуточную память (DD8...DD10). Для отображения сигнатуры в шестнадцатеричном коде в качестве дешифраторов используются четыре микросхемы ПЗУ типа К155РЕ3. Содержимое первых шестнадцати ячеек каждой из микросхем DD15...DD18 показано в таблице.

Сигнатуры, полученные в двух последовательных циклах, сравнивает устройство на микросхемах DD12...DD13 и диодах DD1...DD8. При несовпадении кодов в момент прихода сигнала «Сравнение сигнатур» запуска-

Таблица

Адрес ячейки ПЗУ	Содержимое ПЗУ							Символ
	1	2	3	4	5	6	7	
0 0 0 0	0	0	0	0	0	0	1	0
0 0 0 1	1	0	0	1	1	1	1	1
0 0 1 0	0	0	1	0	0	1	0	2
0 0 1 1	0	0	0	0	1	1	0	3
0 1 0 0	1	0	0	1	1	0	0	4
0 1 0 1	0	1	0	0	1	0	0	5
0 1 1 0	0	1	0	0	0	0	0	6
0 1 1 1	0	0	0	1	1	1	1	7
1 0 0 0	0	0	0	0	0	0	0	8
1 0 0 1	0	0	0	0	1	0	0	9
1 0 1 0	0	0	0	1	0	0	0	А
1 0 1 1	1	1	0	0	0	0	0	Б
1 1 0 0	0	1	1	0	0	0	1	С
1 1 0 1	1	0	0	0	0	1	0	Д
1 1 1 0	0	1	1	0	0	0	0	Е
1 1 1 1	0	1	1	1	0	0	0	F

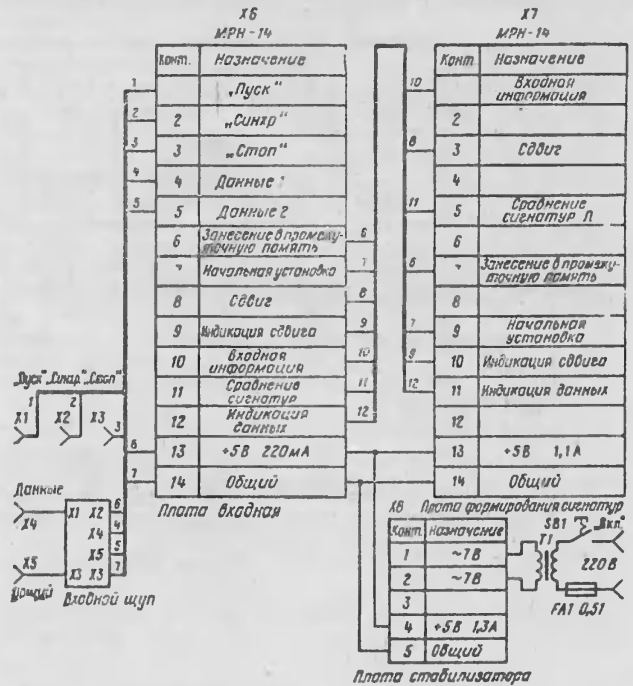


Рис. 8. Схема соединений сигнатурного анализатора

ется ждущий одновибратор (DD14.2...DD14.3) и вспыхивает светодиод HL1 («Сбой»).

Схема соединений трех плат сигнатурного анализатора показана на рис. 8.

Работа с сигнатурным анализатором. Общие вопросы диагностики МП-систем сигнатурным анализатором освещены подробно [1, 8], поэтому рассмотрим работу сигнатурного анализатора конкретно, когда тест-программа и эталонные сигнатуры известны, а точки подключения входов сигнатурного анализатора указаны в инструкции по проверке.

После запуска тест-программы на передней панели сигнатурного анализатора должен гореть светодиод «Сдвиг», а на индикаторном табло появиться какое-нибудь число. Это будет свидетельствовать о поступлении на входы анализатора сигналов «Пуск», «Синхр.» и «Стоп». Перед снятием сигнатур полезно убедиться в исправности сигнатурного анализатора. Это легко сделать, если коснуться щупом общего провода (на вход «Данные» подать «Лог. 0») — на индикаторном табло должны появиться нули. Если же входным щупом дотронуться до шины питания +5 В (т. е. на вход «Данные» подать «Лог. 1»), то на индикаторе должна быть определенная сигнатура.

Подключая входной щуп к различным точкам МП-системы, сверяют получаемые сигнатуры с записанными в таблице или на схеме. Если в какой-либо точке сигнатуры не совпадают, переходят к следующей контрольной точке в соответствии с алгоритмом контроля. Последняя контрольная точка, в которой результат плохой (а в следующей он уже хороший), и есть та самая, вблизи которой существует неисправность. Неисправный элемент (микросхема) тот, на входах которого сигнатуры совпадают с эталонными, а на выходе не совпадают. Этот элемент подлежит замене или ремонту.

Сигнатурный анализ как метод диагностики МП-систем значительно снижает эксплуатационные издержки, а главное, повышает производительность труда ремонтного персонала [9].

Адрес для справок: 127427, Москва, ул. Академика Королева, 12, ТТЦ им. 50-летия Октября, ОПЗ и РО; тел. 215-69-29.

ЛИТЕРАТУРА

1. Гордон, Надиг. Локализация неисправностей в микропроцессорных системах при помощи шестнадцатеричных ключевых кодов//Электроника. — 1977. — С. 23—33.
2. Проспект международной отраслевой выставки «Связь-86». Советские экспонаты. Сигнатурные анализаторы.
3. Hewlett-Packard. Electronic Instruments and Systems. Каталог. — 1983. — С. 132.
4. Кирьянов К. Г., Соловейчик Э. Б. К проектированию РЭА, ориентированной на диагностику сигнатурным анализом//Техника средств связи. — 1980. — Вып. 1 (26).

5. Кудряшов В. И., Новак Г. Х. Сигнатурные анализаторы как средства формализации наладки и эксплуатации дискретной техники//ПСУ. — 1984. — № 3. — С. 25—26.
 6. Мирский Г. Я. Микропроцессоры в измерительных приборах. — М.: Радио и связь, 1984, с. 147—158.
 7. Гилмор Ч. Введение в микропроцессорную технику. — М.: Мир, 1984. — С. 319—323.
 8. Микропроцессоры. Системы программирования и отладки/Под ред. В. А. Мясникова, М. Б. Игнатьева. — М.: Энергоатомиздат, 1985. — С. 41—53.
 9. Сигнатурный пробник АМЦ 0566 для контроля цифровых устройств//ПСУ. — 1985. — № 6. — С. 4 обл.
- Статья поступила 27 января 1987 г.

УДК 681.32

В. П. Воевудский, В. В. Каштанов, О. С. Семенова, А. А. Фокин

СИСТЕМА ДИАГНОСТИРОВАНИЯ В ПЕРИФЕРИЙНОМ ПРОЦЕССОРЕ «ЭЛЕКТРОНИКА МС 1603»

Быстродействующий периферийный процессор (БПП) «Электроника МС 1603» [1] — программируемый специализированный процессор (Фурье-процессор) для обработки сигналов в системах реального времени. Он подключается к межмодульному параллельному интерфейсу микроЭВМ «Электроника 60». Базовый пакет БПП включает 32 стандартные подпрограммы, среди которых подпрограммы быстрого комплексного и действительного преобразования Фурье, свертки и корреляции. Операции над массивами осуществляются в формате с обычной 16-разрядной и двойной точностями.

Технические характеристики БПП

Число разрядов	16
Способ представления данных	с фиксированной занятой
Емкость программ ОЗУ, 32-разрядных слов	512
Емкость данных ОЗУ, К слов	32
Максимальный объем адресуемой памяти данных, К слов	25
Цикл обращения к памяти данных, нс	200
Время параллельного выполнения операций сложения и умножения, нс	200

Достоинство конструкции БПП — функциональная законченность каждого из четырех блоков элементов (240×280 мм), установленных в комбинированном блоке (132×483×627 мм):

Блок интерфейса БЭ АП-01 организует протокол обмена с межмодульным параллельным интерфейсом (МПИ) в программном режиме, а также в режимах прямого доступа и прерываний при обмене данными с прямоадресуемыми регистрами БПП. Он выполняет обмен данными с памятью данных БПП и некоторые функции управления ходом вычислительного процесса в БПП (режимы старт-стоп, обмен данными с внутренними регистрами БПП и т. д.);

Время выполнения стандартных алгоритмов, поставляемых с БПП

Алгоритмы	Время обработки массива из 1024 комплексных точек, мс
Быстрое преобразование Фурье (БПФ)	13
Обратное быстрое преобразование Фурье (ОБПФ)	13
Свертка	18
Корреляция	18
Умножение массивов с обычной точностью	1,0
Умножение массивов с двойной точностью	1,2
Комплексное умножение массивов	1,4

Блок управления БЭ АП-02 генерирует коды микрокоманд, управляющих ходом вычислительного процесса и адреса обращения к памяти данных БПП;

Блок арифметики БЭ АП-03 — вычислитель, осуществляет арифметические и логические операции над данными;

Блок памяти данных БЭ АП-04. Две группы блоков: 1) интерфейса и памяти данных; 2) управления и арифметики — функционально независимы.

Особенность БПП — наличие системы диагностирования, использующей средства встроенной диагностики [2]. Такая система (рис. 1) обеспечивает автоматизированный поиск и локализацию неисправности с точностью до функционального узла, а в отдельных случаях до микросхемы.

Средства встроенной диагностики включают диагностический регистр,

управляющий генератором тактовых импульсов (ГТИ), и сдвиговой регистр, предназначенный для хранения состояния 48 диагностических сигналов, называемых контрольными точками (КТ).

Режим проверки средствами встроенной диагностики (ДГН) проводится при остановленном генераторе БПП, а дальнейшая работа происходит под управлением микроЭВМ через регистр диагностики путем задания определенного числа вырабатываемых синхронных импульсов и уровня замириания (0 или 1).

В режиме ДГН в микроЭВМ могут быть считаны данные с шин ШД, ШК и данные о состоянии 48 КТ, загружаемых в сдвиговой регистр с выработкой каждого синхронного импульса. Доступ к данным на ШК позволяет осуществлять контроль над выборкой команд БПП из памяти программ, а информация о состоянии диагностических сигналов — обнаружить и локализовать неисправность при выполнении команды.

Программную часть диагностической системы составляют:

программа «тестовый монитор», управляющая диагностической системой;

библиотека макроопределений, предназначенная для автоматизации разработки диагностических программ. Функции этих макроопределений выполняются либо целиком с помощью соответствующих макрорасширений, либо с помощью подпрограмм резидентной библиотеки, встроенной в тестовый монитор;

диагностические программы; командный файл, обеспечивающий выполнение диагностических программ в последовательности, соответствующей расширяющемуся диагностическому ядру.

Тестовый монитор функционирует в среде операционной системы ФОДОС и обеспечивает загрузку тестов с внешних устройств и их запуск, выполнение командных файлов, а также предоставляет возможность задавать и оперативно изменять режимы выполнения тестов.

В монитор встроены программа «Отладчик», обеспечивающая доступ к регистрам БПП, загрузку, выполнение и отладку программ в кодах

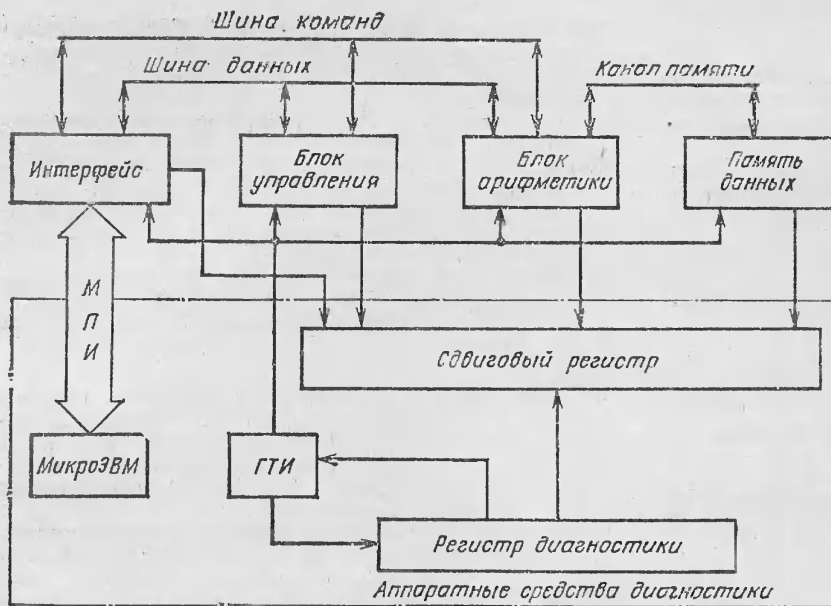


Рис. 1. Структура БПП

БПП и резидентная библиотека подпрограмм, вызов которых осуществляется макрорасширениями соответствующих макроопределений, входящих в состав макробиблиотеки.

Диагностические программы (тесты) проверяют БПП на функциональном уровне и в режиме ДГН. Диагностическая система предоставляет возможность выбора одного из следующих режимов проверки средствами встроенной диагностики:

проверка в режиме ДГН в случае обнаружения ошибки на функциональном уровне;

обязательная проверка в режиме ДГН независимо от результатов проверки на функциональном уровне;

запрет выполнения проверки в режиме ДГН (проверка осуществляется только на функциональном уровне).

Кроме того, разрешается печать данных с ШК и ШД и состояния всех диагностических сигналов, полученных с помощью средств встроенной диагностики, независимо от результатов проверки.

В общем случае в диагностических программах тестирование осуществляется в следующем порядке:

после выполнения программы в БПП микроЭВМ проверяет результаты на функциональном уровне;

при обнаружении ошибки БПП переводится в режим ДГН, а та же программа выполняется в БПП по полутактам;

в каждом полутакте генератора считываются и проверяются данные с ШК и состояние диагностических сигналов.

В основе алгоритма, реализующего проверку в режиме ДГН, программная дешифрация кода команды, счита-

ваемого с ШК в микроЭВМ с помощью средств встроенной диагностики. Команды БПП (32-разрядные) хранятся в памяти программ и состоят из нескольких (до пяти) операций, выполняемых одновременно в течение одного или двух тактов генератора БПП (длительность такта — 200 нс) (рис. 2). В первой половине такта происходит выборка старшей части кода команды (разряды 31, ..., 16), во второй — младшей части (разряды 15, ..., 00). Результат программной дешифрации — генера-

ция тестовым монитором (резидентной библиотекой подпрограмм) базы данных, состоящей из номеров и эталонов состояний контрольных точек, которые должны быть проверены при выполнении этой команды.

Для обеспечения дешифрации каждой операции БПП поставлены в соответствие номера КТ, непосредственно или косвенно участвующих в выполнении операции, и эталонные состояния КТ (эталон КТ). В считанной с ШК команде определяются операции, из которых она состоит, и формируется объединенная БД из данных для каждой операции.

При выполнении программы БПП выборка текущей команды совмещена с выполнением предыдущей. Поэтому при проверке средствами встроенной диагностики создаются базы данных для выбираемой команды (БДВК) и исполняемой команды (БДИК).

Алгоритм обеспечивает простоту проверки средствами встроенной диагностики (рис. 3), сокращает объемы программ, освобождает от трудоемких операций по подготовке эталонов диагностируемых сигналов и связанных с этим ошибок.

Программа проверки выполняет следующие действия:

считывает и проверяет данные на ШК, при этом эталон кода команды хранится в диагностической программе (ошибки, обнаруженные на этом этапе, являются серьезными и должны устраняться в первую очередь);

дешифрует код команды, полученный с ШК (формируются номера и эталоны КТ, которые проверяются при выполнении этой команды в следующем такте);

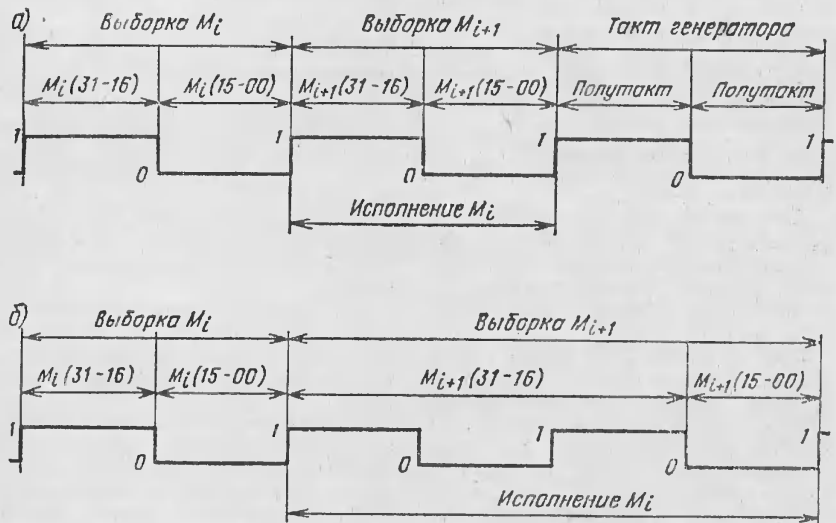


Рис. 2. Выборка и выполнение команд в БПП:

а — выполнение команды M_i за один такт генератора БПП, б — за два такта (точка остановки генератора БПП: уровень замораживания $P3=1$, $P3=0$)

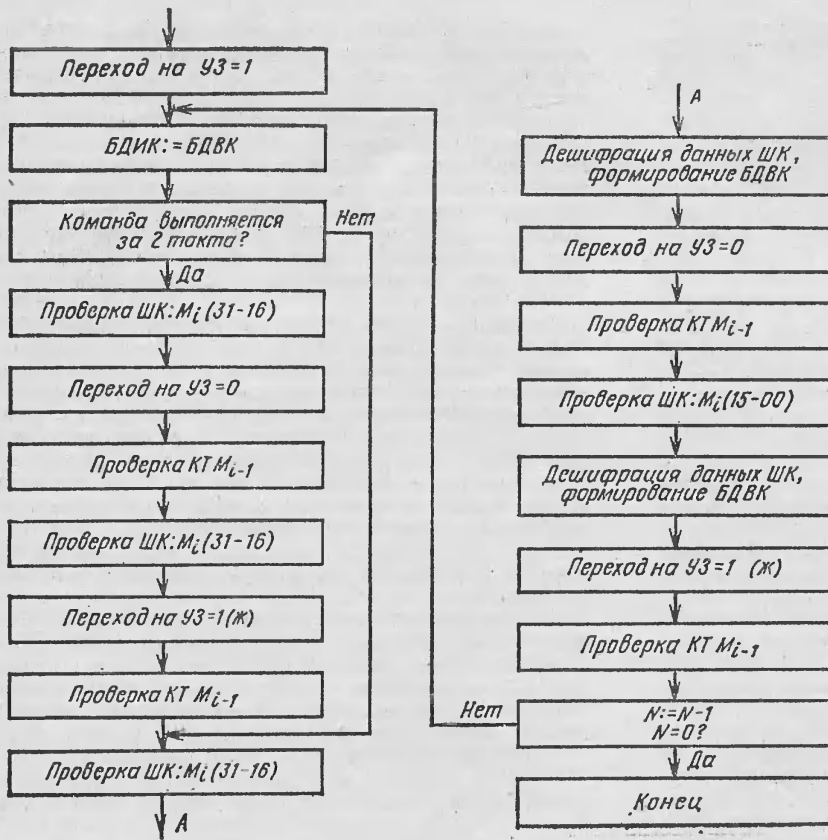


Рис. 3. Алгоритм проверки БПП средствами встроенной диагностики

считывает и проверяет состояния КТ, номера и эталоны которых были сформированы в процессе дешифрации в предыдущем такте.

При обнаружении несовпадения состояния КТ с эталонами управление передается встроенной в монитор программе анализа полученных данных. В основе ее работы лежит принцип однозначной связи ошибочного состояния сигнала с источником, вызывающим эту неисправность (он верен для большинства из диагностических сигналов).

Введение дополнительных аппаратных средств диагностики в систему диагностирования позволило существенно увеличить глубину поиска дефекта БПП и использовать эту систему не только в процессе ремонта БПП в условиях эксплуатации, но также и в процессе настройки БПП в целом и его отдельных плат в условиях производства. Таким образом, эта система обеспечивает более высокую технологичность ремонта и производства БПП.

ЛИТЕРАТУРА

1. Дыбой В. А., Каштанов В. В., Лазарев В. О., Фокин А. А. Быстродействующий периферийный процессор «Электроника МО 1603» // Автометрия.— 1985.— № 2.
 2. Дыбой В. А., Семенова О. С., Фокин А. А. Диагностика периферийных процессоров и контроллеров // Автометрия.— 1985.
- Статья поступила 21 мая 1986 г.

УДК 681.3.06 : 326

В. М. Каширин, М. С. Понтак, Д. И. Рубинштейн, В. С. Смолкин

КОМПЛЕКС ДЛЯ ОТЛАДКИ СИСТЕМ НА БАЗЕ ОДНОКРИСТАЛЬНЫХ ЭВМ СЕРИИ КМ1816

Комплекс позволяет автоматизировать процесс разработки программного обеспечения и отладки МП-систем и реализовать следующие основные процессы:

создание и редактирование прикладных программ, написанных на языке ассемблера однокристальной ЭВМ (ОЭВМ) КМ1816ВЕ48;

трансляцию исходных программ и получение загрузочных модулей;

отладку программ в кроссрежиме с помощью программно-логической модели ОЭВМ;

отладку программ на реальном объекте (проектируемой МП-системе) с помощью отладочного устройства;

сохранение отлаженной программы в дисковом файле и занесение ее в ППЗУ ОЭВМ или внешнее ПЗУ.

Аппаратные средства комплекса имеют структуру, представленную на рис. 1. УВК СМ-1800 содержит базовую ЭВМ, ОЗУ емкостью 64К байт, таймер, видеотерминал, НГМД, устройство печати, два модуля ИРПР для связи с отладочными устройствами, модуль ИРПС для моделирования канала связи разрабатываемой МП-системы с другими устройствами.

Отладочные модули «Электроника ОУ-48» [1] выполняют функции внутрисхемных эмуляторов. Программа встроенного монитора отладочного устройства обеспечивает работу под управлением СМ-1800. С помощью комплекса возможно отлаживать двухпроцессорные системы (благодаря наличию двух отладочных устройств).

Программатор ППЗУ предназначен для занесения рабочих программ в микросхемы памяти К573РФ2, К573РФ5, К573РФ6, К556РТ4, К556РТ5, К556РТ7 и в ОЭВМ КМ1816ВЕ48.

Программное обеспечение комплекса включает операционную систему ДОС 1800 [2]. Кроссмакроссамблер создает абсолютный объектный файл в шестнадцатеричном формате. Объединение программ выполняется на уровне исходного текста. В целом возможности языка и транслятора соответствуют описанию, приведенному в работе [3]. Центральный компонент программного обеспечения — программа управления моде-

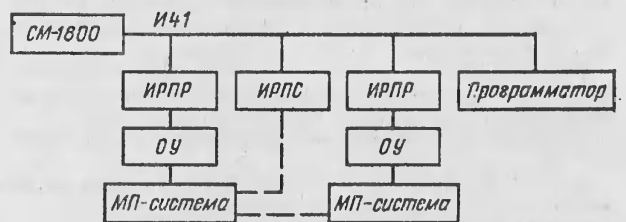


Рис. 1. Структура аппаратных средств комплекса

1. ЧТЕНИЕ С ДИСКА: ИМЯ ФАЙЛА: MODK.HEX
2. РЕЖИМ РАБОТЫ с МОДЕЛЬ/ЭМУЛЯТОР (М/Е) N
3. РЕЖИМ ОТЛАДКИ: ВЫГОВОР/НЕПРЕРЫВНЫЙ (S/C) S
4. РЕЖИМ ТРАССИРОВКИ: (1/2) 1
КОНТРОЛЬ ОЗУ: ДА/НЕТ (Y/N) N
5. КОНФИГУРАЦИЯ ИИ-СИСТЕМЫ:
ОБЪЕМ ПЗУ: 03FF
ВНЕШНЕЕ ОЗУ: ЕСТЬ/НЕТ(Y/N) N
БИС НА ШИНЕ "BUS": ЕСТЬ/НЕТ(Y/N) N
ПОРТЫ P4-P7: ЕСТЬ/НЕТ(Y/N) N
6. ВЫВОД ПРОГРАММЫ НА ДИСК:
7. ВЫВОД ЛИСТИНГА: ПЕЧАТЬ/ДИСК(P/D) P
8. ИНИЦИАЛИЗАЦИЯ:
9. КОМАНДНЫЙ РЕЖИМ: 0
10. КОНЕЧНЫЙ АДРЕС: 03FF
11. ТЕКУЩИЙ АДРЕС: 0000

```

1000H 0006 2300 NGU A.00B
1000FF R4=FF A=0B PSJ=00
101002 R5=02 F0=0 F1=0 TF=0
1020EC R6=EC 00=FF 01=02 02=EC
103000 R7=00 00J=00 004=FD 005=00
! КОНТРОЛЬНАЯ ТОЧКА № A(3)=1
    
```

Рис. 2. Бланк исходных данных

лированием и отладкой, включающая программно-логическую модель и интерпретатор команд ОЭВМ.

Интерфейс пользователя с системой организован в форме экранных бланков. Позволяя одновременно представить на экране большой объем необходимой информации, такая форма значительно облегчает работу с системой, особенно на этапе ее освоения. Исходный бланк (рис. 2) служит для задания основных режимов работы, конфигурации системы и функций файлового ввода-вывода. В некоторых бланках используется управление с помощью команд. Список возможных команд может присутствовать на экране, облегчая выбор. Бланки имеют фиксированный и переменный состав, в зависимости от текущего состояния управляющей программы.

В процессе работы производится диагностика конфликтных ситуаций, а также неправильных действий оператора при вводе (синтаксические ошибки, нарушение диапазона вводимых чисел и т. д.). При этом в соответствующем месте экрана появляется сопровождаемое звуковым сигналом сообщение об ошибке, которое сохраняется на экране в течение трех секунд, после чего восстанавливается прежнее состояние экрана. Перейти на другой бланк можно в любое время с помощью специальных кодов, вводимых с клавиатуры.

Работу с бланком в любой момент можно прервать и перейти в режим экранного ввода. При этом текущая информация сохраняется на экране и в то же время оператору предоставляется возможность проводить любые экранные манипуляции, доступные в автономном режиме работы видеотерминала, например ввод комментариев. Содержимое экрана может быть выведено на печать.

В режиме моделирования предусмотрены следующие возможности:

- задание конфигурации отлаживаемой МП-системы;
- отладка программ в режимах непрерывного выполнения и пошаговом;
- контроль выполнения отлаживаемой программы (трассировка), включая дизассемблирование выполняемых команд;
- контроль выполнения программы с одновременным представлением на экране основных ресурсов ОЭВМ;
- контроль и модификация всех программно-доступных ресурсов ОЭВМ (мониторные функции);
- непосредственное выполнение любой заданной оператором команды ОЭВМ без изменения текущего счетчика команд;
- оперативное изменение памяти программ объектной ОЭВМ;
- задание контрольных точек по разнообразным условиям;
- имитация любых внешних сигналов ОЭВМ;
- вставка программных блоков в исполняемую программу без внесения в нее изменений;

загрузка объектной программы с диска и вывод отлаженной программы на диск или печать;

вывод листинга отлаженной программы на диск или печать;

контроль времени выполнения программы.

Конфигурация моделируемой МП-системы определяет характер данных, представленных в бланке имитации внешних признаков (рис. 3), а также действия интерпретатора при обработке команд ввода-вывода. Кроме встроенных портов ОЭВМ в моделируемую систему могут быть включены внешние порты P4...P7, обращенные к которым обеспечивается с помощью команд MOV, ANLD и ORLD.

Пошаговый режим отладки не требует пояснений. В непрерывном режиме перед переходом к следующей команде может быть задана пауза, в течение которой оператор имеет возможность проследить правильность выполнения по выведенным результатам («следу») последней выполненной команды. При нулевой паузе трассировка отсутствует и выполнение программы останавливается в контрольных точках либо при достижении заданного конечного адреса, либо по команде оператора в произвольный момент времени.

Результаты отладки с помощью монитора отображаются на экране также в форме определенного бланка (рис. 4).

Ниже приведен перечень команд монитора (без параметров): D — вывод на экран внутреннего ОЗУ; XD — вывод на экран внешнего ОЗУ; S — контроль и модификация во внутреннем ОЗУ; XS — контроль и модификация во внешнем ОЗУ; M — перемещение во внутреннем ОЗУ; XM — перемещение во внешнем ОЗУ; FI — заполнение внутреннего ОЗУ константой; XF1 — заполнение внешнего ОЗУ константой; A — контроль и модификация содержимого аккумулятора; PSW — конт-

====ЗАДАНИЕ ВНЕШНИХ СИГНАЛОВ==== ! БЛАНК 2 ! NO. = 0

<p>1. ВХОД T0: СИГНАЛ: 1</p> <p>3. ПОРТ P1: РЕЖИМ ВВОДА: ФИКС./ОПРАВ. (F/Q) F</p> <p>РЕГИСТР ВВОДА: ----- РЕГИСТР ВЫВОДА: 11100101 РЕЗУЛЬТ. ВВОДА: 11100101</p> <p>5. ШИНА "BUS": ОЗУ/БИС(R/L) L РЕЖИМ ВВОДА: ФИКС./ОПРАВ. (F/Q) Q</p> <p>РЕГИСТР ВВОДА: 01011010</p> <p>7. КОМАНДНЫЙ РЕЖИМ:</p>	<p>2. ВХОД T1: ПЕРЕМ. ЗАПРЕД. (T/СЧ. ВЫКЛ.) СИГНАЛ: 1 T=00 TF=0</p> <p>4. ПОРТ P2: РЕЖИМ ВВОДА: ФИКС./ОПРАВ. (F/Q) F</p> <p>РЕГИСТР ВВОДА: 00000000 РЕГИСТР ВЫВОДА: 11111111 РЕЗУЛЬТ. ВВОДА: 00000000</p> <p>6. ВХОД ПРЕРЫВАНИЯ: ЗАПРЕД. СИГНАЛ: 1</p> <p>8. ПОРТЫ P4-P7:</p> <table border="0" style="width: 100%; text-align: center;"> <tr> <td>0 ПОРТА:</td> <td>P4</td> <td>P5</td> <td>P6</td> <td>P7</td> </tr> <tr> <td>РЕЖИМ: (F/Q/O)</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>РЕГ. ВВОДА:</td> <td>0000</td> <td>0000</td> <td>1111</td> <td>1111</td> </tr> <tr> <td>РЕГ. ВЫВОДА:</td> <td>0000</td> <td>0000</td> <td>1111</td> <td>1111</td> </tr> </table>	0 ПОРТА:	P4	P5	P6	P7	РЕЖИМ: (F/Q/O)	0	0	0	0	РЕГ. ВВОДА:	0000	0000	1111	1111	РЕГ. ВЫВОДА:	0000	0000	1111	1111
0 ПОРТА:	P4	P5	P6	P7																	
РЕЖИМ: (F/Q/O)	0	0	0	0																	
РЕГ. ВВОДА:	0000	0000	1111	1111																	
РЕГ. ВЫВОДА:	0000	0000	1111	1111																	

Рис. 3. Бланк имитации внешних сигналов

====МОНИТОР==== ! БЛАНК 4 !

	ACC	ВНУТРЕННЕЕ ОЗУ
R0 FF FF	01110100=5A	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
R1 02 82		00 FF 02 EC 00 FD 00 EF 00 FF 00 FF 02 FD 00 ED 00
R2 EC FF		01 FD 00 EF 00 FF 02 FF 02 FF 02 FF 00 7F 02 FF EC
R3 00 00	PSW	02 FF FF FF 3B FF 02 FF 02 BF 02 FF 02 FF 0A FF 02
R4 FD 7F		03 77 02 F7 02 FF 0A 77 02 77 02 FF 0A 7F 02 7F 0A
R5 00 82	00001000=03	
R6 EF FF		
R7 00 82		
	ОЛАГЧ	ПОРТ P1
	Г0=0 F1=0 D0F=0 TF=0	11100101=E3
		00000000=00
		01011010=5A
AA=00-5AH		ТАЙМЕР
*F1=1=0		МАСКИ ПРЕРЫВАНИЙ
*S20H 7F-FFH 33-FFH BC-FFH FF-3BH		00000000=00
		ВНЕШНИЕ: I=0
		ДИТРЕЙЗЫ: TCNT1=0
		PC=0017

Рис. 4. Бланк монитора

роль и модификация содержимого регистра состояния; В — контроль и модификация указателя банка памяти программ; I — установка (сброс) маски внешнего прерывания; TC — установка (сброс) маски внутреннего прерывания; TM — контроль и модификация содержимого регистра таймера-счетчика; F0 — установка (сброс) флага F0; F1 — установка (сброс) флага F1; Z — контроль и изменение памяти программ; J — переход в другой бланк; C — непосредственное выполнение заданной команды ОЭВМ; G — выполнение отлаживаемой программы с текущего адреса; GS — выполнение одного шага с текущего адреса в непрерывном режиме работы.

Результаты выполнения команд отображаются в мониторинговом бланке. Режим выполнения команд монитора доступен во всех бланках. При переходе к нему в правом или левом нижнем углу экрана появляется командное поле и символ «*», означающий готовность к вводу очередной команды. В процессе работы в этом режиме осуществляется построчный сдвиг вверх (режим свитка).

Трассировка выполнения программы может осуществляться в двух режимах. В первом режиме (см. рис. 1) на экране сохраняется текущий бланк, а «след» последней выполненной команды выводится в командном поле. Перечень контролируемых ресурсов фиксируется и включает дизассемблированный код команды, счетчик команд, все регистры текущего банка, аккумулятор, регистр состояния (PSW), флаги F0 и F1, а также флаг таймера-счетчика. Дополнительно можно заказать для вывода две независимые области размером в три последовательно расположенные ячейки памяти, расположенные во внутреннем или внешнем ОЗУ.

Во втором режиме (рис. 5) текущий бланк исчезает с экрана, а результаты выполнения каждой команды выводятся в одну строку со сдвигом вверх предыдущего содержимого экрана. Перечень трассируемых ресурсов задается в бланке 1 и ограничивается только длиной экранной строки. В любой момент выполнение может быть остановлено с возможностью продолжения или перехода в любой бланк.

Контрольные точки устанавливаются в отлаживаемой программе с помощью специального бланка. В комплексе реализованы критерии контроля по установленному заданному значению в аккумуляторе, регистре состояния или любой заданной ячейке ОЗУ (в том числе внешней); изменению текущего или заданного значения перечисленных регистров или ячеек ОЗУ; установлению или изменению указанных разрядов в вышеперечисленных ресурсах; заданному коду команды; заданному значению на шине данных; установлению и изменению флагов F0, F1 и флага таймера-счетчика. По достижении контрольной точки выводится предупредительное сообщение, после чего оператор может продол-

жить выполнение программы (сохранив данную контрольную точку или удалив ее) или перейти в командный режим.

Если в процессе отладки выявилась необходимость вставить в определенном месте программы некоторую последовательность команд, это можно сделать, не повторяя полный цикл «редактирование — трансляция» и не нарушая программу. Требуемую последовательность можно разместить в свободных ячейках адресного пространства (4К байта), помстив ее адресом команды, перед которой она должна выполняться. Переход к дополнительной последовательности и возврат в исходную последовательность команд будут осуществляться интерпретатором автоматически. Команды, выполняемые из области вставок, при трассировке выделяются символом «*».

Средства работы со вставками сведены в отдельный бланк. Они обеспечивают автоматический поиск свободных участков памяти указанной длины, ввод, редактирование и исключение вставок. Ввод производится в шестнадцатеричных кодах, однако для удобства контроля команда сразу же дизассемблируется, текущие вставки отображаются в таблице, которая может быть вызвана оператором.

Имитация внешних сигналов обеспечивается с помощью бланка 2, в котором пользователю предоставляется возможность непосредственно задать состояние внешних входов ОЭВМ, а также установить режимы обработки интерпретатором команд ввода (фиксированный режим опроса). В фиксированном режиме каждый раз вводится одно и то же значение, заданное пользователем в бланке 2. В режиме опроса вводимые данные запрашиваются у оператора. Такая работа возможна для портов P0...P2 и внешних портов P4...P7, подключаемых через стандартный расширитель.

Физическая отладка МП-системы производится с помощью отладочного устройства. При пошаговой работе так же, как и при непрерывной работе с ненулевой паузой, пользователю доступны все описанные средства режима моделирования, за исключением бланка 2. При прогоне программы (непрерывный режим с нулевой паузой) имеются дополнительные ограничения: невозможна работа со вставками, и контрольные точки могут быть установлены только по адресу.

Отладка двухпроцессорных систем предполагает асинхронный режим работы двух ОЭВМ. В простейшем случае программы в отладочных устройствах выполняются в режиме прогона, при этом в контрольных точках оператор может пользоваться любыми мониторинговыми функциями в отношении каждого устройства. Более сложным является случай, когда одно из устройств работает в режиме прогона, а другое — либо в шаговом режиме, либо в непрерывном режиме с ненулевой паузой (т. е. с контролем результатов выполнения каждой команды). В дальнейшем предполагается реализовать полностью независимые режимы отладки для каждого отладочного устройства.

В настоящее время комплекс используется при разработке систем автоматического управления лифтами на базе ОЭВМ КМ1816ВЕ48.

Телефон для справок: 246-67-66, Москва
ЛИТЕРАТУРА

1. Автономные обучающие устройства для отладки микроконтроллеров./Н. И. Гутовец, В. И. Жданов, В. Н. Колодяжный и др.//Электронная промышленность. — 1986. — № 9.
2. МикроЭВМ СМ 1800. Архитектура, программирование, применение./А. В. Гиглавы, Н. Д. Кабанов, Н. Л. Прохоров, А. Н. Шакарда. — М.: Финансы и статистика, 1984.
3. Кобылинский А. В., Сабадаш Н. Г., Тесленко А. К. Система автоматизации программирования однокристалльной микроЭВМ//Микропроцессорные средства и системы. — 1986. — № 3. — С. 23—25.

Статья поступила 11 декабря 1986 г.

АДРЕС	КОД	ИМЕННИК	A	PSW	R0	R1	F0	F1	P0	R5	10B	109	80A	80B
0005	90	MOVX	0R0.A	D3 0B	FF	02	0	0	5A	80	FF	00	7F	6F
0006	230B	MOV	A.10B	0B 0B	FF	02	0	0	5A	80	FF	00	7F	6F
030B	90	MOVX	0R0.A	0B 0B	FF	02	0	0	5A	80	FF	00	7F	6F
0009	2331	MOV	A.131	31 0B	FF	02	0	0	5A	80	FF	00	7F	6F
000B	90	MOVX	0R0.A	31 0B	FF	02	0	0	5A	80	FF	00	7F	6F
000C	2350	MOV	A.150	50 0B	FF	02	0	0	5A	80	FF	00	7F	6F
000E	90	MOVX	0R0.A	50 0B	FF	02	0	0	5A	80	FF	00	7F	6F
000F	2390	MOV	A.190	90 0B	FF	02	0	0	5A	80	FF	00	7F	6F
0011	90	MOVX	0R0.A	90 0B	FF	02	0	0	5A	80	FF	00	7F	6F
0012	80	MOVX	A.0R0	5A 0B	FF	02	0	0	5A	80	FF	00	7F	6F
0013	530F	ANL	A.10F	0A 0B	FF	02	0	0	5A	80	FF	00	7F	6F

Рис. 5. Пример трассировки

„ПОКАЯНИЕ“

авторов ПЭВМ «Ириша»

В 1—4-м номерах журнала за 1986 год был опубликован цикл статей, посвященных описанию конструкции ПЭВМ «Ириша». Желание авторов в сжатые сроки подготовить к публикации большой объем материала с тем, чтобы он не потерял своей новизны, имело, увы, и свои отрицательные стороны.

В процессе переработки конструкторской документации в форму журнальной статьи, возникли неточности и ошибки. Авторы постарались сделать так, чтобы они не мешали повторению конструкции. Например, топология печатных плат приведена в виде копии с реальных фототаблонов. Это позволяет воспроизвести ПЭВМ, несмотря на неточность в принципиальной схеме. Безусловно, сказанное выше не украшает статью, но позволяет как-то объяснить возникшую ситуацию.

Во многих письмах читатели отмечают, что авторы при записи информации в ПЗУ K556PT4 использовали «зеркальную» кодировку, что затруднило повторение ПЭВМ. Здесь, как нам кажется, винить только авторов нельзя. Мы пользовались одним из действующих руководств по применению этих микросхем в котором, как

оказалось, есть разночтения с другими справочниками и руководствами. Кстати, если использовать материал по применению микросхем K556PT4, приведенный в первом номере журнала МПСИС за 1987 год, то никаких ошибок не будет.

Некоторые немногочисленные ошибки в принципиальной схеме и «прошивках» ПЗУ, не приводящие к неработоспособности машины, были выявлены на этапе эксплуатации опытной серии ПЭВМ. Это, видимо, естественные ошибки, всегда сопровождающие новые разработки как машин, так и программ, и к ним нужно относиться с пониманием.

Авторы получили много писем, в которых читатели сообщают о замеченных неточностях и ошибках. Мы, пользуясь случаем, хотим, во-первых, поблагодарить авторов писем за столь внимательное отношение к опубликованному материалу и, во-вторых, принести извинения за допущенные огрехи. Заинтересованным читателям мы рассылаем полную информацию по корректировке опубликованного материала.

Адрес для справок: Москва, Ленинские горы, МГУ, химический факультет, лаборатория стабильных изотопов. Телефон для справок: 939-44-33

От имени авторов В. Романов

УДК 681.3.06

В. Н. Барышников, И. Н. Бондарь, М. А. Воронов, В. Ю. Романов

ПРОГРАММЫ СТАРТОВОГО ПЗУ ПЭВМ «ИРИША»

В предыдущих номерах журнала [1, 2] были приведены программы резидентного ПЗУ, отвечающего за вывод текстовой и графической информации. Здесь дано описание оставшейся части резидентных программ, используемых при начальном старте ПЭВМ. Эти программы содержатся во втором ПЗУ модуля процессора. Минимально необходимый объем ПЗУ составляет 2К байта. В таком комплекте обеспечиваются инициализация внутренних устройств ПЭВМ и загрузка программ или ядра операционной системы с магнитной ленты или 8-дюймового гибкого магнитного диска. ПЗУ может быть расширено программой монитора-отладчика, требующего еще 4К байта памяти. ПЗУ минимального объема (2К байта) имеет маркировку BOOT, а максимального (8К байт) — BOOTM. Последнее включает в себя кроме монитора-отладчика набор подпрограмм для работы с дополнительной периферией. В данной статье подробно рассматриваются программы ПЗУ минимального объема. Программа монитора-отладчика имеет самостоятельное значение и может быть заменена любой другой программой пользователя подходящего объема. Она должна иметь стартовый адрес 800Н. ПЗУ BOOT и BOOTM устанавливаются в розетку E5 модуля процессора [3].

При включении питания или после получения сигнала «сброс», модуль процессора ПЭВМ начинает выполнять программу, записанную в ПЗУ

BOOT или BOOTM. В ПЗУ BOOT (объем 2К байта) содержатся программы «холодного» старта, работы с «меню», загрузчики с различных внешних устройств, подпрограммы обслуживания накопителя на бытовом кассетном магнитофоне, подпрограммы работы с клавиатурой и ряд других вспомогательных подпрограмм (рис. 1). ПЗУ BOOTM содержит все

эти программы, а также программы работы с простой матричной клавиатурой, накопителями на 5-дюймовых магнитных гибких дисках, подпрограммы обслуживания игрового адаптера, работы с локальной сетью. Кроме того, оно включает монитор-отладчик, аналогичный по своим функциям утилите DDT, входящей в состав операционной системы CP/M.

Оба варианта ПЗУ включают сходную программу холодного старта с адресом 0000Н, устанавливающую режимы работы БИС процессорного и дисплейного модулей, а также формирующую в ОЗУ ПЭВМ файл управления. По окончании работы она передает управление программе вы-

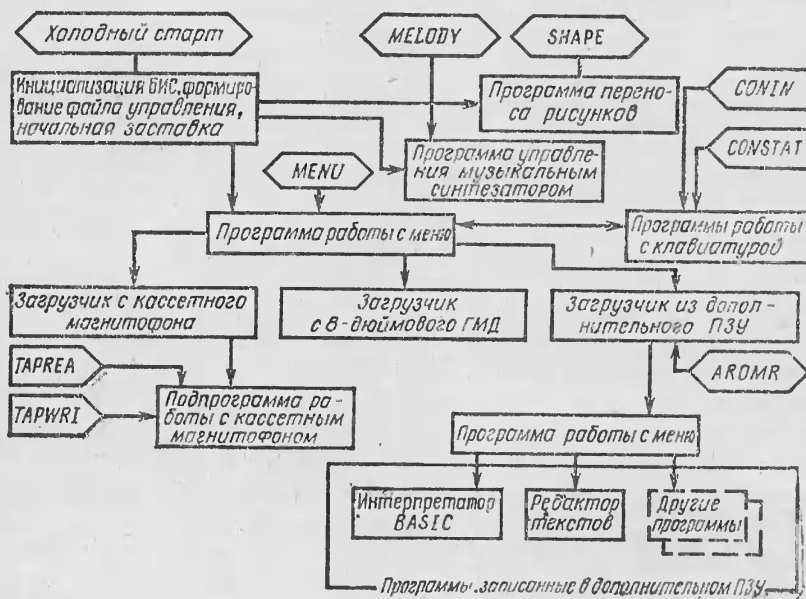


Рис. 1. Структура ПЗУ BOOT

Входы ПЗУ BOOT и BOOTM

Адрес	Способ обращения	Выполняемая функция
0000H	JMP	CSTART — подпрограмма «холодного старта». Инициализирует БИС ПЭВМ, файл управления, формирует стартовую заставку. Далее передает управление программе выбора начальной загрузки
0040H	CALL	TAPREA — подпрограмма чтения блока данных с магнитофона. В HL — адрес начала буфера, BC — число считываемых байт. Возвращает флаг CY=0, если считывание без ошибок или CY=1, если обнаружена ошибка или в процессе считывания была нажата клавиша
0043H	CALL	TAPWRI — подпрограмма записи блока данных на магнитофон. В rg. A указывается тип заголовка (0 — короткий, 1 — длинный) в HL — адрес начала буфера, в BC — длина
0046H	CALL	SHAPE — подпрограмма переноса изображения рисунка на экран из файла путем «исключения» с видеоОЗУ. Адрес начала файла в ячейке FFFEH
0049H*	CALL	GAME — подпрограмма обслуживания игровых пультов, (только в ПЗУ BOOTM)
004CH*	CALL	RES1 — резервный вход для программы, записываемых в ПЗУ BOOTM
0052H	JMP	MENU — программа работы с меню. Начало файла меню указывается в регистрах HL число записей в файле не более 8. Запуск по нажатию на клавишу BK
0055H	CALL	MELODY — подпрограмма проигрывания мелодии из одноименного файла. Адрес начала файла в ячейке FFFCH. Ноты кодируются кодами 1...40. Коды с 40...127 интерпретируются как пауза. Коды больше 127 задают длительность звучания. Окончание файла по коду 00H
004FH	JMP	APOMR — программа считывания программ из ПЗУ дополнительной платы и запуска на исполнение
0058H	CALL	CONSTAT — подпрограмма проверки готовности клавиатуры. Возвращает флаг CY=1, если клавиша нажата
005BH	CALL	CONIN — подпрограмма приема символа с клавиатуры. Семьбитный код символа в регистрах A и C

* — Подпрограммы содержатся только в ПЗУ BOOTM.

бора варианта начальной загрузки. Последняя проверяет состав комплекта машины, наличие факультативных аппаратных средств и в зависимости от конфигурации предлагает пользователю меню выбора начальной загрузки. К числу факультативных средств относится плата дополнительного ПЗУ, контроллер локальной сети, накопитель на 5-дюймовых гибких магнитных дисках. Кроме того, эта программа выясняет тип стартового ПЗУ и наличие программы монитора.

Проверка установки платы дополнительного ПЗУ, увеличивающей общий объем резидентных программ на 32-64К байт, производится выполнением 65536 циклов считывания из адреса 14H. Если при этом обнаруживается хотя бы один байт, отличный от FFH, то считается, что такая плата есть. Наличие сетевого контроллера определяется по состоянию сигнала EXT2: если он имеет низкий уровень, то сетевой контроллер входит в состав ПЭВМ. Признаком НГМД служит сигнал INT1 с внешней магистрали после свитча сигнала «сброс». Существенно, что программы обслуживания сетевого контроллера и НГМД исполняются только ПЗУ BOOTM. Программа, находящаяся в ПЗУ BOOT, фиксирует наличие последних двух устройств, но при попытке работы с ними выдает сообщение об ошибке.

В минимальном комплекте ПЭВМ обеспечивает загрузку с диска, имеющего встроенный контроллер (например, ГМД 7012, СМ5631) с накопителя на базе бытового кассетного магнитофона или старт с адреса 4000H, что необходимо для восстановления работоспособности программ, выполнение которых было прервано командой RESET. При использовании ПЗУ BOOTM старт с адреса 4000H заменяется вызовом монитора.

Кроме того, ПЗУ BOOT и BOOTM содержат ряд вспомогательных подпрограмм, используемых при выдаче стартовой заставки и в программах загрузчиков. Некоторые из них имеют самостоятельное значение и могут вызываться из программ пользователя. К их числу относятся программы работы с музыкальным синтезатором, переноса изображений, работы с меню, переноса файлов, записанных в дополнительном ПЗУ, работы с клавиатурой. Обращение к ним оформлено в виде таблицы переходов в начале обоих типов ПЗУ (табл. 1).

Для расширения адресного пространства процессора ПЭВМ «Ириша» свыше 64К байт используется динамическое переключение карт распределения памяти. На рис. 2 приведены возможные карты распределения памяти для типовой «прошивки» ПЗУ селектора адресов MC D22 в ро-

зетке A10 [3], которая дана на рис. 3. Выбор необходимой карты осуществляется записью битов pC2 и pC3 в порт С БИС ППА модуля процессора [3]. Переключение памяти производится сегментами по 16К байт каждый. Карта 1 предоставляет процессору доступ к ОЗУ модуля графического адаптера, включая видео-ОЗУ. Карта 0 всегда устанавливается при включении ПЭВМ и разрешает работу программ резидентного ПЗУ, расположенного в адресах с 0000H до 3FFFH. Остальные карты распределения могут использоваться для подключения дополнительной памяти.

Для работы с клавиатурой ПЗУ содержит две подпрограммы:

CONSTAT и CONIN. Первая фиксирует нажатие клавиши, вторая осуществляет собственно прием кода символа. В первом случае программе пользователя возвращается флаг CY=1 при условии, что клавиша нажата, а во втором — семьбитный код символа в регистрах A и C. ПЗУ BOOTM позволяет, кроме того, работать с простой матричной клавиатурой, имеющей упрощенную схему образования. Тип подключенной клавиатуры определяется автоматически.

Программа управления музыкальным синтезатором MELODY предназначена для генерации различных мелодий или отдельных звуков. Пользователь имеет возможность применить ее в своих программах, обра-

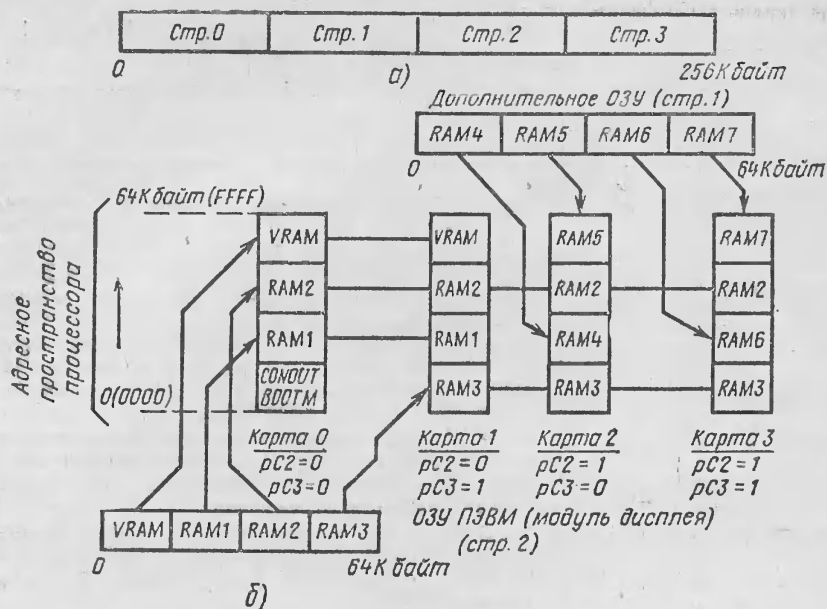


Рис. 2. Схема использования адресного пространства ПЭВМ «Ириша»: а) расположение страниц памяти; б) структура карт распределения памяти

Адрес	X=	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0X	70	FA	F9	F8	80	FA	F9	F8	FB	FA	F9	F8	FB	FA	F9	F8	
1X	FB	FA	F4	F5	FB	FA	F4	F5	FB	FA	F6	F7	FB	FA	F6	F7	

Рис. 3. Прошивка ПЗУ управления памятью

щаяся к ней как к подпрограмме с помощью команды CALL 0055H (см. табл. 1). Перед вызовом под-

программы необходимо сформировать в доступной области ОЗУ нулевой карты распределения памяти

Таблица 2

Кодировка нот

Код		Нота	Код		Нота
десят.	шестнадц.		десят.	шестнадц.	
00	00	Конец мелодии	20	14	ля #
01	01	ми	21	15	си
02	02	фа Б о	22	16	до
03	03	фа # о к	23	17	до #
04	04	соль л т	24	18	ре
05	05	соль # ь а	25	19	ре # В о
06	06	ля ш в	26	1A	ми т к
07	07	ля # а а	27	1B	фа о т
08	08	си я	28	1C	фа # р а
09	09	до	29	1D	соль а в
10	0A	до #	30	1E	соль # я а
11	0B	ре П о	31	1F	ля
12	0C	ре # с к	32	20	ля #
13	0D	ми р т	33	21	си
14	0E	ми # а	34	22	до
15	0F	фа в в	35	23	до # Т о
16	10	фа # а а	36	24	ре р к
17	11	соль я	37	25	ре # е т
18	12	соль #	38	26	ми т а
19	13	ля	39	27	фа ь в
			40	28	фа # я а

файл мелодии и занести адрес его начала в ячейку FFFCH (FFFCH — младший байт, FFFDH — старший байт адреса). При начальном старте в эти ячейки записится адрес файла мелодии, записанной в ПЗУ, поэтому выполнение операции CALL 0055H без записи адреса файла пользователя вызывает проигрывание стартовой мелодии.

Файл мелодии представляет собой набор байт, кодирующих ноты и длительность их звучания. Признаком конца файла мелодии служит код 00H. Байты с нулевым битом в старшем разряде кодируют ноты (табл. 2), а байты с единицей в этом разряде — длительность (табл. 3). Однажды установленная длительность действует до получения следующей команды ее изменения. Ноты пронумерованы от 1 до 40 в порядке увеличения высоты звука. Коды больше 40 интерпретируются как паузы

Таблица 3

Кодировка длительности звучания

Код (шестнадц.)	Длительность, с	Доля
8F	0.123	1,16
9F	0.246	1/8
BF	0.491	1/4
FF	0.983	1/2

с установленной на данный момент длительностью.

Программу музыкального синтезатора использует также программа CONOUT [2] при обработке кода 07H (BELL) и последовательности кодов ESC,M. В первом случае выработывается двухтональный звуковой сигнал, а во втором происходит проигрывание мелодии, записанной в ПЗУ.

Подпрограмма переноса изображений SHAPE формирует в заданном месте экрана рисунок определенных размеров с помощью побитной операции «исключающее ИЛИ» между текущим содержимым видеоОЗУ и файлом рисунка. Результат помещается в экранную область памяти. Таким образом, первое обращение к подпрограмме накладывает рисунок на существующее изображение, а повторное обращение с неизменными параметрами восстанавливает исходную картинку. Вызывается подпрограмма командой CALL 0046H при установленной нулевой карте распределения памяти. Файл рисунка располагается в любом месте ОЗУ, доступном при нулевой карте распределения памяти. Файл начинается с дескриптора, в первых двух байтах которого указывается размер рисунка

ка на экране в строках по вертикали и в байтах по горизонтали. Следующие два байта содержат адрес начала изображения в байтах от начала строки видеоОЗУ. Положение верхнего левого угла рисунка определяется координатами: $Y = \text{целая часть} (\lfloor \text{АДРЕС} \rfloor / L)$; $X = \lfloor \text{АДРЕС} \rfloor - Y * L$.

Здесь АДРЕС — запись адреса в файле; Y — начало первой строки изображения в строках от верха экрана; X — начало левой границы рисунка в байтах от начала строки на экране; L — число байт в строке изображения на экране (40 или 80).

Обратный пересчет выполняется по формуле $\text{АДРЕС} = Y * L + X$.

Задавая параметры в дескрипторе файла рисунка, следует иметь в виду, что подпрограмма не осуществляет контроля за допустимостью их значений, поэтому ошибки могут привести к непредсказуемым последствиям. Общая структура файла рисунка показана на рис. 4. Адрес начала файла указывается в ячейке FFFEH. При начальном старте в эту ячейку заносится адрес файла рисунка заставки ПЭВМ, записанного в стартовом ПЗУ. Программа CONOUT также использует описываемую подпрограмму при обработке последовательности символов ESC, I. Пользователь имеет возможность, таким обра-

на экране перемещается стрелка, указывающая на название выбранной программы. Нажатие на клавишу ВК передает управление этой программе.

Пользователь имеет возможность задействовать программу работы с меню в своих программах в качестве переключателя хода их исполнения. Для обращения к ней необходимо установить нулевую карту распределения памяти, в которой становятся доступными программы резидентного ПЗУ и выполнить операцию JMP 0052H. Перед вызовом программы работы с меню в области ОЗУ, доступной в нулевом распределении памяти, необходимо сформировать файл управления меню и записать адрес его вершины в регистры HL процессора.



Рис. 5. Структура файла меню

Файл управления меню имеет организацию, аналогичную стеку, и расчет по мере увеличения числа записей в нем в сторону меньших адресов (рис. 5). Каждая запись в файле состоит из четырех байт. Два первых байта указывают на адрес начала символической строки, являющейся названием программы, а два следующих — на стартовый адрес, которому нужно передать управление при запуске выбранной программы. Адреса запуска должны быть больше 4000H, иначе управление будет передано на какой-то адрес резидентного ПЗУ, что может привести к возникновению ошибочных ситуаций. В вершине файла управления находятся три служебных байта. Первые два байта являются адресом начала строки символов, составляющих заголовок меню, а третий указывает на число записей в файле. Если в этой ячейке содержится число большее 8, то программа работы с меню автоматически устанавливает его значение равным 8, ограничивая тем самым число записей в файле.

Названия программ и заголовок меню должны укладываться в одну строку экрана (40/80 символов) и не использовать команд, которые могут изменять текущее положение курсора, а также команд ПС и ВК.

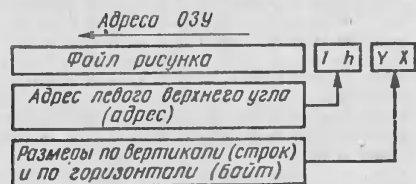


Рис. 4. Структура файла рисунка

зом, обращаться к подпрограмме переноса изображений через общий вход вывода текстовой информации.

Работая с цветным изображением, необходимо учитывать, что в этом режиме за засвечивание одной точки экрана отвечают два соседних бита, комбинация которых несет информацию о цвете. Поэтому наложение рисунка из файла на существующее на экране изображение может привести к появлению дополнительных цветов. Более подробно с кодировкой цветов и способом формирования цветного изображения в ПЭВМ «Ириша» можно ознакомиться в описании графического адаптера [4].

Подпрограмма работы с меню MENU позволяет запустить прикладную программу из заданного списка. Число программ в списке может изменяться от одной до восьми. Выбор программы производится с помощью кольцевого циклического перебора записей в списке по нажатию на любую из клавиш клавиатуры, исключая клавишу ВК. В процессе выбора

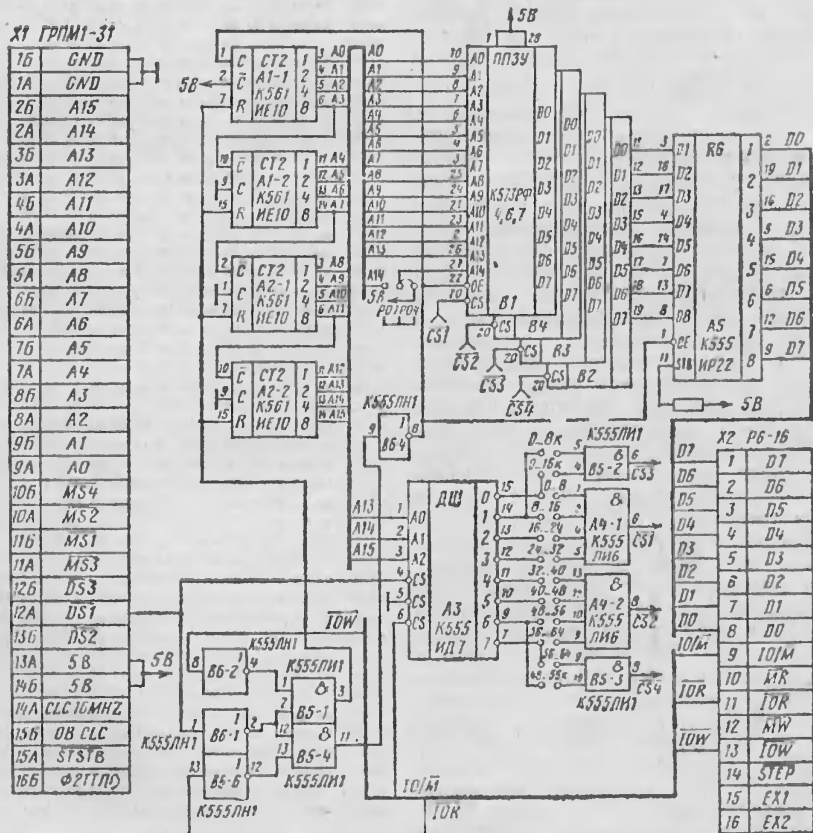


Рис. 6. Принципиальная схема платы дополнительного ПЗУ

Каждая текстовая строка должна заканчиваться кодом 00Н. В начало текстового сообщения заголовка меню необходимо включить команды очистки экрана (ОСН), две команды ПС (ОАН) и команду запрета формирования изображения курсора ESC. (61Н). Рабочий алфавит меню должен быть установлен в текстовом сообщении заголовка. Перед переходом на выбранную прикладную программу, программа работы с меню устанавливает в качестве рабочего латинский алфавит и разрешает формирование изображения курсора.

Программы, предназначенные для записи в ПЗУ дополнительной пла-

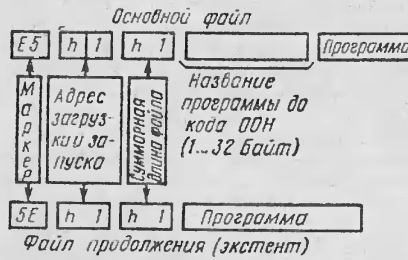


Рис. 7. Структура файлов программ для дополнительного ПЗУ

Дополнительное ПЗУ располагает-ся на плите расширения (рис. 6), подключаемой к технологическим разъемам модуля процессора, и служит для хранения часто используемых программ, таких как, например, интерпретатор языка Бейсик, отладчик и т. п. Она позволяет расширить объем программ, хранимых в ПЗУ, до 64К байт. Данные из этих ПЗУ считываются побайтно через резервированный адрес 14Н в области адресов ввода-вывода модуля процессора. Плата расширения имеет свой внутренний счетчик, вырабатывающий адреса для считывания данных из ПЗУ. Последовательное выполнение операции чтения из адреса 14Н позволяет перенести данные из ПЗУ в память ПЭВМ. Выполнение операции записи по адресу 14Н приводит к обнулению внутреннего счетчика адресов платы.

Выборку данных и программ с платы расширения производит специальная программа АРОМР, хранящаяся в стартовом ПЗУ. Эта программа инициализируется после проверки наличия дополнительной платы. В меню начального старта включается ссылка на программу работы с дополнительной платой, которая может быть запущена как один из вариантов начального загрузчика. Она просматривает содержимое ПЗУ дополнительной платы, проверяет соответствие форматов записи файлов и подготавливает по их заголовкам файл для последующего запуска программы меню файлов ПЗУ. Если формат записей отличается от требуемого, то выдается сообщение об ошибке и по нажатию на любую из клавиш происходит возврат в стартовое меню. Выбранный программой файл из ПЗУ переносится в ОЗУ и запускается второй частью программы. Программа перезагрузки и запуска во время работы располагается в невидимой по ходу начального старта части видеоОЗУ, начиная с адреса DF40Н. Размещение этой части программы вне зоны рабочих адресов ОЗУ позволяет выполнять операцию пересылки в любую часть памяти, начиная с адреса 0000Н до С000Н, после необходимого переключения карт распределения памяти.

ты, должны быть оформлены по определенным правилам (рис. 7). Файл должен начинаться с кода 5Н, за которым следуют 4 байта дескриптора и название файла для программы меню. Первые два дескриптора содержат адрес, с которого должна начинаться программа в ОЗУ. По этому же адресу передается управление после перезагрузки программы. Следующие два байта определяют размер информационной части файла. Программа перезагрузки не контролирует выход за допустимые границы адресов ОЗУ, поэтому ошибки в задании начального адреса и размера

```

0000:  F3 31 00 00 03 47 72 00 C3 F5 9C 00 00 00 00 00
0010:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0020:  00 00 09 00 00 30 00 00 00 00 00 00 00 00 00
0030:  00 00 00 00 30 30 00 00 00 00 00 00 00 00 00
0040:  C3 35 00 C3 5A 28 C3 4B 01 C3 00 00 C3 00 00 C3
0050:  00 00 C3 24 03 C3 0B 01 C3 67 00 C0 67 00 D2 5B
0060:  00 0B 04 86 7F 4F C9 3E 0E 03 13 DB 11 47 C9 7E
0070:  12 23 13 0B 78 81 C2 64 0C 3F 00 C2 85 00 01
0080:  47 31 C3 88 01 01 04 01 C3 0B 01 F6 F9 C0 C3 4B
0090:  01 30 00 00 04 CE 15 FF FF 00 00 50 41 00 00 00
00A0:  00 39 13 00 00 FF 20 21 09 00 72 F6 FF C0 4B 01
00B0:  C0 85 04 01 37 01 C0 D6 01 11 84 00 C3 0C 04 1B
00C0:  47 43 31 50 32 34 38 2C 33 34 53 30 2C 30 46 30
00D0:  2C 30 52 33 32 49 54 54 00 06 0F 2C 04 C1 83 F9
00E0:  F6 FC 41 C1 82 08 04 82 63 C1 82 0B 7C FC 55 C1
00F0:  82 08 04 82 49 C1 82 09 F8 FC 41 C1 80 0C 00 00
0100:  C0 83 00 00 00 00 00 C7 9F 06 C3 0C 38 C0 98 6E
0110:  73 6C 65 09 98 6B F2 64 44 F1 9B 67 03 6C FC E1
0120:  98 67 33 6C 06 C1 9F 06 33 FC C0 00 18 00 00 00
0130:  00 FF 99 FF FF FF FF A0 15 19 1C 1E 1F 1E 1C 19
0140:  A0 15 1B 1C 1E 1F 1E A0 1C 19 00 2A FF FF 4K 23
0150:  46 23 1E 23 56 23 K5 3A 45 FF FF 10 21 00 C0 CA
0160:  63 01 21 00 02 19 01 03 5A 66 44 F5 1A AX Y7
0170:  13 23 0B C2 6D 01 F1 4F 06 00 09 C1 05 C5 F5 C7
0180:  6E 01 F1 C1 C9 3E 06 0B 0E 3E 0A 03 13 AF D3 1A
0190:  3E 00 83 13 3E 0C 03 13 C9 C5 79 D3 0A 78 D3 0A
01A0:  3E 20 C0 0E 01 K5 2A 8E FF 2B 7B 04 C2 A9 01 01
01B0:  C0 8D 01 C1 C9 5E F5 C5 21 F3 01 F6 28 D2 01 01
01C0:  00 00 4F 09 09 7E 47 23 7E 4F CD 99 01 C1 F1 K1
01D0:  C9 3E 28 C0 01 2A FC FF 44 04 0A 17 DA KH 01
01E0:  1F K5 00 C8 CD 05 01 03 C3 0B 01 1F E6 7F 03 32
01F0:  83 FF C3 0B 01 2A 23 27 C5 2B 23 0E 21 71 1F
0200:  21 71 1F 01 1D C3 1C 1F 1A 8B 19 0E 17 A6 16 62
0210:  15 11 43 12 C5 11 07 10 89 0F 0E 0E K6 0E 10
0220:  00 16 05 C7 0B D3 0B 29 0A 89 09 71 09 62 08 0C
0230:  08 5C 07 E4 07 73 07 06 0E 43 0C 43 05 K9 05 94
0240:  05 44 04 F9 04 B1 04 6K 00 00 3K 82 D3 13 21 9A
0250:  00 01 10 00 11 74 FF C0 72 00 2E 76 D3 0B 3E 21
0260:  03 09 3E 1B D3 09 3K 3G D3 0C 3E 8B D3 0D 3K YF
0270:  03 0B 3E 20 D3 0C 0K 0E 21 94 00 7B 03 07 0D 23
0280:  C2 7B 02 D8 0E 3E 3E D3 0B 0C 0C D3 08 3E 00 D3
0290:  08 21 3A 01 22 FC FF CD 0A 20 CD AA 00 3E C3 32
02A0:  68 FF 32 64 FF 21 7D 00 22 89 FF 21 8E 00 22 65
02B0:  FF 21 00 C2 6E FF 21 00 00 39 31 FD 0C 06 03
02C0:  3A 00 08 FE F3 CA 85 03 11 A6 03 D5 11 00 08 D5
02D0:  11 B7 03 D5 11 0D 04 D5 11 C7 03 D5 3E 0E 03 13
02E0:  DB 11 17 1D D2 0F 01 11 3B 04 D5 05 15 D3 14 21 00
02F0:  00 22 00 CF CD 01 04 E1 CA 03 04 11 C2 03 D5
0300:  13 00 05 00 00 00 1F DA 13 03 11 C2 03 D5 04 11
0310:  2D 04 05 78 32 FD CE 11 95 03 EB 22 FX 0C KB Y9
0320:  21 FF 0E 5E 2B 5K 2B 85 CD 0B 04 3E 0E 1K D2 32
0330:  03 77 4E 4B 2B 5E 2B BK 2B 2B CD 03 04 0D C2
0340:  35 03 0E 11 86 03 CD 0B 04 AF 11 F3 03 CD 0E 04
0350:  K5 C5 CD 5K 00 FE 0D CA CA 0A 03 F1 3C D8 D2 49
0360:  03 11 F8 03 CD 0B 04 C5 0D 03 F1 81 87 47 2B 2B
0370:  2B CA 7A 03 2D 2B 3D C3 6F C3 56 2E 5K 1K 11 00
0380:  04 CD 0B 04 89 11 AE 03 D5 11 00 04 C3 CF 0E 11
0390:  22 0B C3 KA 02 0B 0A 1B 40 1B 81 0E 20 20 73
03A0:  74 61 72 74 3F 00 8D 6F 6K 69 74 6F 72 00 81 64
03B0:  72 2E 34 30 30 30 60 61 67 6K 69 74 6F 68 6F
03C0:  6K 00 64 6F 70 2K 70 7A 75 00 73 65 74 78 00 84
03D0:  69 73 6B 00 0A 00 20 20 20 20 20 20 20 20 20
03E0:  21 0A 00 20 20 00 1B 5B 35 2A 70 75 73 6B 30 77
03F0:  6B 0F 00 0D 2B 0B 0A 0A 0D 20 0A 0A 0D 01 1A 00

```

```

0400:  1B 62 00 D5 11 D4 03 CD 0B 04 D1 F5 C5 K5 1A FK
0410:  00 CA 1K 04 4F D5 CD 00 20 D1 13 C3 0K 04 K1 C4
0420:  F1 C9 11 2B 04 C3 82 04 44 4B 32 3F 00 21 33 04
0430:  C3 82 04 4K 45 54 3F 00 3E A0 D3 50 CD 82 04 DA
0440:  3B 04 8E 23 D3 50 21 00 A0 0E 80 D8 17 D2 4B
0450:  04 DB 51 77 23 0D C2 4B 04 CD 82 04 DA 3B 04 C3
0460:  00 A0 D8 50 17 17 17 17 62 04 1F C3 11 3B 04
0470:  CD 0B 04 21 00 A0 01 00 01 C2 CD 84 0D 00 A0 11
0480:  B1 04 F5 11 AB 04 CD 0B 04 D1 CD 0B 04 CD 5K 00
0490:  0E 0C CD 00 20 C1 B1 02 0C CF 1B 59 30 2A 0E 4D 4C
04A0:  2D 5A 41 47 52 55 5A 4B 41 0F 00 1B 59 32 2A 07
04B0:  00 0E 4F 5B 49 42 4B 41 20 4D 4C 0F 0B D8 12 FE
04C0:  0B D3 12 7B B1 CA 53 FF DB 14 12 0B 13 C3 4E DF
04D0:  DB 12 8B 73 D3 12 C9 DB 12 F6 0B D3 12 E9 78 81
04E0:  C8 0B DB 14 C3 DE 04 2A 00 CF 09 22 02 C7 CD FE
04F0:  04 2A 0E 0F 23 7C B5 C8 DB 14 FK FF C0 C3 F4 04
0500:  D3 14 11 01 C1 21 FD CE 22 02 CF 21 00 00 22 00
0510:  CF AF 32 FD CE CD F1 04 22 00 CF CA 2K 05 FE K5
0520:  CA 72 05 F1 5E CA AA 05 11 48 05 C3 B2 04 SA FD
0530:  CE B7 C2 E1 05 11 3B 05 C3 82 04 0E 50 5A 5D 50
0540:  50 55 53 54 4F 0F 07 00 0E 67 7B 2E 70 7A 75 0F
0550:  00 21 5D 05 22 FK C2 21 FF CE C3 23 0C 0A 1B
0560:  81 0A 07 0E 20 20 48 41 4A 4C 59 20 00 5A 5B
0570:  0D 00 21 7D CE 34 CD B3 05 21 00 00 39 22 04 CF
0580:  2A 02 CF F9 D5 21 0D 05 21 00 00 39 22 02 0F
0590:  2A 04 CF F9 2A 0D CF 23 22 00 CF DB 14 12 13 FK
05A0:  0D C2 94 05 CD 07 04 C3 18 05 CD 0B 03 0D K7 04
05B0:  C3 18 05 2A 00 CF 23 23 23 22 00 CF DB 14 DB
05C0:  14 DB 14 47 DB 14 4F C9 DB 14 FE FF CA 8C 05 C9
05D0:  04 C5 21 B0 04 11 0D D5 01 21 0E CD 72 00 C3 14
05E0:  CD C8 05 C1 05 CA 03 0B C5 CD 0D 05 CD 1F 06 CD
05F0:  DE 04 CD C8 05 FK E5 CA E3 05 CD 0D 05 CD DK 04
0600:  C3 22 05 CD 07 06 62 6B C1 1F 06 CD 04 DF CD C8
0610:  05 FK E5 CA 5A DF CD 27 06 CD 04 DF C3 0K 06 DB
0620:  14 FK 00 C8 C3 1F 06 DB 14 57 DB 14 5F DB 14
0630:  DB 14 4F C9 3E 0K D3 13 25 C5 CD 02 07 C1 E1 85
0640:  C5 CD 5D 07 C1 E1 DA 51 06 77 0B 23 7B 81 C2 3F
0650:  0E 25 F5 CD 8E 08 F1 E1 C9 K5 CD C8 AK 0E C1 K1
0660:  C5 25 72 CD B9 06 F1 C1 23 0B 7B 81 C2 60 0E 08
0670:  CD 75 06 K1 C9 C5 F5 01 00 00 0B 7B 81 C2 7A 0B
0680:  F1 C1 F5 3K 0D D3 07 3E 0K D3 13 F1 F9 C9 B7 F5
0690:  3E 02 D3 07 21 00 2B 0C 2B 7C 85 CD 37 0E F1 01 A5
06A0:  0F CA A7 06 01 80 3E 3K 0F D3 13 F3 0E KA 06 C9
06B0:  E2 0E 08 7B B1 C2 AC 06 C9 21 2E 2A F5 07 D6 02
06C0:  6F CD FE 06 F1 06 08 0F DC E3 0E D4 06 05 CA
06D0:  C7 0C CD E3 06 CD E3 06 B7 C3 21 2E 26 00 00 CD
06E0:  KF 0C C9 CD EA 0B CD EA 0E C9 21 17 10 00 00 F5
06F0:  2D C2 F0 0E 3K 80 D3 10 25 C2 F8 06 3K 00 D3 10
0700:  F1 C9 F3 3K 0E D3 13 0E 02 D3 07 21 A0 07 51 CD
0710:  07 D8 79 FE DE D2 0E 07 FE 05 DA 0B 07 82 D2
0720:  24 07 2P C3 FE 0E D2 07 07 2B 05 C2 0E 07 21
0730:  00 00 45 55 CD DE 07 D8 09 15 C2 34 07 01 AK 0B
0740:  09 7C 1F E6 7F 57 29 7C 32 57 D8 04 3E K6 FF 7A
0750:  87 82 0E 0F 0F D8 07 D2 6C FF F9 C9 3A FE D8 FF
0760:  57 CD 0F 0B D8 07 07 D2 61 07 CD 0F 07 D8 FF
0770:  07 07 DA 6B 07 1E 00 CD C8 07 41 CD C8 07 D8 FF
0780:  81 DA 7A 07 DA 7A 07 2E 08 CD 0E 07 FE 04 3F
0790:  D8 FE 02 3F 7A 1F 57 79 0F DA CC 07 CD 0E 07 2D
07A0:  C2 8A 07 CD 0F 7A C9 3A BC FF 47 0E 0B D8 07
07B0:  AB F2 BE 07 7B 2F 5F 0C 05 C2 AE 07 79 C9 00 00
07C0:  00 3C 05 C2 AK 07 79 C9 CD 07 07 D8 0E 0C CA
07D0:  DC 07 07 AB F2 07 7B 2F 5F C9 0D C9 CD F9
07E0:  07 D8 07 07 DA FE 07 1E 00 CD C8 07 C3 CE 07
07F0:  DB 11 07 3C 3C 3C 3C 3C 72 74 76 30 33 2A 38 37

```

Рис. 8. Стартовая программа

(строка с адресом 0050 должна содержать коды 00 05 C3 — далее верно).

могут привести к непредсказуемым последствиям. Непосредственно за дескриптором располагается текст названия файла. Правила его оформления в основном сходны с правилами оформления названий для программы работы с меню. Особенностью данного случая является то, что в заголовке меню производится установка работы с русским алфавитом, поэтому для выдачи названий латинскими буквами необходимо перейти в латинский алфавит, а в конце заголовка вновь вернуться в русский.

Файл в дополнительном ПЗУ может состоять из нескольких частей, размещаемых в разные адреса ОЗУ, т. е. иметь эксценты. Первая запись такого файла является основной и оформляется по правилам, описанным выше. Эксценты начинаются с кода 5ЕН и должны следовать за основной частью файла.

Пользователь имеет возможность прямо из своих программ обращаться к программе вызова файлов из дополнительного ПЗУ. Для этого после соответствующего переключения карт распределения памяти необходимо выполнить команду JMP 004FH, после чего на экран будет выдан список файлов ПЗУ и можно будет произвести выбор и запуск необходимой пользователю программы. Если обращение по этому входу произошло в отсутствие дополнительной платы ПЗУ, то на экран будет выдано сообщение об отсутствии. Таким образом, можно организовать файл в ПЗУ любой готовой программы, подготовленный на «Ирише» или другой ЭВМ.

Для работы с магнитофоном используются процедуры TAPREA — считать блок и TAPWRI — записать блок. Обе процедуры выполняют перенос блока данных длиной, задаваемой регистровой парой BC, с магнитной ленты в ОЗУ или наоборот. Начальный адрес блока в ОЗУ задается регистровой парой HL.

Процедура TAPREA производит поиск на ленте заголовка и автоматически подстраивается под скорость записи, затем переносит в ОЗУ следующий по заголовкам блок данных заданного размера. Таким образом, эта программа использует заголовки не только в качестве разделителя данных, но и для автоматической настройки на частоту записи. Это позволяет читать записи, сделанные на других ЭВМ, со скоростью 200..3000 бит/с. Определяемые программой параметры записываются в ячейки памяти и могут быть использованы другими программами. Если во время чтения возникла ошибка, то по возвращении из подпрограммы флаг CV будет установлен в единицу. При безошибочном чтении флаг равен 0. Процедура TAPREA может быть прервана нажатием на любую клавишу клавиатуры, что устанавливает признак ошибки чтения.

Процедура записи TAPWRI выполняет запись блока данных с предварительной записью заголовка. Тип заголовка указывается в аккумуляторе. Если он равен 0, происходит запись короткого заголовка, если 1 — длинного. Данные записываются с фиксированной скоростью 1200 бит/с. В процессе работы с магнитофоном

можно обращаться и к другим программам, таким как отдельная запись заголовка или чтение-запись отдельного байта. На рис. 8 приведена распечатка содержимого ПЗУ BOOT для записи в микросхемы K573PФ2 и K573PФ5.

За справками обращаться: 117234, Москва, Ленинские горы, МГУ, химический факультет. Тел. 939-44-33.

ЛИТЕРАТУРА

1. Барышников В. Н., Воронов М. А., Голутин Ю. В., Романов В. Ю., Рушайло Арно А. Б. Программное обеспечение ПЭВМ «Ириша» // Микропроцессорные средства и системы.— 1986.— № 3.— С. 59—64.
2. Барышников В. Н., Воронов М. А., Кулаков В. Б., Романов В. Ю. Программа вывода графической и символической информации ПЭВМ «Ириша» // Микропроцессорные средства и системы.— 1986.— № 4.— С. 79—89.
3. Барышников В. Н., Быстров В. П., Воронов М. А., Паначев Ф. И., Романов В. Ю. Модуль процессора персональной ЭВМ «Ириша» // Микропроцессорные средства и системы.— 1986.— № 2.— С. 52—62.
4. Романов В. Ю., Барышников В. Н., Воронов М. А., Паначев Ф. И. Графические возможности персональной ЭВМ «Ириша» // Микропроцессорные средства и системы.— 1986.— № 1.— С. 61—72.

Статья поступила 14 апреля 1987 г.

УДК 681.3.06

В. Н. Барышников, И. Н. Бондарь, М. А. Воронов,
В. Б. Кулаков, В. Ю. Романов

МОНИТОР-ОТЛАДЧИК ПРОГРАММ ДЛЯ ПЭВМ «ИРИША»

Публикуемая программа универсального монитора-отладчика, предназначенного для подготовки и запуска программ на языке ассемблера МП КР580ВМ8СА, продолжает цикл статей [1—3] по программному обеспечению ПЭВМ «Ириша». Ориентированная на минимальный комплект периферийного оборудования (без накопителя на ГМД), она позволяет создавать, просматривать, тестировать и изменять любую программу, представленную в машинных кодах. Наличие встроенного модуля ассемблера/дисассемблера предоставляет пользователю возможность программировать на уровне мнемокода команд процессора, что значительно облегчает создание программ малого объема. Кроме того, монитор обеспечивает просмотр и модификацию содержимого оперативной памяти, заполнение константами заданной области ОЗУ, перенос информации из одного участка памяти в другой и т. д. Предусмотрен обмен программами и данными с внешним носителем на магнитной ленте, а также вывод информации на принтер.

Основой для разработки монитора послужила утилита DDT операционной системы CP/M, которая подробно описана в [4, 5]. Распределение памяти во время работы монитора показано на рис. 1. Программа монито-

ра объемом около 4К байт располагается с адреса 9000H, что обусловлено требованиями к отладке различных программ на ПЭВМ «Ириша». Переносимость программ монитора на другие ЭВМ обеспечивается тем, что все машинно-зависимые процедуры ввода-вывода вынесены в отдельную область памяти в виде внешних подпрограмм. Первые 21 байт программы отведены под таблицу переходов на эти подпрограммы (табл. 1). Область ОЗУ размером 160 байт, начиная с адреса A000H, используется монитором для хранения собственных переменных. В этой же области расположен и служебный стек монитора. Команда JMP 9CFB, нахо-

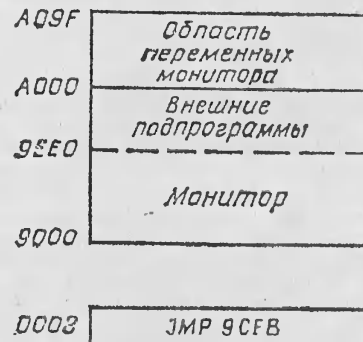


Рис. 1. Рабочая область памяти монитора-отладчика

Таблица 1

Таблица переходов на внешние подпрограммы монитора

Адрес	Обращение	Функция и параметры
9000H	JMP INIT	Программа инициализации. Аппаратно зависима. Должна заканчиваться командой JMP 9003
9003H	JMP START	Вход запуска монитора после инициализации ЭВМ
9006H	JMP STAT	Подпрограмма определения готовности клавиатуры. Возвращает монитору флаг CY=1, если клавиша нажата, иначе CY=0
9009H	JMP INPUT	Подпрограмма приема символа с клавиатуры. Возвращает монитору семитбитный код символа в рег. А и С. Все коды меньше 20H (кроме 0DH и 08H) монитором не обрабатываются
900CH	JMP OUTPUT	Подпрограмма вывода символа на экран. Получает от монитора код символа в рег. С. Все управляющие коды (включение печати и т. д.) обрабатываются этой подпрограммой
900FH	JMP READ	Программа чтения с магнитофона. Получает от монитора: HL — адрес начала буфера, BC — число считываемых байт. Оканчивается командой JMP 9AEB по ошибке чтения и командой JMP 96E0 при нормальной загрузке
9012H	JMP WRITE	Программа записи на магнитофон. Получает от монитора: HL — адрес начала буфера, BC — длина. Оканчивается командой JMP 96E0

дящаяся по адресу 0008H, предназначена для обработки возврата в монитор по команде PST1. В ПЭВМ «Ириша» эта команда продублирована в системных ПЗУ BOOT и BOOTM [3]. Остальная память принадлежит программам пользователя.

Монитор-отладчик может работать на любой ПЭВМ с системой команд, совместимой с командами МП КР580ВМ80А, и объемом памяти, не менее 4К байт ПЗУ и 1К байт ОЗУ. При переносе монитора на другую ЭВМ необходимо реализовать указанные в табл. 1 внешние подпрограммы ввода-вывода и внести соответствующие адреса в таблицу переходов. Входная точка программы — 9000H. При этом возможно размещение программы монитора как в ОЗУ, так и в ПЗУ по адресам 9000H...9FFFH. Область переменных монитора с адреса A000H должна всегда принадлежать ОЗУ.

Внешние программы для работы монитора на ПЭВМ «Ириша» (табл. 2) предназначены для организации интерфейса с консолью оператора, накопителем на МЛ и принтером УВВПЧ-30. Распайка кабеля для связи ПЭВМ с принтером показана в табл. 3. Управляющий код 10H (ctrl P) служит для включения режима печати. В этом режиме вся информация, выводимая на экран дисплея, дублируется на принтере. Повторный ввод кода 10H во время набора командной строки монитора выключает режим печати. Выключение печати осуществляется нажатием любой клавиши в процессе вы-

Таблица 2

Подпрограммы ввода-вывода монитора-отладчика

Адрес	Команда	Мисмокод	Комментарий
Подпрограмма вывода символа на дисплей			
Параметры: рег. С — код выводимого символа OUTPUT:			
9EE0	3A A093	LDA A093H	Проверка признака печати: -0 ?; да — без печати.
9EE3	B7	ORA A	
9EE4	CA 9EEC	JZ NOLIST	Сохранить код символа.
9EE7	C5	PUSH B	Напечатать символ, восстановить код символа.
9EE8	CD 9F2E	CALL LST	
9EEB	C1	POP B	
NOLIST:			
9EEC	DB 12	IN 12H	Вызвать карту распределения памяти и сохранить ее.
9EEE	F5	PUSH PSW	
9EEF	E6 F3	ANI 0F3H	Сбросить биты 2 и 3.
9EF1	D3 12	OUT 12H	Установить карту 0.
9EF3	CD 2000	CALL 2000H	Выдать символ на экран.
9EF6	F1	POP PSW	Вызвать текущую карту.
FEF7	D3 12	OUT 12H	Восстановить текущую карту.
9EF9	C9	RET	Возврат в монитор.
Подпрограмма ввода состояния клавиатуры			
Результат: CY=1 — клавиша нажата CY=0 — клавиша не нажата			
STAT:			
9EFA	DB 12	IN 12H	Сохранить текущую карту распределения памяти и установить карту 0.
9EFC	F5	PUSH PSW	
9EFD	E6 F3	ANI 0F3H	
9EFF	D3 12	OUT 12H	
9F01	CD 0058	CALL 58H	Прем статуса клавиатуры.
9F04	E1	POP H	Вызвать текущую карту в рег. H, переслать ее в рег. A, восстановить текущую карту.
9F05	7C	MOV A, H	Возврат в монитор.
9F06	D3 12	OUT 12H	
9F08	C9	RET	
Подпрограмма ввода символа с клавиатуры			
Результат: рег. AIC — код принятого символа			
INPUT:			
9F09	DB 12	IN 12H	Сохранить текущую карту распределения памяти и установить карту 0.
9F0B	F5	PUSH PSW	
9F0C	E6 F6	ANI 0F3H	
9F0E	D3 12	OUT 12H	

Адрес	Команда	Мнемокод	Комментарий
9F10	CD 005B	CALL 5BH	Прием символа с клавиатуры.
9F13	E1	POP H	Вызвать текущую карту в рег. H, сохранить принятый символ.
9F14	F5	PUSH PSW	переслать карту в рег. A, восстановить текущую карту.
9F15	7C	MOV A, H	Восстановить принятый символ, сброс старшего бита кода, сохранить код символа.
9F16	D3 12	OUT 12H	Проверка на управляющий код A P:
9F18	F1	POP PSW	нет — выход, да — вызвать признак печати, инвертировать его, записать новое значение.
9F19	E6 7F	ANI 7FH	Восстановить глубину стека.
9F1B	F5	PUSH PSW	Повторить ввод символа.
9F1C	FE 10	CPI 10H	
9F1E	C2 9F2C	JNZ LRET	
9F21	3A A093	LDA A093H	
9F24	2F	CMA	
9F25	32 A093	STA A093H	
9F28	F1	POP PSW	
9F29	C3 9F09	JMP INPUT	
9F2C	F1	LRET: POP PSW	
9F2D	C9	RET	Восстановить принятый символ, возврат в монитор.

Подпрограмма вывода символа на принтер
 Параметр:
 рег. C — код символа

Адрес	Команда	Мнемокод	Комментарий
9F2E	3E 0A	MVI A, 0AH	
9F30	B9	CMP C	Проверка на код перевода строки: если нет — печать символа, да — отработка перевода строки.
9F31	C2 9F45	JNZ LLIST	Загрузить рег. B — счетчик пробелов, рег. C — код пробела.
9F34	CD 9F45	CALL LLIST	Печать 8 пробелов: сохранить счетчик и символ, печать пробела, восстановить рег. пару BC, уменьшить счетчик пробелов.
9F37	06 08	MVI B, 8	Конечная часть цикла ?
9F39	0E 20	MVI D, 20H	нет — продолжить, да — возврат в монитор.
9F3B	C5	PUSH B	
9F3C	CD 9F45	CALL LLIST	
9F3F	C1	POP B	
9F40	05	DCR B	
9F41	C2 9F3B	JNZ PRBL	
9F44	C9	RET	

Драйвер принтера
 Параметр:
 рег. C — код символа

Адрес	Команда	Мнемокод	Комментарий
9F45	CD 9F7A	CALL L1	Проверка готовности принтера.

Автор	Команда	Мнемокод	Комментарий
9F48	CD 9F67	CALL TEST	Проверка отмены печати.
9F4B	B7	ORA A	Установка флага Z.
9F4C	CA 9F45	JZ LLIST	Нет готовности — ожидание.
9F4F	79	MOV A, C	Принтер готов, выдать символ в порт A БИС ППА.
9F50	D3 10	OUT 10H	Сохранить текущую карту распределения памяти.
9F52	DB 12	IN 12H	
9F54	F5	PUSH PSW	
9F55	F6 83	ORI 83H	Сформировать строб принтера, выдать строб принтера.
9F57	D3 12	OUT 12H	
9F59	DB 11	L10: IN 11H	Прием байта состояния принтера, проверка подтверждения.
9F5B	E6 20	ANI 20H	Проверка отмены печати.
9F5D	CD 9F67	CALL TEST	Ожидание подтверждения принтера.
9F60	CA 9F59	JZ L10	Восстановить текущую карту распределения памяти, возврат из подпрограммы.
9F63	F1	POP PSW	
9F64	D3 12	OUT 12H	
9F66	C9	RET	

Подпрограмма проверки отмены печати.
 Результат: отмена печати по нажатию любой клавиши

Адрес	Команда	Мнемокод	Комментарий
9F67	F5	PUSH PSW	Сохранить рег. A и флаги.
9F68	CD 9006	CALL 9006H	опрос клавиатуры: клавиша не нажата — возврат, клавиша нажата — отмена печати.
9F6B	D2 9F78	JNC NOPR	
9F6E	F1	POP PSW	
9F6F	3A A093	LDA A093H	Вызов признака печати, инверсия признака, запомнить новое значение.
9F72	2F	CMA	
9F73	32 A093	STA A093H	Изменение адреса возврата, возврат по отмене печати.
9F76	F1	POP PSW	
9F77	C9	RET	
9F78	F1	NOPR: POP PSW	Восстановить рег. A и флаги, возврат на следующий шаг.
9F79	C9	RET	

Ввод слова состояния принтера
 Результат:
 рег. A=0 — принтер не готов
 рег. A=1 — принтер готов

Адрес	Команда	Мнемокод	Комментарий
9F7A	DB12	L1: IN 12H	Сохранить текущую карту в регистре B.
9F7C	47	MOV B, A	

Адрес	Команда	Мнемокод	Комментарий
9F7D	F6 81	ORI 81H	Выбор и установка функций порта В БИС ППА. Прием слова состояния, проверка готовности принтера. Восстановить текущую карту распределения памяти. Признак готовности — в рег. А, возврат с признаком готовности. Признак готовности в рег. А, возврат по неготовности.
9F7F	D3 12	OUT 12H	
9F81	DB 11	IN 11H	
9F83	E6 30	ANI 30H	
9F85	78	MOV A, B	
9F86	D3 12	OUT 12H	
9F88	3E 01	MVI A, 1	
9F8A	C8	RZ	
9F8B	AF	XRA A	
9F8C	C9	RET	

Программа инициализации памяти

Адрес	Команда	Мнемокод	Комментарий
9F8D	31 A080	INIT: LXI SP, 0A080H	Установить служебный стек. Установить рабочую карту 1, подключить ОЗУ. Переход в монитор.
9F90	DB 12	IN 12H	
9F92	F6 08	ORI 08H	
9F94	D3 12	OUT 12H	
9F96	C3 9003	JMP 9003H	

Программа считывания с МЛ

Адрес	Команда	Мнемокод	Комментарий
9F99	DB 12	READ: IN 12H	Сохранить текущую карту распределения, подготовить и установить карту 0. Чтение с МЛ. Выход по ошибке чтения. Восстановить текущую карту распределения памяти. Возврат в монитор. Восстановить текущую карту распределения памяти. Возврат по ошибке чтения.
9F9B	F5	PUSH PSW	
9F9C	E6 F3	ANI 0F3H	
9F9E	D3 12	OUT 12H	
9FA0	CD 0040	CALL 0040H	
9FA3	DA 9FAC	JC OUTER1	
9FA6	F1	POP PSW	
9FA7	D3 12	OUT 12H	
9FA9	C3 96E0	JMP 96E0H	
9FAC	F1	OUTER1: POP PSW	
9FAD	D3 12	OUT 12H	
9FAF	C3 9AEB	JMP 9AEBH	

Программа записи на МЛ

Адрес	Команда	Мнемокод	Комментарий
9FB2	DB 12	WRITE IN 12H	Сохранить текущую карту распределения, подготовить и установить карту 0.
9FB4	F5	PUSH PSW	
9FB5	E6 F3	ANI 0F3H	
9FB7	D3 12	OUT 12H	

Адрес	Команда	Мнемокод	Комментарий
9FB9	AF	XRA A	Гашение экрана для ускорения доступа процессора к ОЗУ. Запись на МЛ. Восстановление вывода изображения на экран после завершения записи. Восстановить текущую карту распределения памяти. Возврат в монитор.
9FBA	D3 D8	OUT 0D8H	
9FBC	CD 0043	CALL 0043H	
9FBF	3E 1A	MVI A, 1AH	
9FC1	D3 D8	OUT 0D8H	
9FC3	F1	POP PSW	
9FC4	D3 12	OUT 12H	
9FC6	C3 96E0	JMP 96E0H	

Таблица 3

Таблица соединений для ПЭВМ „Ириша“ и принтера УВВПЧ-30

Модуль процессора, разъем X1 тип СНО-51-20-2-В		Принтер УВВПЧ-30 ГРПМШ-1-31-ШУ2	
контакт	сигнал	контакт	сигнал
15	P0	Б-14	D0
13	P1	Б-13	D1
11	P2	Б-12	D2
9	P3	Б-11	D3
7	P4	Б-10	D4
5	P5	Б-8	D5
3	P6	Б-9	D6
1	P7	Б-7	D7
12	SO2	Б-15	ГП
19	SO1	Б-16	ГИ
17	SO2	Б-6	СТР
10	SI1	Б-4	ЗП
20	SI3	Б-1	СП1
18	SI4	Б-3	СП2
16	SI5	Б-2	СП3
14	SI6	Б-5	СП4
6, 8, 2	GND	A1...A15	ОВ

дачи. Управляющие коды, за исключением 0DH, 7FH, 08H, которые обрабатываются монитором, передаются с клавиатуры прямо на программу CONOUT [2]. Внешние подпрограммы монитора широко используют программы, записанные в системном ПЗУ ПЭВМ. Для размещения внешних подпрограмм отведена промежуточная зона памяти с адресами 9EE0H...9FFF.

Часть «прошивки» ПЗУ BOOTM, содержимое которой может быть пристыковано к «прошивке» ПЗУ BOOT [3], начиная с адреса 0800H, приведена на рис. 2. Программа инициализации, находящаяся в ПЗУ BOOT, автоматически определяет наличие монитора в исходной конфигурации и позволяет пользователю запустить его из стартового меню. Если монитор выбран, то управление передается по адресу 0800H, где находится программа переноса отладчика в рабочую область ОЗУ, которая после переноса запускает монитор. Собственно программа монитора хранится в ПЗУ по адресам 0810H...16EFH, а начиная с адреса 16F0H расположен

Рис. 2. (Начало) →

0800: 21 10 08 11 00 90 01 00 10 CD 6F 00 C3 00 90 FF
0810: C3 8D 9F C3 15 90 C3 FA 9E C3 09 9F C3 8D 9E C3
0820: H2 9F C3 99 9F 21 FB 9C 22 09 00 3E C3 32 08 00
0830: 3E 20 32 2B A0 21 2C 97 11 83 A0 06 0B 7E 12 23
0840: 13 05 C2 2D 90 21 4A 90 4E AF 32 93 A0 B9 CA B3
0850: 96 23 E5 CD 0C 90 E1 C3 38 90 0C 0E 69 72 69 7R
0860: 61 20 6F 74 6C 61 64 7E 69 6B 0F 0D 0A 00 F5 79
0870: CD 60 9E F1 C9 FE 20 C8 FE 09 C8 FE 2C C8 FE 0D
0880: C8 FE 7F CA 5D 95 C9 0E 0D CD 5E 90 0E 0A CD 5E
0890: 90 C9 CD 04 9B FE 0D CA 51 95 CD 65 90 CA 82 90
08A0: 0E 04 21 09 A0 36 20 23 0D C2 95 90 0E 05 21 09
08B0: A0 77 CD 04 9B CD 65 90 CA B3 90 23 0D CA 51 95
08C0: C3 A1 90 3A 09 A0 FE 20 C9 CD A9 9B 3D C2 51 95
08D0: EB 4E 23 46 79 05 04 C9 CD B9 90 C2 51 95 C9 17
08E0: 17 17 E6 38 C9 17 17 17 E6 30 C9 E2 2A 09 A0
08F0: EB 7B BE C2 EE 90 23 7A BE C8 2B 2B 0D C2 E1
0900: 90 0D C9 06 04 D5 11 09 A0 1A BE C2 06 91 23 13
0910: 05 C2 F9 90 D1 C9 23 05 C2 06 91 11 F8 FF 19 D1
0920: 13 0D C2 F3 90 0D C9 C5 CD 82 90 CA 51 95 0E 08
0930: 21 99 96 CD DC 90 C2 51 95 0D 79 C1 C9 C5 CD 82
0940: 90 CA 51 95 0E 05 21 AH 96 CD F3 90 C2 51 95 0D
0950: 79 C1 C9 CD 2D 91 FE 04 CA 51 95 C9 CD 2D 91 FE
0960: 03 CA 51 95 FE 04 0D 3D C9 21 09 A0 11 0A 0A 9E
0970: 02 1A 77 23 13 0D C2 61 91 1A FE 20 C2 51 95 77
0980: 21 89 96 0E 08 CD DC 90 C2 51 95 0D 79 CD CF 90
0990: C9 CD 59 91 F5 CD H9 90 F1 F6 C0 C9 1A 2A 05 A0
09A0: 77 23 22 05 A0 C9 CD 04 9B FE 0D CA 79 95 FE 2E
09B0: CA 79 95 CD 85 90 CA 51 95 0E 11 21 DF 95 11 7E
09C0: 95 CD F3 90 C2 BA 91 C3 8C 91 0E 0A 21 07 96 CD
09D0: F3 90 C2 CE 91 CD 8C 91 CD C8 90 C3 8D 91 0E 06
09E0: 21 1F 96 CD F3 90 C2 86 91 CD 8C 91 CD B9 90 CD
09F0: 8D 91 78 C3 8D 91 0E 01 21 23 96 CD F3 90 C2 02
0AA0: 92 CD 17 91 CD CF 80 47 0E 40 CD 17 91 B1 B0 C3
0A10: 8D 91 0E 08 21 43 96 CD F3 90 C2 18 92 0D 79 90
0A20: CF 90 47 0E 80 C3 FA 91 0E 02 21 4B 96 CD F3 90
0A30: C2 30 92 0C 0C CD 17 91 CD CF 90 B1 C3 8D 91
0A40: 0E 01 21 4E 96 CD F3 90 C2 4C 92 CD 17 91 CD CF
0A50: 90 F6 06 CD 8D 91 CD C8 90 C3 8D 91 0E 06 21 67
0A60: 96 CD F3 90 C2 72 92 79 FE 04 DA 5F 92 C6 05 47
0A70: CD 43 91 CD D5 90 B0 CD 8D 91 EG CF FE 01 C0 C5
0A80: DC 91 0E 01 21 6B 96 CD F3 90 C2 8D 92 CD C8 90
0A90: FE 08 D2 51 95 CD CF 90 F6 C7 C3 8D 91 0E 02 21
0AA0: 77 96 CD F3 90 C2 AD 92 0D C2 A1 92 0F C1 C3 A3
0AB0: 92 0E C5 CD 41 CD D5 90 B1 C3 8D 91 3A 09 A0
0AC0: FE 4A C2 8D 92 CD 81 91 F6 02 C3 C7 92 FE 43 C2
0AD0: D2 92 CD 81 91 F6 04 CD 8D 91 79 CD 8D 91 78 C3
0AE0: 8D 91 FE 52 C2 51 95 CD 59 91 F6 C0 C3 8D 91 2A
0AF0: 02 A0 D5 EB 2A 00 A0 7B 95 7A 9C D2 F3 92 2A 07
0B00: A0 F9 C9 D1 7E 23 22 00 A0 C9 3C E6 07 FE 06 DA
0B10: 04 93 06 03 FE 05 DA 0B 93 C6 02 C6 41 4F C3 5E
0B20: 90 47 E6 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F 0F
0B30: 5E 90 78 E6 0F C6 90 27 CE 40 27 4F C3 5E 90 06
0B40: 04 4E CD 5E 90 23 05 C2 31 93 0E 20 C3 5E 90 7A
0B50: E6 38 0F 0F 0F C9 CD 3F 93 87 4F 21 7B 96 09 4E
0B60: CD 5E 90 23 4E CD 5E 90 0E 20 CD 5E 90 C3 5E 90
0B70: CD 3F 93 E6 0E FE 06 C2 FA 92 0E 53 CD 5E 90 0E
0B80: 50 C3 5E 90 CD 77 90 2A 00 A0 7C CD 11 93 7D CD
0B90: 11 93 0E 20 CD 5E 90 CD 5E 90 C9 21 00 00 39 22
0BA0: 07 A0 3A 04 A0 B7 CA AD 93 21 FF FF 22 02 A0 3C
0BB0: C2 AD 93 3C 32 04 A0 2A 00 A0 C3 D0 93 CD BK 9E
0BC0: DA 79 95 21 04 A0 7E B7 CA BF 93 35 CA 79 95 2A
0BD0: 00 A0 CD 77 90 0E 20 CD 5E 90 CD 5E 90 CD 77 93
0BE0: CD DF 92 57 21 7E 95 01 11 06 BK CA 36 95 23 0D
0BF0: C2 DA 93 0E 0A BK CA 22 95 23 0D C2 E5 93 0E 06
0C00: BK CA 07 95 23 0D C2 F0 93 E6 C0 FE 40 CA E0 94
0C10: FK 80 CA DE 9A 7A E6 C7 D6 04 CA CF 94 3D CA C9
0C20: 94 3D CA B5 94 7A E6 C0 CA 83 94 7A E6 07 CA 7B
0C30: 94 D6 02 CA 6D 94 D6 02 CA 62 94 D6 03 CA 53 94
0C40: 7A E6 08 C2 44 95 7A E6 07 4F 3D 21 72 96 09 CD
0C50: 2F 93 CD 3F 93 FK 06 C2 D3 94 21 6F 96 CD 2F 93
0C60: C3 AD 93 21 6B 96 CD 2F 93 CD 3F 93 CD 11 93 C3
0C70: AD 93 0E 43 CD 5E 90 CD 46 93 C3 12 95 0E 4A CD

0C80: 5E 90 CD 46 93 C3 12 95 0E 52 CD 5E 90 CD 46 93
0C90: C3 AD 93 21 53 96 7A E6 07 CA 44 95 7A E6 0F 3D
0CA0: CA A7 94 FE 03 DA 9A 94 D6 05 87 87 4F 09 CD 2F
0CB0: 93 CD 60 93 C3 AD 93 CD 2F 93 CD 60 93 0E 2C CD
0CC0: 5E 90 C3 12 95 21 4F 96 CD 2F 93 CD 3F 93 CD FA
0CD0: 92 0E 2C CD 5E 90 C3 2D 95 21 4E 96 C3 D2 94 21
0CE0: 47 96 CD 2F 93 CD 3F 93 CD FA 92 C3 AD 93 7A E6
0CF0: 38 0F 4F 21 27 96 09 CD 2F 93 C3 FE 94 21 23 96
0D00: CD 2F 93 CD 3F 93 CD FA 92 0E 2C CD 5E 90 7A E6
0D10: 07 CD FA 92 C3 AD 93 79 87 87 4F 21 07 96 09 CD
0D20: 2F 93 CD DF 92 F5 CD DF 92 57 F1 5F CD 46 9B C3
0D30: AD 93 79 87 87 4F 21 DF 95 09 CD 2F 93 CD DF 92
0D40: CD 2C 9B C3 AD 93 79 87 87 4F 21 9B 95 09 CD 2F
0D50: 93 C3 AD 93 21 AF 96 CD 2F 93 7A CD 2C 9B C3 AD
0D60: 93 CD 77 90 0E 3F CD 5E 90 2A 07 A0 F9 21 06 00
0D70: 39 22 07 A0 CD 74 93 22 05 A0 CD 6B 9E CD 96 91
0D80: 2A 05 A0 22 06 A0 C3 64 95 2A 07 A0 F9 C0 00 C7
0D90: 0F 17 1F 27 2F 3F 76 C9 E3 E9 EB F3 F9 FE C6
0DA0: CE D3 D6 DB DE EG EE EF FE 22 2A 32 3A C3 CD 45
0DB0: 49 20 20 53 50 48 4C 44 49 20 20 58 43 48 47 50
0DC0: 43 48 4C 58 54 48 4C 52 45 54 20 48 4C 54 20 43
0DD0: 4D 43 20 53 54 43 20 43 4D 41 20 44 41 41 20 52
0DE0: 41 52 20 52 41 4C 20 52 52 43 20 52 4C 43 20 4E
0DF0: 4F 50 20 43 50 49 20 4F 52 49 20 58 52 49 20 41
0E00: 4E 49 20 53 42 49 20 49 4E 20 20 53 55 49 20 4F
0E10: 55 54 20 41 43 49 20 41 44 49 20 43 41 4C 4C 4A
0E20: 4D 50 20 4C 44 41 20 53 54 41 20 4C 48 4C 44 53
0E30: 48 4C 44 4D 4F 56 20 41 44 44 20 41 44 43 20 53
0E40: 55 42 20 53 42 42 20 41 4E 41 20 58 52 41 20 4F
0E50: 52 41 20 43 4D 50 20 49 4E 52 20 44 43 52 20 4D
0E60: 56 49 20 4C 58 49 20 53 54 41 58 49 4E 58 20 44
0E70: 41 44 20 4C 44 41 58 44 43 58 20 52 53 54 20 50
0E80: 53 57 20 50 4F 50 20 50 55 53 48 4E 5A 5A 20 4K
0E90: 43 43 20 50 4F 50 45 50 20 4D 20 42 20 43 20 44
0EA0: 20 45 20 48 20 4C 20 4D 20 41 20 42 20 20 20 44
0EB0: 20 20 20 48 20 20 20 53 50 20 20 50 53 57 20 3F
0EC0: 3F 3D 20 32 15 A0 32 92 A0 21 06 01 22 00 A0 22
0ED0: 23 A0 22 8E A0 22 90 A0 22 4D A0 22 7F A0 31 7D
0EE0: A0 E5 21 02 00 E5 2B 23 22 7D A0 E5 E5 22 13 A0
0EF0: 31 75 A0 CD 3C 9B 3E 3E CD 60 9E CD 6B 9E CD 04
0F00: 9B FE 0D CA E0 96 D6 41 DA EB 9A FE 1A D2 EB 9A
0F10: 5F 16 00 21 AE 97 19 19 5E 23 56 EB E9 CD A9 9B
0F20: FE 02 C2 EB 9A CD 7F 9B 22 23 A0 7D 32 84 A0 CD
0F30: 7F 9B 22 25 A0 7D 32 86 A0 C3 83 A0 3E 00 D3 00
0F40: C3 E0 96 DB 00 C9 CD A9 9B FE 02 C2 EB 9A CD 7F
0F50: 9B E5 CD 7F 9B 44 4D E1 C3 12 90 CD A9 9B FE 02
0F60: C2 EB 9A CD 7F 9B E5 CD 7F 9B 44 4D E1 C3 0F 90
0F70: CD A9 9B 3D C2 EB 9A CD 7F 9B 22 23 A0 7D 32 8B
0F80: A0 CD 8A A0 F5 CD 77 90 CD 2C 9B CD 77 90 3E 37
0F90: F5 CD 60 9E F1 3D FE 2F C2 80 97 CD 77 90 3E 08
0FA0: 32 23 A0 F1 07 F5 3E 30 D2 9D 97 3E 31 CD 60 9E
0FB0: 3A 23 A0 3D 32 23 A0 C2 93 97 F1 C3 8D 96 E2 97
0FC0: EB 9A EB 9A 2A 9B EB 9A C0 96 00 99 80 99 60 97
0FD0: EB 9A EB 9A FB 27 AA 99 EB 9A 0B 97 DA 98 EB 9A
0FE0: 36 9F EC 99 35 9A 31 9A EB 9A 4B 97 5E 9A EB 9A
0FF0: EB 9A CD E3 99 D2 EB 9A CD A9 9B 3D C2 EB 9A CD
1000: 7F 9B 22 00 A0 CD 5D 95 C3 80 96 CD E3 99 D2 EB
1010: 9A CD A9 9B CA 1F 98 CD 7F 9B 22 00 A0 3D CA 1F
1020: 98 CD 7F 9B 22 00 A0 3D C2 EB 9A CD C3 21 98 3E
1030: 0C 32 04 A0 CD 8B 93 C3 80 96 CD A9 9B CA 49 98
1040: CD 7F 9B DA 33 98 22 23 A0 E6 7E 3D CA 49 98 CD
1050: 7F 9B 3D C2 EB 9A C3 54 96 2A 23 A0 7D E6 F0 6F
1060: 11 7F 00 19 22 25 A0 CD 3C 9B CD BC 9E DA 0E 96
1070: 2A 23 A0 22 27 A0 CD 47 9B CD F6 9A 7E CD 2C 9B
1080: 23 CD 5E 9B DA 7D 98 7D E6 07 C2 69 98 22 23 A0
1090: 2A 21 A0 EB CD F6 9A 1A CD 4F 9B 13 2A 23 A0 7D
10A0: 93 C2 87 98 7C 92 C2 A9 87 98 2A 23 A0 CD 5E 9B DA
10B0: F0 96 C3 57 98 CD A9 87 98 C2 C2 EB 9A CD 7F 9B
10C0: E5 CD 7F 9B E5 CD 7F 9B D1 C1 C9 7B 91 7A 98 C9
10D0: CD A5 98 7C B7 C2 EB 9A CD B8 9E DA 0E 96 7D 02
10E0: 03 C3 8B 98 CD 3C 9B 2A 90 A0 7F 9E CD E2 EB 9A
10F0: 54 5D 13 13 13 3A 92 A0 B7 CA F6 98 E5 2A 7B A0

1100:	23 23 22 7B A0 E1 AF 32 92 A0 3E 02 B7 C3 19 99	1470:	06 00 CD 3C 9B C5 E5 CD 33 9C E1 C1 04 23 7F FK
1110:	AF 32 92 A0 CD 3C 9B CD A9 9B CD 7F 9B E5 CD 7F	1480:	05 C2 77 9C CD 77 90 FE 0B D2 92 9C FK 09 C7 87
1120:	9B E5 CD 7F 9B 44 4D D1 E1 EB 22 8E A0 EB EB K5	1490:	9C CD 77 90 C3 65 9C FE 06 DA 65 9C CD F6 9A C3
1130:	2A 8E A0 22 90 A0 E1 22 BE A0 EB F3 CA 47 99 DA	14A0:	65 9C CD F6 9A CD B8 9D F5 D5 C5 CD E3 99 D2 B2
1140:	35 99 22 7F A0 E6 7F 3D CA 47 99 CD 58 99 3D CA	14B0:	9C 2A 77 A0 22 00 A0 21 04 A0 36 FF CD 8B 93 C3
1150:	47 99 59 50 CD 58 99 31 75 A0 D1 C1 F1 E1 F9 2A	14C0:	DB 9C 2E 22 25 A0 2A 7F A0 7E CD 2C 9B 23 CD 5E
1160:	7F A0 E5 2A 7D A0 FB C9 F5 C5 21 15 A0 7E 34 B7	14D0:	9B DA D3 9C F5 CD F6 9A F1 B3 CA D7 9C 5E 23 56
1170:	CA 73 99 23 7E 23 46 23 BB C2 73 99 78 BA C2 73	14E0:	EB CD 47 9B C3 DB 9C 7E CD 2C 9B C1 D1 F1 C9 43
1180:	99 7E 12 23 73 23 72 23 1A 77 3E CF 12 C1 F1 C9	14F0:	5A 4D 45 49 41 42 44 48 56 50 F6 FA FC FA FE 01
1190:	CD A9 9B FE 02 C2 EB 9A CD 7F 9B E5 CD 7F 9B D1	1500:	07 08 03 05 21 00 00 22 13 A0 C9 F3 22 7D A0 E1
11A0:	E5 CD 3C 9B 19 CD 47 9B CD F6 9A E1 AF 95 6F 3E	1510:	2B 22 7F A0 F5 21 C2 00 39 F1 31 7D A0 E5 F5 B7
11B0:	00 9C 67 19 CD 47 9B CD A5 98 CD BB 98	1520:	D5 2A 9F A0 7E FE CF F5 E5 21 15 A0 7E 36 00 75
11C0:	DA E0 96 0A 03 77 23 C3 AD 99 21 65 00 7E E6 7F	1530:	3A 30 7D AD 23 5E 23 56 23 7E 12 78 C3 1F 9D
11D0:	FE 48 C0 23 7E E6 7F FE 45 C0 23 7E E6 7F FE 58	1540:	E1 F1 CA 51 9D 23 22 7F A0 EB 01 FF FF CD BB 98
11E0:	C9 EB 2A 4D A0 7D 93 7C 9A EB C9 CD D1 99 D0 22	1550:	DA 51 9D CD F4 9C 2A 10 A0 EB 3E 82 B7 37 C3 1E
11F0:	4D A0 C9 E5 21 FF FF CD D1 99 E1 C9 CD A9 9B 3D	1560:	99 FB 2A 13 A0 7C B5 CA 77 9D 2B 22 13 A0 CD BC
1200:	C2 EB 9A CD 7F 9B CD 3C 9B E5 CD 47 9B CD F6 9A	1570:	9E DA 77 9D 3A 12 A0 B7 C2 71 9D CD 8B 9D C3 1E
1210:	E1 7E E5 CD 2C 9B CD F6 9A CD 6B 9E CD 04 9B E1	1580:	99 CD 5D 9C C3 1E 99 CD F4 9C 3E 20 CD 60 9E 3E
1220:	FE 0D CA 2D 9A FE 2E CA E0 96 E5 CD AC 9B 3D C2	1590:	20 CD 60 9E 3E 2A CD 60 9E 2A 7F A0 CD E3 99 D2
1230:	EB 9A CD 7F 9B 7C B7 C2 EB 9A 7D E1 77 23 C3 F6	15A0:	95 9D 22 00 A0 CD 47 9B A0 2D A0 22 23 A0 C3 E0
1240:	99 AF C3 37 9A 3E FF 32 12 A0 3E 01 32 9B A0 CD	15B0:	9E 11 0D 00 21 C6 9E 7A A0 7B BE 23 CA B4 9D 14
1250:	A9 9B 21 00 00 CA 55 9A 3D C2 EB 9A CD 7F 9B 7D	15C0:	1D C2 A7 9D 5A 16 00 C9 2A 7F A0 46 23 E5 CD A1
1260:	B4 CA EB 9A 2B 22 13 A0 CD 5E 9C C3 1E 99 CD 04	15D0:	9D 21 0F A0 73 21 CF 9D 19 19 5E 23 5E EB E9 EB
1270:	9B FE 0D C2 6C 9A CD 5E 9C C3 E0 96 01 0B 00 21	15E0:	9D 11 9E EB 9D 11 9E F1 9D 23 9E 35 9E 57 9E 57
1280:	DF 9C BE CA 7F 9A 23 04 0D C2 72 9A C3 EB 9A CD	15F0:	9E 54 9E 54 9E 4A 9F 57 9E 45 9E CD FF 9D C2 5A
1290:	04 9B FE 0D C2 EB 9A C5 CD 3C 9B CD 33 9C CD F6	1600:	9E CD 0A 9E C3 5A 9E 3E FF BB C0 3E FF BA C9 C1
12A0:	9A CD 6B 9E CD A9 9B E7 CA E0 96 3D C2 EB 9A CD	1610:	E1 5E 23 56 23 E5 C5 C3 F7 9D 2A 7B A0 5E 23 56
12B0:	7F 9B C1 7E 9E 05 D2 00 9A 7C B7 C2 EB 9A 7D FE	1620:	C9 CD FF 9D CA 1E 9E C1 C5 3E 02 C3 5C 9E D1 D5
12C0:	02 D2 EB 9A CD FC 9B 67 41 3E FE CD CA 9A A4 41	1630:	C3 5A 9E 78 FE CF C2 2D 9E AF C3 5E 9E E6 38 5E
12D0:	67 7D CD CA 9A B4 12 C3 E0 96 05 C8 07 C3 CA 9A	1640:	16 00 C3 5A 9E 2A 7D A0 EB CD F7 9D C2 5A 9E C3
12E0:	C2 E0 9A 7C B7 C2 EB 9A 7D 21 7A A0 77 C3 K0 96	1650:	F1 9D C3 5A 9E D1 D5 C3 5A 9E CD 0A 9E C1 C5 3E
12F0:	E5 CD 1A 9C D1 73 23 72 C3 K0 96 CD 3C 9B 3E 3F	1660:	02 C3 5C 9E D1 13 25 D1 13 D5 3E 01 3C 37 E1 C9
1300:	CD 60 9E C3 E0 96 3E 20 C3 60 9E FE 7F C8 FE 61	1670:	E5 D5 C5 4E CD 0C 90 C1 D1 E1 C9 11 2D A0 06 00
1310:	D8 E6 5F C9 E5 21 2C A0 7E B7 3E 0D CA 1B 9B 35	1680:	C5 D5 CD 09 90 FE 08 CA 97 9E FE 7F CA 97 9E F3
1320:	2A 29 A0 7E 23 22 29 A0 CD FB 9A E1 C9 FE 0A D2	1690:	CD 60 9E F1 D1 C1 12 13 FE 0D CA B2 9E 04 3E 23
1330:	27 9B C6 30 C3 60 9E C6 37 C3 60 9E F5 1F 1F 1F	16A0:	B8 CA EB 9A C3 70 9E C1 C1 AF B0 CA 70 9E 05 1B
1340:	1F E6 0F CD 1D 9B F1 E6 0F C3 1D 9B 3E 0D CD 60	16B0:	3E 08 CD 60 9E 3E 20 CD 60 9E 3E 08 CD 60 9E C3
1350:	9E 3E 0A C3 60 9E EB 7C CD 2C 9B 7D C3 2C 9B FE	16C0:	70 9E 21 2D A0 22 29 A0 2B 70 23 C9 C5 D5 E5 CD
1360:	7F D2 59 9B FE 20 D2 60 9E 3E 2K C3 60 9E EB 2A	16D0:	06 90 E1 D1 C1 C9 FF C3 C7 C2 FF CD C7 C4 FF C9
1370:	25 A0 7D 93 6F 7C 9A EB C9 FE 0D C8 FE 2C C8 FE	16E0:	C7 C7 FF FF C7 06 C7 C6 C7 C6 01 E7 22 C7 C0 F7 D3
1380:	20 C9 D6 30 FE 0A D8 C6 F9 FE 10 D8 C3 EB 9A EB	16F0:	3A 93 A0 B7 CA EC 9E C5 CD 2E 9F C1 DB 12 F5 E6
1390:	5E 23 56 23 C3 9B EB 21 00 00 CD 72 9B 29 29 29	1700:	F3 D3 12 CD 00 20 F1 D3 12 C9 DB 12 F5 E6 F3 D3
13A0:	29 B5 6F CD 04 9B CD 69 9B C2 BA 9B EB C9 73 23	1710:	12 CD 58 00 E1 7C D3 12 C9 DB 12 F5 E6 F3 D3 12
13B0:	72 23 E5 21 1C A0 34 E1 C9 CD 04 9B 21 1C A0 36	1720:	CD 5B 00 E1 F5 7C D3 12 F1 E6 7F F5 FE 10 C2 2C
13C0:	00 23 FE 0D CA EB 9B FE 2C C2 C7 9B 3E 80 32 1C	1730:	9F 3A 93 A0 2F 32 93 A0 F1 C3 09 9F F1 C9 3E 0A
13D0:	A0 11 00 00 C3 CA 9B CD 86 9B CD 9E 9B FE 0D CA	1740:	B9 C2 45 9F CD 45 9F 06 08 0E 20 C5 CD 45 9F C1
13E0:	FE 9B CD 04 9B CD 86 9B CD 9E 9B FE 0D CA FE 9B	1750:	05 C2 3B 9F C9 CD 7A 9F CD 67 9F B7 CA 45 9F 79
13F0:	CD 04 9B CD 86 9B CD 9E 9B FE 0D C2 EB 9A 11 1C	1760:	D3 10 DB 12 F5 F6 83 D3 12 DB 11 E6 20 CD 67 9F
1400:	A0 1A FE 01 CA EB 9A 13 37 07 0F C9 E5 21 FF 9C	1770:	CA 59 9F F1 D3 12 C9 F5 CD 06 90 D2 78 9F F1 3A
1410:	58 16 00 19 4E 21 79 A0 7E EB E1 C9 CD FC 9B 0D	1780:	93 A0 2F 32 93 A0 F1 C9 F1 C9 DB 12 47 F6 81 D3
1420:	CA 17 9C 1F C3 0F 9C E6 01 C9 D6 06 21 EA 9C 5F	1790:	12 DB 11 E6 30 78 D3 12 3E 01 C8 AF C9 31 8C A0
1430:	16 00 19 5E 16 FF 21 81 A0 19 C9 CD 1A 9C 5E 23	17A0:	DB 12 F6 08 D3 12 C3 03 90 DB 12 F5 E6 F3 D3 12
1440:	56 EB C9 7E CD 60 9E 78 FE 05 D2 44 9C CD 0C 9C	17B0:	CD 40 00 DA AC 9F F1 D3 12 C3 E0 96 F1 D3 12 C3
1450:	CD 1D 9B C9 F5 3E 3D CD 60 9E F1 C2 56 9C 21 7A	17C0:	EB 9A DB 12 F5 E6 F3 D3 12 AF D3 D8 CD 43 00 3E
1460:	A0 7E CD 2C 9B C9 CD 2B 9C CD 47 9B C9 21 DF 9C	17D0:	1A D3 D8 F1 D3 12 C3 E0 96 00 00 00 00 00 00

Рис. 2. (Окончание)

блок внешних подпрограмм, переносимых вместе с монитором. Возможен вариант загрузки монитора с магнитофона или платы дополнительного ПЗУ [3].

После старта монитор устанавливает программный счетчик и стек программы пользователя на ячейку с адресом 0100H и ожидает ввода команд. Диалоговый отладчик обеспечивает выполнение стандартных функций своего дискового аналога. Кроме того, этот набор дополнен функциями для работы с файлами на магнитной ленте и функциями доступа к портам ввода-вывода. Обращение к функциям отладчика осуществляется по-

средством команд, вводимых оператором с клавиатуры. Любая команда представляет собой ключевую букву (имя команды), за которой без пробела следуют несколько (или ни одного) параметров. Разделителем параметров является символ «». Все задаваемые параметры и результаты, выводимые на экран, представляются в шестнадцатеричной системе счисления. Число параметров не превышает трех. Приглашением к вводу служит знак «>» в крайней левой позиции строки. Ошибочная команда игнорируется, и монитор реагирует на это символом «?».

Команда D (Display) — просмотр — обеспечивает вывод содержимого блока ячеек памяти в шестнадцатеричном и символьном видах. Адреса начала и конца выводимого блока указываются в команде. Если адрес конца опущен, то выводятся 16 строк по 8 байт в каждой. Управляющие коды в поле символов помечаются точками. Если отсутствует и адрес начала, то вывод начинается с байта, следующего за последним байтом, выведенным предыдущей командой D. Избыточная выдача прекращается по нажатию любой клавиши. Например, команда D100,105 выведет содержимое памяти из ячеек 0100H...0105H включительно, и по команде выдача будет продолжена с ячейки 0106H.

Команда S (Set) — установка. С помощью этой команды можно побайтно просмотреть и изменить содержимое памяти. Адрес начала указывается в команде, отладчик выводит адрес текущей ячейки, ее содержимое и ожидает ввода значения для записи в эту ячейку. Если значение ячейки изменить не нужно, то по нажатию клавиши BK просмотр памяти продолжится. Выход из команды — ввод любого символа, отличного от шестнадцатеричной цифры (обычно символ «.»). Например, команда S100 позволит просмотреть и при необходимости изменить ряд значений, начиная с адреса 0100H.

Команда F (Fill) — заполнение — заполняет указанный массив ячеек памяти заданным кодом. Параметрами являются начальный и конечный адреса массива и код заполнения. Например, команда F100,1FFF,E5 заполнит ОЗУ по адресам 0100H...01FFFH кодом E5H.

Команда M (Move) — копирование массива значений из одного места памяти в другое. Оригинал сохраняется. В директиве задаются начальный и конечный адреса перемещаемого массива и адрес, начиная с которого необходимо расположить перемещаемый массив. Например, команда M100,1FF,200 скопирует массив по адресам 0100H...01FFFH на адреса 0200H...02FFFH.

Команда H (Hex) — калькулятор. При исполнении этой команды происходит сложение и вычитание двух параметров, причем задается уменьшаемое. Действия выполняются над шестнадцатеричными числами без знака. Результат выводится на экран.

Команда X (eXamine) — статус. Команда X без параметра дает возможность просмотреть содержимое регистров процессора. Пример формата выдачи:

COZOMEOIO

A = FF B = 0000 D = 0000 H = FFFF

S = 0100 P = 0100 MOV A, C

С помощью этой команды можно изменить содержимое регистров, если за ней следует один из двухбайтных регистров B, D, H, S, P, аккумулятор A или биты регистра состояний Z, C, M, E, I. Команда выдает текущее значение регистра (бита) и ожидает ввода нового значения. Для отказа от ввода надо нажать клавишу BK. Например, команда XC позволяет посмотреть и изменить бит переноса.

Команда L (List) — дизассемблер — выводит на дисплей дизассемблированный текст программы. Формат обращения аналогичен команде D. Если второй параметр опущен, то выдается блок из 12 команд. Например, команда L4000 выведет 12 команд, начиная с шага 4000H в формате:

4000 LXI SP, 8000

4003 LHLD 8000

4006 MVI A, 00

4008 OUT 12 и т. д.

Команда A (Assemble) — миниассемблер позволяет изменить часть программы или создать небольшую программу, используя мнемокоды команд процессора. Параметром служит адрес начала модификации или ввода

программы. Отладчик последовательно выводит адреса и ожидает ввода строки. Строка, не являющаяся мнемокодом, игнорируется. Отладчик предлагает повторить ввод. Окончание ввода — по клавише BK, нажатой в первой позиции поля мнемокода или после ввода символа «.» Команда A4000 позволит ввести программу с адреса 4000H.

Команда G (GO) — переход — осуществляет запуск программы пользователя. Первый параметр команды — адрес начала исполняемой программы, второй и третий параметры (если они есть) — адреса контрольных точек. По достижении одной из этих точек управление возвращается отладчику, а на экран выводится адрес останова. Отлаживаемая программа должна оканчиваться командой RST 1, иначе ошибка в выборе точек останова может привести к потере контроля над тестируемой программой и непредсказуемым результатам. Точки останова, назначенные в ПЗУ, не будут отработаны. Если первый параметр команды опущен, то выполнение программы начнется с текущего адреса. Примеры обращения к команде: G,5000 запустит программу с текущего адреса с остановом по адресу 5000H, а команда G4000 — с адреса 4000H. Управление возвращается отладчику, если во время работы программы встретится команда RST 1.

Команды T (Trace) и U (Untrace) — трассировка — обеспечивают пошаговое исполнение программы пользователя. Параметром служит число шагов. Команда без параметра соответствует одному шагу. Единственное отличие команд заключается в том, что трассировка с помощью команды T сопровождается выводом данных в формате команды X после каждого шага, а команда выводит состояние регистров только после останова. Прервать трассировку можно нажатием на любую клавишу. Выполнение программы в режиме трассировки происходит примерно в 500 раз медленнее, чем обычно, поэтому программы, работающие в режиме реального времени, адекватно трассировать нельзя. Попытка трассировки программы, размещенной в ПЗУ, приводит к потере контроля над программой.

Команда T5 исполнит пять шагов тестируемой программы, сопровождая каждый шаг выдачей информации.

Все описанные команды соответствуют утилите DDT [4, 5]. Монитор дополнен рядом команд, расширяющих его возможности.

Команда R (Read) — чтение программы или данных с магнитофона. Параметрами служат адрес загрузки и длина загружаемого модуля. Если длина текущего блока на МЛ меньше указанной в команде, то чтение будет продолжено в следующих блоках до приема нужного числа байтов. Если ввод файла заданной длины оканчивается внутри блока, то оставшаяся информация игнорируется. Например, команда RE000,1F3F загрузит модуль длиной 1F3FH с адреса E000H. Для ПЭВМ «Ириша» это соответствует первой странице видеоОЗУ. Если при считывании возникают ошибки или загрузка прерывается по нажатию любой клавиши, то выводится стандартное сообщение об ошибке — символ «?».

Команда W (Write) — запись программы или данных на магнитофон. Параметры аналогичны команде R. Данные будут записаны в одном блоке на МЛ. Пример записи команды: WE000,1F3F.

Команда I (Input) — ввод. При исполнении этой команды на дисплей выводится байт, принятый из порта ПЭВМ с адресом, заданным параметром команды. Байт выводится в двоичной и шестнадцатеричной коливках. Например, команда I12 отобразит на экране состояние порта I2H в формате:

A8 — шестнадцатеричное значение

76543210 — номера битов в байте

10101000 — побитное представление считанного байта.

Команда O (Output) — вывод информации в порт ввода-вывода. Задается выводимый байт и адрес порта. Команда OА8, 12 запишет байт А8Н в порт 12Н.

Команда P (Pass) — обход. Если в процессе отладки нет необходимости трассировать какую-либо подпрограмму, вызываемую инструкцией CALL, то, используя команду P, можно обойти эту подпрограмму и продолжить трассирование со следующего шага.

Команда часто используется при трассировке для обхода уже отлаженных подпрограмм и подпрограмм, записанных в ПЗУ.

За справками обращаться: 117234, Москва, Ленинские горы, МГУ, химический факультет. Тел. 939-44-33.

ЛИТЕРАТУРА

1. Барышников В. Н., Воронов М. А., Галутиц Ю. В., Романов В. Ю., Рушайло-Арно А. Л. Программное обеспечение ПЭВМ «Ириша»//Микропроцессорные средства и системы.— 1986.— № 3.— С. 59—64.
2. Барышников В. Н., Воронов М. А., Кулаков В. Б., Романов В. Ю. Программа вывода сим-

вольной и графической информации ПЭВМ «Ириша»//Микропроцессорные средства и системы.— 1986.— № 4.— С. 79—89.

3. Барышников В. Н., Бондарь И. Н., Воронов М. А., Романов В. Ю. Программы стартового ПЗУ ПЭВМ «Ириша»//Микропроцессорные средства и системы.— см. наст. номер. С. 56.
4. CP/M Operating System Manual. McGraw Pesigns, 1982.
5. М. Уэйт, Дж. Ангеймер. Операционная система CP/M. М.: Радио и связь, 1986.
6. Барышников В. Н., Быстров В. П., Воронов М. А., Паначев Ф. И., Романов В. Ю. Модуль процессора персональной ЭВМ «Ириша»//Микропроцессорные средства и системы.— 1986.— № 2.— С. 52—62.
7. Романов В. Ю., Барышников В. Н., Воронов М. А., Паначев Ф. И. Графические возможности персональной ЭВМ «Ириша»//Микропроцессорные средства и системы.— 1986.— № 1.— С. 61—72.

Статья поступила 29 апреля 1987 г.

КРАТКОЕ СООБЩЕНИЕ

УДК 681.327.2.01

В. М. Калужный, А. А. Корнилович, Е. И. Уваров, П. Н. Федан

ПРОГРАММНЫЙ КОНТРОЛЛЕР СОПРЯЖЕНИЯ МИКРОЭВМ «ИСКРА 226» И СИСТЕМЫ КАМАК

При комплексной автоматизации научных исследований возникает необходимость разгрузить основную ЭВМ от многочисленных операций с промежуточной информацией и организовать обмен лишь начальными, корректирующими и конечными данными.

Для этой цели разработан интеллектуальный контроллер, работающий в системе управления второго уровня на основе микроЭВМ «Искра 226» и системы КАМАК. В состав контроллера входят интерфейс микроЭВМ «Искра 226», микропроцессор, ОЗУ, ПЗУ и интерфейс магистрали КАМАК. Режим преобразования форматов данных и пересылки массивов данных в микроЭВМ и обратно осуществляется в полном соответствии с протоколом обмена микроЭВМ и магистрали КАМАК. Контроллер выполнен на базе МПК БИС серии К580 и имеет следующие характеристики: быстродействие — 625 тыс. операций/с; длительность машинного такта — 400 нс; объем ПЗУ — 2К байт; объем ОЗУ — 62К байт.

Архитектура интерфейса КАМАК соответствует формуле (N), (A), (F).

Основу аппаратно-программного интерфейса микроЭВМ составляет БИС КР580ВВ55. Аппаратно реализована команда установления связи и запись необходимых состояний на шине состояний в ответ на каждую команду. Процессор контроллера, находясь в рамках ожидания или прерывания, считывает флаг обмена. Если флаг взведен, контроллер переходит к обработке программы анализа и выполнения команды ввода-вывода микроЭВМ. При необходимости контроллер формирует запрос на прерывание микроЭВМ.

Центральный процессор выполнен на БИС КР580ИК80А. Магистраль состоит из 16-разрядной шины адресов, 8-разрядной шины данных и шины управления. В контроллере используется синхронная дисциплина обмена.

ПЗУ К556РТ5 содержит монитор, в состав которого входят программы инициализации работы контроллера, организации циклов обмена по магистрали КАМАК, пересылки массивов из контроллера в микроЭВМ и обратно, инициализации запуска загруженной программы.

ОЗУ выполнено на БИС К565РУ5. Регенерация происходит в момент, когда нет обращения к памяти, и является прозрачной для процессора. Интерфейс магистрали КАМАК собран на четырех микросхемах К580ВВ55.

Тел. для справок: 46-06-55, Новосибирск. Сообщение поступило 21 октября 1986 г.

МАГИСТРАЛИ МП-СИСТЕМ

УДК 681.327

В. В. Кипаренко

СТАНДАРТИЗАЦИЯ ИНТЕРФЕЙСОВ МИКРОПРОЦЕССОРНЫХ СИСТЕМ

Возрастающие требования к производительности и надежности информационно-управляющих систем (ИУС) могут быть обеспечены магистрально-модульными мультипроцессорными системами (ММС), построенными на базе стандартных интерфейсов. Стандартизация интерфейсов на государственном уровне с учетом использования результатов международного сотрудничества облегчит за-

дачу разработчиков ММС. В настоящее время имеются отечественные стандарты в основном отраслевого значения, кроме ГОСТ 26.765.51—86. На международном уровне разработкой нормативно-технической документации на интерфейсы занимаются технические комитеты ТК 47В, ТК 45К, ТК 65С, IЕС (МЭК), Европейский комитет по стандартизации в ядерной электронике ESONE (ЭСОНЕ),

ISO (ИСО) — технический комитет ТК 97. В разработках зарубежных фирм отсутствует единая стандартизация: в 50 типах зарубежных одноплатных микроЭВМ используются свыше 30 различных фирменных и национальных стандартов на интерфейс. Основные сигналы, сравнительные данные и состояние со стандартизацией основных интерфейсов отражает табл. 1... 3.

При составлении таблиц 1... 3 учитывались информационно-логические, диагностические, структурные, временные, программные и физические возможности интерфейсов. Данные об информационно-логических и диагностических возможностях приведены в

табл. 1. Сигналы в этой таблице можно подразделить на следующие основные группы: адреса и данных, прерывания и арбитража, управления (командные, состоящие), диагностики и питания.

Организация передачи адреса и данных. Линии адреса и данные могут быть раздельными и мультиплексируемыми. Мультиплексирование — это использование одной и той же линии в различные моменты времени для передачи функционально различной информации. Мультиплексирование адреса и данных обеспечивает на один адресный цикл передачу нескольких циклов данных. За счет уменьшения скорости обмена данными можно использовать мультиплексируемые линии адреса и данных в режиме управления, прерываний и арбитража [1—4].

Достоинства систем с мультиплексируемыми шинами:

конструкция более надежна и технологична благодаря меньшему числу контактов соединения и разводки внешних цепей модулей;

простота построения отказоустойчивой системы путем дублирования шин;

меньший уровень перекрестных помех;

меньшее число интерфейсных БИС.

К недостаткам систем с мультиплексируемыми шинами следует отметить:

значительное снижение пропускной способности при частой смене адресов (единичные передачи);

ухудшение характеристик немультимплексируемых процессоров;

увеличение числа типов сигналов синхронизации последовательности операций передачи данных и адреса (синхросигналы AS, DS-задатчика и сигналы подтверждения АК, ДК-исполнения);

необходимость в дополнительном оборудовании для организации мультиплексирования.

Раздельные шины адреса и данных применены в одном из новых стандартов VME-BUS. Для систем с немультимплексируемыми шинами адреса (данных) характерны простота интерфейсной схемы и повышенная пропускная способность при одиночных передачах данных. При многопроцессорной работе и большом количестве единичных передач данных по раздельным шинам адреса и данных в системе ориентировочный выигрыш по производительности не должен превышать 30% по сравнению с мультиплексируемыми шинами адреса и данных.

При использовании в системе процессоров различной разрядности (8... 32) число разрядов в передаваемом по магистрали слове адреса и данных задается кодом на линиях управле-

ния размером. Режимы передачи информации во времени подразделяются на асинхронный и синхронный. Для асинхронного обмена данными и адреса (АСОДА) характерны дополнительные управляющие сигналы начала и конца обмена.

Достоинства АСОДА: обеспечивается работа устройств с различным быстродействием; отсутствует общая синхронизация устройств;

осуществляется контроль правильности обмена данными;

обеспечивается возможность модернизации при разработке новых быстродействующих технических средств.

Недостатки АСОДА:

наличие двух дополнительных линий для передачи управляющих сигналов задатчика и исполнителя;

ограниченная пропускная способность системы при передаче данных на большие расстояния.

При синхронном способе взаимный обмен данными между задатчиком и исполнителем обеспечивается общим синхронизирующим сигналом длительности. Синхронный обмен данными (СОД) имеет наивысшую пропускную способность при передаче информации от задатчика к исполнителю.

Недостатки СОД:

отсутствует контроль правильности обмена данными;

нужна единая системная синхронизация всех функциональных устройств;

требуется оценка временных задержек в линиях связи и устройствах;

необходимо устройства с одинаковым быстродействием;

требуется дополнительное оборудование и ПО.

При использовании смешанного обмена данными [5] используется временное разделение подтверждений. Задатчик синхронно передает данные исполнителю и подсчитывает число подтверждений. В конце обмена он сравнивает эту величину с числом циклов переданных данных. В этом случае значительно уменьшаются задержки, характерные для АСОДА.

Для обращения к памяти или устройствам ввода-вывода (в/в) используются раздельные (чтение, запись в память или в/в) или совмещенные (чтение-запись, память в/в) сигналы синхронизации.

Преимущества метода раздельных линий синхронизации устройств в/в заключаются в простоте декодирующих систем и коротких команд и возможности введения дополнительных сигналов управления передачей информации разнообразных периферийных устройств.

Для совмещенных линий синхронизации устройств в/в характерны использование одних команд для работы с данными как в памяти, так и в устройствах в/в, отсутствие отдель-

ной системы для декодирования и управления в/в.

К недостатку метода совмещенных линий следует отнести сложность интерфейсных БИС, которые должны обеспечивать программную генерацию дополнительных сигналов управления при использовании большого числа разнообразных устройств в/в.

Для обращения к значительному числу периферийных устройств с большим объемом памяти совершенствуются способы адресации в новых стандартах. Так, введение географической адресации [2, 3, 5, 6] исключают предварительную адресацию модулей функциональных устройств (ФУ) до установки в блок. Этот способ адресации позволяет уменьшить разновидность запасных частей, инструмента и принадлежностей (ЗИП), ФУ и исключить средства ручного программирования модулей перед установкой в систему. Сочетание географического и логического способов адресации позволяет обеспечивать программируемую конфигурацию системы.

Арбитраж и прерывания. В разработках новых стандартов на интерфейс уделяется много внимания совершенствованию средств арбитража и прерывания с целью уменьшения времени реакции на установление связи между устройствами.

Схема арбитража описывается следующими параметрами:

число сигнальных линий;
число обрабатываемых запросов;
алгоритмы выбора приоритета;
тип синхронизации (таймирования);
централизованная или распределенная схема арбитража.

Система централизованной схемой арбитража имеет низкую надежность и требует отдельных линий запроса и разрешения для каждого запрашивающего агента. Распределенная схема арбитража [1—3, 6] не чувствительна к повреждениям в отдельной точке и гибка в системном распределении и присвоении уровней приоритетов агентам. Но при этом необходимо организовать системную синхронизацию с учетом временных задержек в схеме арбитража. Система прерываний представляется линиями заявки на прерывание (одна или несколько линий) и механизмом подтверждения.

В новых проектах стандартов [3, 5, 6] на интерфейс разработчики не используют последовательную схему приоритетов из-за низкой скорости выбора агента (значительная суммарная задержка последовательной цепи в большой системе); зависимости приоритетности от расположения агентов в системе; низкой надежности последовательной цепи соединений всех агентов; большого времени при программном обслуживании большого числа внешних устройств (порядка

Перечень сигналов в фирменных и государственных

Сигнал	FASTBUS	Z-BUS	VERSABUS	VME-BUS	K896-BUS
Адрес и данные — мультиплексные — раздельные	AD (31...0)	AD (31...0)	A (31...0) D (31...0)	A (31...0) D (31...0)	AD (31...0)
Управление размером		B/W, W/LW W/LW	DSO, DSI	L.AS, UAS L.WORD, LDS	CX CX
Синхронизация	AS DS	AS DS	AS	DSO, DSI, UDS AS	AS DS
Подтверждение передачи	AK DK RD	STX STX R/W	DTACK WRITE	ACK DTACK WRITE	AK
Управление вводом-выводом и памятью				R/W AMX AMX AMX AMX AMX	CX CX CX CX CX
Географическая адресация	MSX MSX MSX MSX	N/S STX STX STX STX	AMX AMX AMX AMX AMX	AMX AMX AMX AMX AMX	
Состояние системы	GA (4...0) EG	WT BN	SIX WAIT STOP		
Сброс	RB	RESET	SYSRESET BCLR SECRESET	SYSRESET SMRQ	R
Режим ПДП				SMACK	
Режим прерываний	SR	INT (3...1) IEI (3...1)	IRQ (6...0) BGI (4...0)	IRQ (6...0) BGI (3...0)	BRQ
Тип прерываний		IEO (3...1) SECLEN BAI, CAI MMAI BAO, CAO MMAO	BGO (4...0) ACKIN ACK OUT	BGO (3...0) LACK, ACKIN ACK OUT LACK	
Режим арбитража	AL (5...0) AR/GK AG AI, SSX	MMREQ BUSREQ CPVREQ CAVAIL STX	BR (4...0) BBSY	BR (3...0) BBSY	BRS (2...0) IPRQ SC BB
Работа модулей		STX STX			
Диагностика по	PE, SSX SSX	PE SYSPALL	BERR SYSPAIL	DERR BERR	
Четность	PA	P (3...0)	DPARITY (3...0) APARITY (1...0)		
Прерывания из-за неисправности электропитания		PWRBAD	ACFAIL PDCWN BREL	ACFAIL	PEL POI
Резервный источник питания			STDBY	STDBY	
Регенерация динамической памяти					
Запрет обращения к ОЗУ					
Синхронизация		BCLK MCLK	SYSCLK ACCLK	SYSCLK	BCLK
Последовательная магистраль	RX, TX			SERCLK SERDAT	

Примечание. Обозначение и назначение сигналов в фирменных и государственных стандартах на интерфейс для MMMC при

FUTURF-BUS	NU-BUS	MULTUBUS-II	MULTIBUS-I	AMS-BUS	BUSI	EUROPARBUS	SMP-BUS	PCXT-BUS
AD (31...0)	AD (31...0)	AD (31...0)	A (23...0) D (15...0) ALE	A (23...0) D (15...0) WORD, BYTE	A (23...0) D (15...0)	A (19...0) D (15...0)	A (15...0) D (7...0) ALE	A (19...0) D (7...0) ALE
CMX CMX	TMX TMX	SCX	BHEN	BHEN	BHEN	BHEN		
AS DS		SCX		AS DS				
AK, AI DK, DI	ACK	SCX SCX	XACK	XACK	XACK	AREADY	RDYIN	
		SCX SCX		R/W M/I0		WE IOM	BBS	AEN SLCTD
CMX CMX CMX CMX	TMX TMX TMX TMX	SCX SCX SCX SCX	IORC MRDC IOWC MWTC	IORC MRDC IOWC MWTC	IORC MRDC IOWC MWTC	RD RD WR WR	IOR MEMR IOW MEMW	IOR MEMR IOW MEMW
GA (4...0)	ID (2...0)	ID (2...0)				SA (3...0)		
		SCX TIMOUT	WAIT HALT	BTMO HLT		READY	READY	I/O CHROY
RE	RE START	RST RSTNC	INIT	RESET INIT	INIT	RESET	RESET RESIN	RESET
				DDS			DREQ 2...0 DME STB DACK	DRQ (3...1) DABK (3...0) IRQ (7...2)
		INT LACH (n)	INT (7...0) BPRN	LXACK DXACK INT (7...0) BPRN	INT (7...0) BPRN	INT (7...0) EGTI	INT	
			BPRO	BPRO	BPRO	BGTO INTREXP	BVSEN	T/C
			INTA	INTA	INTAE	INTA		
AN (6...0) AR, AC	ARB (3...0) RQST	ARB (5...0) BREQ	CBRQ BREQ	BREQ	CBRQ BREQ	BREQ		
AQ AP, STX		SCX SCX	BUSY	BUSY	BUSY	BUSY		
	TMX	LOCKSCX	LOCK	LOCK	LOCK			
EV STX	SPV	DERR AERR BUSERR						
ED (3...0)	SP	DPAR (3...0) APAR	PAR1, 2	BPAR (1...0)				
CR		SCM						
SP		DCLON PROT	PFIN, ACLO PFSN MPRO	PFIN PFSN MPRO		PFI	MPRO	
		BAT +5B	PESR AUX RE	AUX RE BATT +5B			BATT +5B	
SBO, SBI	CLK	BCLK CCLK SDA, SDB	INH1 INH2 BCLK CCLK	INH1 INH2 BCLK CCLK	INH1 INH2 BCLK CCLK SERL	INHIBO INHIB1 BUSCL	CLK	CLOCK I/O CLCK

ведены в приложении.

Сигнал		ИИ1	МПМ	ИКИ	MICROBUS	ECB-BUS	
Адрес и данные	— мультиплексные — раздельные	A (29...0) D (15...0)	A (20...0) D (7...0)	A (15...0) D (7...0)	A (15...0) D (7...0)	A (15...0) D (7...0)	
Управление размером	— адреса — данных	BHEN	Чт8, Чт16				
Синхронизация	— адреса — данных						
Подтверждение передачи	— адреса — данных	XACK	ПРП МОДЗУ	ОТВ		BUSAK	
Управление вводом-выводом и памятью	— совмещенное — раздельное	— чтение-запись				DESELECT	
		— в/в/память					
		— чтение в/в	IORC	ЧТЭ	ПРМ	IOR. RD	IORQ. FD
		— чтение из памяти	MRDC	ЧТЗУ	ЧТМ	MEMR. RD	MREQ. RD
		— запись в в/в	IOWC	ЗПЭ	ВОЧ	IOW. WR, EWR	IORQ. WR
		— запись в память	MWTC	ЗПЗУ	ЗАП	MEMW. WR, EWR	MREQ. WR
Географическая адресация	— готовность — ожидание — останов	READY WAIT HALT			READY	WAIT HALT	
Сброс	— системный — выборочный	INIT	Сброс	УСТ	RESET	RESET	
Режим ПДП	— запрос				DRQ		
	— разрешение — подтверждение	по	— входу — выходу		DACK	DMARDY	
Режим прерываний	— запрос	INT (7...0)	PRE (63...0)	ЗПР	INTR	INT	
	— разрешение по	BPRN BPRO	РПШ — Вх РПШ — Вых			IEI IEQ	
	— подтверждение по	INTA	ПРП			BAI BAO	
Тип прерывания	— немаскируемый — маскируемый					NMI	
Режим арбитража	— уровни				3DM (7...1)		
	— запрос — подтверждение	BREQ	ЗПШ		PDM (7...1)	BUSRO	
	— разрешение — занято	BUSY	ЗАМ				
Работа модулей	— однопортовая — двухпортовая	LOCK					
Диагностика по	— ошибки						
	— данных						
	— адреса						
	— отказу системы — отказу сетевого электропитания — отказу источника постоянного тока						
Четность	— данных — адреса — команд	PAR1. 2					
Прерывания из-за неисправности электропитания	— переменного тока — постоянного тока — защита устройств — сброс	DEIN ACLO PESN MPRO PESR AUXRE				PWRCL	
Резервный источник питания		+5 В Б +12 В Б				MI RFSH	
Регенерация динамической памяти	— синхронизация — разрешение — запрос						
Запрет обращения к	— ОЗУ — ПЗУ и ППЗУ	INH1 INH2 BCLK CCLK	ЗАПР1 ЗАПР2 ТЧШ ОФЧ		TC		
Синхронизация	— магистрали — отдельных компонентов						
Последовательная магистраль							

VAMOS-80	MPST	CIM-BUS	S100-BUS	G-64	G-96	STD-BUS1	STD-BUS2	STE-BUS
A (19...0) D (7...0)	A (15...0) D (15...0)	A (15...0) D (7...0)	A (15...0) D (15...0)	A (15...0) D (15...0)	A (23...0) D (15...0)	A (15...0) D (7...0)	A (19...0) D (7...0)	A (15...0) A (7...0)
			DBIN	DSO, DSI	DSO DSI	IOEXP MEMEX		
BUSAK		BUSAK		RDY/DTACK	RDY/DTACK	BUSAK	ADRSTB DAT STB	BUSAK
IORQ, RD	R, IOM	IO, RD	ICR	R/W VPA R/W, VED	R/W VPA R/W, VED		XFRACK	
MREQ, RD IORQ, WR	R, IOM WO, W, IOM	MEM. WR IO, WR	MEMR IOW, WO	R/W, VED R/W, VMA	R/W, VED R/W, VMA	RD, IORQ RD, MEMRQ	SIORQ SWRITE	RD, IORQ RD, MEMRQ
MREQ, WR	W, IOM	MEM, WR	WR, MEM, WR	R/W, VMA	R/W, VMA	WR, IORQ WR, MEMRQ	SWRITE SIORQ	WR, IORQ
	RDY		READY	RDY/DTACK	RDY/DTACK		SWRITE	WR, MEMRQ
WAIT	ENF	WAIT				WAIT RQ		
RESET	RS	RESET	RESET	HALT RESET	HALT RESET	SYSRESET	RESET	SYSRESET
						PBRESET		
	DMARQ		DMA 3...0 HOLD					
	DMAACK		HLDA, HLTA	HALT ACK				
INT (1...0)		INT (7...0)	INT (7...0)	IRQ FIRQ	RQ (5...1)	INTRQ	INTRO (3...0) IACKIN IACKOT	INTR (3...1)
	ACKI ACKO			CHI CHO	CHI CHO	PCI PCO		PCI PCO INTAK
		IVTA	INTA	IACK	IACK	INTAK	INTAK	INTAK
NMI			NMI	NMI	NMI	NMI RQ	BACK IN BACK OUT	ACI ACO
BUSRQ	SRQ	BUSRQ		BUSREQ BCACK	BRQ BCACK	BUSRQ	BUSREQ	
	RBB BB			BGRT ENABLE	BGRT BGACK BBUSY		BUSBSY	WAITRQ
			ERROR	PE	PE		BUS ERR	
					SYSFAIL			
			DODSF ADCB CDSB SDSB					
	PFD	PFAIL	PWRF		PWP			
		PWRS	POC					
	BAT	BAT		BAT	BAT	AUX +12	AUX	AUX
		RESH					REFRESH	REFRESH
		INH1						
	CYNC T2	INH2 TMRES	SYNC CLOCK	SYCLK	SYCLK ARBCLK	MCSYNC CLOCK	SYSCLK	MCSYNC CLOCK

Сигнал		EXORCISER	APPLE-BUS	AGAT	TM990	
Адрес и данные	— мультиплексные — раздельные	A (15...0) D (7...0)	A (15...0) D (7...0)	A (15...0) D (7...0)	A (15...0) D (15...0)	
Управление размером	— адреса — данных			I/OSTER	ALATCH	
Синхронизация	— адреса — данных					
Подтвержденные передачи	— адреса — данных					
Управление вводом-выводом и памятью	— совмещенное	— чтение-запись в/в/память	R/W VUA	R/W IOSEL	R/W DEVSEL	BDIN
		— чтение в/в	R/W, VUA	R/W, IOSEL	R/W, IOSEL	BDIN
	— раздельное	— чтение из памяти	R/W, VMA	R/W	R/W	BDIN
		— запись в в/в	R/W, VUA	R/W, IOSEL	R/W, IOSEL	WE, DBIN
		— запись в память	R/W, VMA	R/W	R/W	WE, DBIN
Географическая адресация						
Состояние системы	— готовность	BA	RDY	RDY	READY	
	— ожидание — останов	G/H				
Сброс	— системный	RESET	RES	RES	PRESET	
	— выборочный — запрос	VxA	DMA	DMA	RESTART IORST	
Режим ПДП	— разрешение по входу				HOLD	
	— подтверждение запроса по выходу	MEMRDY IRQ	DMAIN DMAOUT	DMAIN DMOUT	HOLDA INT (15...1) GRI GRO	
Режим прерываний	— разрешение по входу		IRQ INT IN INT OUT	IRQ INT IN INT OUT		
	— подтверждение по выходу				IAO	
Тип прерываний	— немаскируемый	NMI	NMI	NMI		
	— маскируемый					
Режим арбитража	— уровни — запрос — подтверждение — разрешение				XA (3...0)	
	— занято				BUSY	
Работа модулей	— однопортовая					
	— двухпортовая	PE				
Диагностика по	— ошибки	— данных				
		— адреса				
	— отказу системы	DEBVG				
	— отказу сетевого электропитания — отказу источника постоянного тока					
Четность	— данных					
	— адреса — команд					
Прерывания из-за неисправности электропитания	— переменного тока	PWRP				
	— постоянного тока					
	— защита устройств					
	— сброс					
Резервный источник питания	STANDBY					
Регенерация динамической памяти	— синхронизация	MEMCLK			MEMCLK	
	— разрешение	REFGR			REFCLK	
	— запрос	REFREQ				
Запрет обращения к	— ОЗУ			INH	MEMEN	
	— ПЗУ и ППЗУ					
Синхронизация	— магистрали	CLOCK		FRI, FR2 2MHZ	BUSCLK	
	— отдельных компонентов				CPUCLK	
Последовательная магистраль						

E-BUS	EUROBUS	E3S	ИУС	ОШ	Q-BUS	ERIGONEBUS	МПИ	Q-BUS-22
ADI 16/16/7	AD18/16	AD24/16	AD (15...0)	A (17...0) D (15...0)	AD (15...0)	ADI 24/16/6	AD18/16	AD (15...0) A (21...16)
MEWID	Byt Ad Byt WK	Byt Ad Byt WK		CX CX	WTBT	LBS, HBS	ПЗП	ПЗП
			ППА HD	SSYN	REPLY	DACK	ОТВ	ОТВ
			ПВУ	MSYN	BS, BDIN			ВУ
DEN			ЗД, ПВУ	CX	BDIN	Read, Lbs AVE, HBS	ДТЧ, ВУ	ДЧТ, ВУ
DEN			ЗД, ПВУ	CX	BDIN	Read, HBS AVE, HBS	ДЧТ	ДЧТ
WE, DEN			ПРА	CX	BDOUT	DACK, LBS HBS	ДЗП, ВУ	ДЗП, ВУ
WE			ПРА	CX	BDOUT	DACK, LBS HBS	ДЗП	ДЗП
READY	CcAb	CcAb						
PRESET	Rs	Rs	СБР БСВ	INIT	INIT	RESET	ОСТ	ОСТ
IORST			БСВ					
				NRP NPG	DMB DMGI DMGO	INTIN		
GRI GRO	Rq (n) BUSG2 CcBn	Rq (n) BUSG2 CcBn	ПР (7...0)	SACK BR (7...4) BG (7...4)	SACK IRQ	INT OUT DACK INT (6...1) BG1 BGO	ЗПР (7...4) ПРР (7...4)	ЗПР (7...4) ПРРП ПРРИ
INTEN	CcFln CcRes	CcFln CcRes			IACK IAKO		ПЗ	ПЗ
NMI					EVNT		ПВС	ПВС
XA (3...0)	Jt BUSACQ	Jt BUSACQ	РАЗвх РАЗвых			LMS BREQ	ЗМ (1...0) РЗМ (1...0)	ЗМ ПЗ РЗМ, РЗМИ
BUSY	BUSdeal	BUSdeal	ЗАН	BBSY		BBSY		
				РА, РВ				
PWRF		PF		ACLO DCLO	POK DCOK	ACFAIN DCFAIL	АСП АИП	АСП АИП
BAT					REF		РГП	+5В. Б +12В. Б РГП
MEMEN								
BUSCLK C				MSYN	SYNC		ОБМ	ОБМ

Сигнал		FASTBUS	Z-BUS	VERSABUS	VME-BUS	VME-BUS	K816-BUS	FUTUREBUS	NU-BUS	MULTIBUS	MULTIBUS-1	AMS-BUS
Адрес и данные	— мультиплексные — раздельные	+	+	+	л	см	+	+	+	л	см	+
					+	+				+	+	
Управление размером	— данных — адреса		+	+	+	+	м	м	м	м	м	+
					+	+	м	м	м	м	м	+
Синхронизация	— данных — адреса	+	+	+	+	+	+	+		м		+
					+	+	+	+				
Подтверждение	— данных — адреса	+	м	+	+	+		+		+	м	+
					+	м						
Управление вводом-выводом и памятью	— совмещенное — раздельное	+	+	+	+	+	м	м	м	м	м	+
					л	л	л	л	л	л		
					л	л	л	л	л	л		
					л	л	л	л	л	л		
Географическая адресация		+						+	+	+		
								+	+	+		
Состояние системы	— готовность — ожидание — останов	+	м	+						+	м	+
			+								+	+
Сброс	— системный — выборочный	+	+	+	+	+	+	+	+	+	+	+
Режим ПДП	— запрос — разрешение по — подтверждение									+		+
Режим прерываний	— запрос — разрешение по — подтверждение по	+	+	+	8	+				+	8	9
Тип прерываний	— маскируемый — немаскируемый	+	+	+	+	+	+	+	+	+	+	+
Режим арбитража	— уровни — запрос — разрешение — занято	6	+	5	4	3	7	4	4	2	6	+
Работа модулей	— однопортовая — двухпортовая	+	+	+	+	+	+	+	+	+	+	+
Диагностика по	— ошибке — отказу системы — отказу сетевого питания — источника постоянного тока	+	+	+	+	+	+	+	+	+	+	+
Четность	— данных — адреса — команд	+	4	4	2			4		+	4	м
Последовательная связь между	— блоками — модулями	+			+			+		+		

Примечание. л — локальная магистраль; см — системная магистраль; м — модификация режимов кодируемыми линиями.

50 мкс для Q-BUS); невозможности замены модулей в работающей системе.

Оригинальное решение по приоритетам предложено в проектах стандартов [1, 3], где приоритетность осуществляется через адресные пространства сообщений. Организация приоритетов подробно описана в [3].

Системные линии прямого доступа к памяти используются в основном в стандартах, предназначенных для разработки однопроцессорных систем.

Шина управления. Для новых стандартов [2, 3, 5—7] характерно кодирование функциональных линий (линии модификации) вместо многочисленных общесистемных отдельных линий управления вводом-выводом, состоянием и установкой системы в исходное положение и др.

Внутрисистемная диагностика. При создании стандарта на интерфейс для МММС уделяется особое внимание программным и аппаратным средств-

вам диагностики функциональных узлов и системы в целом. Средства диагностики позволяют определить работоспособность (исправность, правильность функционирования), найти и устранить неисправности в системе. Это достигается аппаратным и программным контролем:

- на четность данных, адреса и команд [3, 8];
- постоянного напряжения вторичных источников питания и переменного напряжения сети [9—11];
- времени выполнения операций на шине [6, 10];
- готовности данных и устройств к связи [8—10];
- всех операций на шине [6, 12, 13];
- отсутствующих или лишних модулей в заданной конфигурации системы [1—3, 6].

Для поиска и устранения неисправностей в системе предусматривают-

ся: сервисный процессор и регистры

для поиска неисправностей и реконфигурации системы в случае устойчивого отказа устройства [6, 12, 13]; средства защиты информации при отказах сетевого напряжения питания [9, 16];

средства для коррекции ошибок при передаче информации;

тест-программы для поиска неисправностей и программы для реконфигурации системы в случае устойчивого отказа устройств;

дистанционные средства диагностики [6, 12, 13];

линии готовности данных и ожидания (при остановке цикла) [8—10].

Структурное и конструктивное разнообразие стандартов затрудняет создание единых средств диагностирования. Дальнейшее совершенствование средств диагностирования будет развиваться в направлении увеличения числа линий контроля и коррекции ошибок при передаче информации адресов и команд, а также возраста-

Параметр		FASTBUS	Z-BUS	VERSABUS	VME-BUS	K890-BUS	FUTUREBUS
Способ передачи адреса и данных	<ul style="list-style-type: none"> - мультиплексируемый - раздельный 	+	+	+	+	+	+
Управление разрядностью данных	<32		+	+	+	+	+
Способы адресации	<ul style="list-style-type: none"> - географическая - логическая - многоступенчатая и групповая 	+	+	+	+	+	+
Линии модификации режимов		3	5	8	6	3	3
Схемы арбитража	<ul style="list-style-type: none"> - централизованная - распределенная 	T+	+	+	+	+	+
Протокол передачи	<ul style="list-style-type: none"> - асинхронный - синхронный 	+	+	+	+	+	+
Скорость передачи по см при режиме Мбайт/с)	<ul style="list-style-type: none"> - одиночном - последовательном 	35 70		24	32 57		20 118
Наличие интерфейсных БИС		+			+		
Сигналы ошибок по	<ul style="list-style-type: none"> - отказу сетевого питания - отказу источника постоянного тока - отказу системы - паритету данных - не принятой посылке сообщению 		+	+	+	+	
Сигнальных линий		83	73	91	82	49	71
Линий источников питания, земли		37	21	38	14	14	22
Напряжение источников	<ul style="list-style-type: none"> - 5В - 5В - 5В (батареиное) - ±12 В 	+	+	+	+	+	+
Число контактов соединителей		130	96	140	96/96	64	96
Размер печатной платы модуля		PE1	E1, E3	RP1	E1, E3	E1, E3	PE2

Примечание. Размеры европлаты: PE1—366,7×403,3 мм, PE2—366,7×280 мм и другие меньшего размера, PE3—366,7×280 мм, RP3—160×235 мм, RP4—130×254 мм, RP5—114,3×155,1 мм, RP6—363×216 мм и 114,3×77,4 мм, RP7—330×260 мм и 250×125 мм, RP8—135× ража централизованное.

ния интеллекта сервисных процессоров по поиску и устранению неисправностей.

Комплексная унификация аппаратных и программных средств. Единство аппаратного, программного и диагностического обеспечения для простых и сложных комплексов ММС предъявляет к стандарту на интерфейс противоречивые требования, которым он должен удовлетворять для малых и сложных систем. Так, для систем с минимальной аппаратной избыточностью необходимы:

малая разрядность адреса и данных;

небольшое число линий арбитража и прерываний;

упрощенная структура соединений.

Для сложных высокопроизводительных систем требуется обеспечить работу:

устройств с широким диапазоном скорости действия (10...100 Мбайт/с) и

разрядности адреса и данных (8...32);

неограниченного числа каркасов, большого числа одновременно функционирующих процессоров с надежной синхронизацией их взаимодействий;

многих процессоров с малым временем разрешения конфликтной ситуации в системе,

многоуровневых иерархических структур;

полностью связанных сетевых систем;

наращиваемых (расширяемых) систем по горизонтали и вертикали с гарантией «прозрачности» программирования всех ФУ. Прозрачность программирования — это однотипность программирования управления передачей данных между устройствами всех иерархических уровней системы с гибким программным изменением конфигурации системы (без привязки к физической структуре системы).

Для упрощения разработки ПО необходима унификация системных вза-

имосвязей на уровне регламентации регистров, содержащих адреса, команды, стеки данных, приоритета и запросов на прерывание [6].

При создании интерфейса для систем с широкими децентрализованными возможностями предусматривается многошинная структура [3]: параллельный интерфейс для быстрого доступа к общим ресурсам и передачи данных между устройствами и последовательный интерфейс для передачи информации как внутри блока, так и на большие расстояния. Системную параллельную шину целесообразно использовать для многоблочной организации системы на расстоянии не более 50 м [6]. Но при этом необходимо обеспечить прозрачность программирования и многопроцессорную работу всех устройств как на параллельной, так и на последовательной шинах.

При разработке стандарта на интерфейс для сокращения обращений к

Таблица 3

NU-BUS	MULTIBUS-II	MULTIBUS-I	AMS-BUS	BUSI	DUOPARBUS	SMPM-BUS	ИЧ1	МПМ	ИКИ	MICROBUS	ECB	VAMOS-80	MPST	CIM-BUS
+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
2	10													
T+	T+	+	+	+	+	+	+	+	+	+	+	+	+	+
+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
20 38	20 40	10						2,5 5						
+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
		+	+	+	+	+	+	+	+	+	+	+	+	+
49	64	65	109	66	64	53	73	53	61	1	46		49	59
46	29	20	56	20	9	10	59				7		11	12
+	+	+	+	+	+	+	+	+	+	+	±15		+	+
+	+	+	+	+	+	+	+	+	+	+	+		+	+
+	+	+	+	+	+	+	+	+	+	+	+		±15	±15
96	96	86/60	9 6/96	96	96	96	96	96			64	96	64	64
PE3	E2, E4	PP2	E3	E1, E3	E3	E1, E3	E2		PP3		E4	E1	E3	E1

E1—100×160 мм, E2—100×220 мм, E3—233,35×160 мм, E4—233,35×220 мм; Размер платы: РП1—368×235 мм и 203×235 мм, РП2—304,8×171,4 мм и 240 мм, РП9—214,3×263,4 мм и 214,3×147 мм, РП10—280×240 мм и 135×240 мм, РП11—98×180 мм и 98×240 мм. Т — тактирование арбитра

Параметр		S100-BUS	G-64	G-96	STD-BUS	STD-BUS	STE-BUS
Способ передачи адреса и данных	— мультиплексированный — раздельный	+	+	+	+	+	+
Управление разрядностью данных	≤32	+	+	+			
Способы адресации	— географическая — логическая — многоступенчатая и групповая	+	+	+	+	+	+
Линии модификации режимов							
Схемы арбитража	— централизованная — распределенная		+	+			+
Протокол передачи	— асинхронный — синхронный	+				+	+
Скорость передачи по см при режиме (Мбайт/с)	— одиночном — последовательном	12					
Наличие интерфейсных БИС							
Сигналы ошибок по	— отказу сетевого питания — отказу источника постоянного тока — отказу системы — паритету данных — не принятой посылке сообщению	+		+	+		
Сигнальных линий		89	1	1	45	47	50
Линий источников питания, земля		9			10	16	14
Напряжение источников	— 5В	+8			+	+	+
	— 5В				+	+	
	— 5В (батареиное)	±16			+	+	
	— ±12В				+	+	+
Число контактов соединителей		100	64	96	56	64	64
Размер печатной платы модуля		PT4	E1	E1	PT5	E1, E3	E1

системной шине применяют дополнительные локальные шины [3]. Устройства на локальной шине сопровождаются необходимым объемом памяти («твердого» ПО) для локальной операционной системы, прикладных программ и программ, обеспечивающих работоспособность при реконфигурации системы в случае отказа устройства. При реализации прерываний на локальной шине обеспечивается наиболее быстрая реакция в реальном масштабе времени. В некоторых проектах стандартов [3, 5] нормируется шина ввода-вывода для расширения и подключения широкого набора периферийных устройств. Разработка стандарта на интерфейс со многими шинами без соответствующей унификации статусных регистров и протоколов обмена данными может значительно усложнить ПО.

Элементная база и конструктивы. Стандарт должен обеспечивать применение развивающихся МП комплектов БИС (по взаимодействию и разрядности) совместно с интерфейсными БИС [3, 6, 7, 14, 15].

Из рассмотренных стандартов интерфейсными БИС обеспечены широко

применяемые системы на базе EUROBUS, FASTBUS, MULTIBUS, VME-BUS. Доступность использования различных МП будет определяться минимальной дополнительной аппаратной частью к интерфейсным БИС и прозрачностью (однотипностью) программирования ФУ, работающих в реальном масштабе времени на всех шинах. Процессорная независимость интерфейса будет определяться сложностью логики интерфейсных БИС, которая должна обеспечивать следующие основные функции: приведение к единому формату данных и команд;

логическое и временное согласование алгоритмов обмена;

согласование электрических сигналов по уровню, форме и нагрузочной способности.

При разработке интерфейса для ММС необходимо ориентироваться на конструктивы, отвечающие современным требованиям международной стандартизации. Опыт применения стандарта на интерфейс типа MULTIBUS в нашей стране показывает, что различия в конструктивах и разводке линий шины не обеспечива-

ют аппаратной и программной совместимости. Зарубежные фирмы в последнее время в основном используют конструктивы Евромеханики. Из 40 фирменных и государственных проектов и стандартов на интерфейс 60% реализованы на конструктивах Евромеханики по МЭК 297-3 (платы размером E1, E2, E3, E4) с накладными разъемами C96 по МЭК 603-2. Однофункциональные модули выполняются на печатных платах малых габаритов (E1, E4), имеют низкую стоимость, упрощенную интерфейсную логику и синхронизацию тех процессоров, которые первоначально использовались [16, 17]. Но малые размеры печатной платы, как правило, требуют многослойного монтажа соединений, а применение большого числа однофункциональных модулей снижает надежность системы из-за большого числа соединений между устройствами. Применение в новых стандартах накладных разъемов C96 по МЭК 603-2 позволяет обеспечить:

легкость замены разъема в случае повреждения;

независимость качества контактирования от толщины платы;

EXORCISER	Apple-BUS	AGAT	TM-990	E-BUS	EUROBUS	E3S	ИУС	ОШ	Q-BUS	Erigone-BUS	МПИ	Q-BUS-22	PCXT-BUS
+	+	+	+	+	+	+	+	+	+	+	+	+	+
+	+	+	+	+	+	+	+	+	+	+	+	+	+
	+	+		+	+	+			+	+	+	+	+
+				+				+	+	+		+	
45	39	44	69	39	30	40	37	61	36	46	44	48	54
28	2	9		19	10	13	29	18		15		12	8
+	+	+		+	+	+	+	+	+	+		+	+
+	+	+		+			+		+	+		+	+
86	49	60		64	64	64	96	48	72	64		99	62
	РП5	РП17		Е1	Е1, 2, 3, 4	Е2, Е4	Е2	РП8	РП9	Е3	РП10	Е2	РП11

совместимость модулей различных изготовителей;

установку на двусторонние и многослойные печатные платы.

Применение разъемов прямого контактирования в ранее разработанных стандартах (например, МПИ по ОСТ 11 305.903-80) предъявляет повышенные требования к технологии изготовления и применения модулей из-за необходимости строгого соблюдения толщины платы, обработки концевых печатных контактов на плате, полной замкнутости модуля в случае выхода из строя печатного контакта, ограниченного числа контактных позиций на плате, отсутствия защиты от пыли контактируемых поверхностей разъема, большой зоны гальваникопокрытия.

Уровни сигналов в интерфейсной части в основном соответствуют элементам с технологией TTL, за исключением системы FASTBUS [6], где применены элементы, изготовленные по ЭСЛ-технологии и SIM-BUS — по КМОП-технологии. Имеется тенденция к использованию в системе единого источника питания с постоянным напряжением 5В.

Для разработки единого проекта международного стандарта некоторые фирмы объединяются (табл. 4). Так, фирмы Motorola, (Exorziser, Versabus), Mostek (STD-BUS1, 2), Thomson (G-64), Signetics (I2S-BUS) объединились в разработке перспективного проекта стандарта VME-BUS, который в настоящее время является стандартом МЭК IEC821.

Фирмы Intel (MULTIBUS, IEEE 796), Siemens (SMP-BUS, AMS-BUS) разработали проект стандарта BUS1, который является документом секретариата подкомитета 47В МЭК.

Фирмы Intel (MULTIBUS, IEEE 796), Zilog (Z-BUS), Siemens (SMP-BUS, AMS-BUS), Hewlett Packard (Proprietary, MP-IB, МЭК 625-1), National Semiconductor (Microbus, SIM-BUS) участвуют в разработке технических требований к проекту стандарта MULTIBUS II.

В перспективных проектах стандартов VME-BUS, P896, MULTIBUS II используются многие решения, впервые принятые в стандарте FASTBUS:

распределенные средства арбитража;

последовательная шина данных; кодированные функциональные линии управления;

32-разрядная мультиплексируемая шина данных и адреса;

сочетание географической и логической адресации;

групповая адресация;

унификация статусных регистров.

Но организация параллельной независимой работы сегментов (магистраль в крейте) и обеспечение взаимосвязи сегментов по параллельной магистральной при одном протоколе обмена, реализованном в FASTBUS, не использованы ни в одном из перспективных стандартов.

Из рассмотренной информации по стандартам (табл. 1, 2) можно отметить влияние основных технических решений, принятых в MULTIBUS (IEEE-796) и FASTBUS (IEEE-960), на реализацию зарубежных фирменных и государственных проектов стандартов.

Сложность развиваемых архитектур МММС, высокие требования к производительности и надежности,

Разработчики стандартов на интерфейсе

Фирма, министерство	Страна	Наименование фирменного стандарта	Состояние со стандартизацией	МП комплект	Литература
Applied computer Gildemeister DEC	США ФРГ США	APPLE-BUS MPST Q-BUS ERIGONE-BUS UNIBUS		1, 3 1, 3, 4 21, 24 21, 24	18 19 20 21
Ferranti	Великобритания	EUROBUS (MODBUS) E3S G64 GS6	DSMP/7232, ISO6951 Проект ESONE	2, 3, 10, 11, 13 2—5, 10, 11, 13—15 17, 19, 21—23	22—26, 15 22, 23, 27—29 30, 31 30, 31
Gespas IEEE Intel	Швейцария США США	FUTUREBUS MULTIBUS I BUS I MULTIBUS II ECB-BUS VAMOS-80 K-896-BUS	IEEEP-896.6.2 IEEE-796 Проект IEC47B Проект IEEEP1296	1, 3—8, 10—14 16—20, 24, 26 6, 8, 14 1, 2, 3, 16 1, 2, 3, 16 26	1, 5, 32, 22, 23 22, 23, 32 33, 34 3, 35—40 41 42 43
Kontron	ФРГ	EUROPAR-BUS OIL HUC H41 IKI S-100		1, 3, 4	44 45 46
Mania МПСА и СУ	ФРГ СССР	ОИШ HUC H41 IKI S-100	ОСТ25795-78 ОСТ25904-80 ОСТ25969-82 ОСТ25984-82	1, 4	10, 22, 23, 47 48
MITS Mosiek	США Бельгия	STD-BUS I VME-BUS VERSABUS VME-BUS EXORRISER II BUS-II	IEEEP961 IEEEP1014 IEEEP970 IEEEP1014 IEC821	1—6, 8, 10, 13, 14, 17	5, 49, 50
Motorola	США	МПИ, АГАТ Q-BUS-22 MICROBUS CIM-BUS FASTBUS	ОСТ11805.903—80 ГОСТ26.765.51—86 IEEEP696.3	16	32, 51
МЭП	СССР	МПИ, АГАТ Q-BUS-22 MICROBUS CIM-BUS FASTBUS	ОСТ11805.903—80 ГОСТ26.765.51—86 IEEEP696.3	30, 31, 32, 33, 21 30, 31, 32, 33, 21	57 11
National Semiconductor NIM	США США	STD-BUS I VME-BUS VERSABUS VME-BUS EXORRISER II BUS-II	IEEEP960 ESONE/FB/01 IEC45 (CB) 186	14	6, 23, 23, 27, 60—62 49, 50
Pro-Log	США	STD-BUS STE-BUS	IEEEP961	1, 3, 5, 9—11, 13, 14, 16 2, 12, 13, 16 2, 12, 13, 16	17, 63, 64 17, 64
МПП Siemens	СССР ФРГ	МПП SMP-BUS AMS-BUS BUS I VME-BUS E-BUS TM-990 (T-100) NU-BUS G-64 VME-BUS Z-BUS	IEEEP1000 ОСТ4.ГО.304.203	1, 4 2, 3 1, 4, 24	65, 66 67 9 33, 34
Signetic/Philips Texas Instruments	Нидерланды США	VME-BUS E-BUS TM-990 (T-100) NU-BUS G-64 VME-BUS Z-BUS	Проект IEC47B IEEEP1014	25, 28, 29 25	4 16 2
Jhompson-CSF	Франция	G-64 VME-BUS Z-BUS	IEEEP1296		32, 68, 69
Zilog	США	VME-BUS Z-BUS	IEEEP1014	1, 2, 18, 19, 26, 27 5	
IBM	США	PCXT			

Примечание. Тип процессоров: 1. I8080. 2. Z80. 3. I8085. 4. I8086. 5. I8088. 6. I80186. 7. I80188. 8. 180286. 9. M6502. 10. M6800. 11. M6902. 12. M6808. 13. M6809. 14. M68000. 15. M68008. 16. NS C800. 17. NS I6032. 18. Z800. 19. Z8000. 20. T88000. 21. j—11 (1811). 22. NS32016. 23. T9995. 24. LSI —11/23. 25. TMS9900. 26. Z8001. 27. Z8002. 28. TMS9981. 29. TMS9995. 30. K581. 31. K588. 32. K1801. 33. K1806.

значительная трудоемкость разработки программного и диагностического обеспечения определила ряд концепций по развитию и совершенствованию технических характеристик интерфейса:

организация мультипроцессорной работы по параллельным и последовательным магистралям;

использование многошинной структуры (локальная, системная межсегментная, последовательная и др.), обеспечивающей создание сложных многокаркасных комплексов с высокой производительностью и надежностью;

создание единого аппаратного, программного и диагностического обеспечения для построения простых и сложных комплексов;

унификация статусных регистров и

протоколов обмена данных для обеспечения прозрачности программирования устройств, работающих в реальном масштабе времени, на всех иерархических уровнях системы;

расширение средств арбитража от централизованных к распределенным и прерываний от последовательных к векторным с обеспечением минимального времени реакции при обслуживании;

применение мультиплексирования при передаче функционально различной информации по одним и тем же шинам;

использование кодирования функциональных линий для цепей управления, состояния и др.;

совершенствование логических и географических способов адресации устройств для обеспечения гибкости

изменения конфигурации системы программным способом;

использование микропроцессорных средств с различной технологией изготовления и разрядностью 8...32;

обеспечение процессорной независимости магистрали с использованием интерфейсных БИС;

совершенствование дистанционных, аппаратных и программных средств диагностики и восстановления в работающей системе;

использование в интерфейсной части единого вторичного источника питания напряжением 5В;

применение единых конструктивов Евромеханика, разработанных на основе рекомендаций МЭК 297-1, МЭК 297-3, МЭК 603-2,

1. Futurebus. Specifications for advanced microcomputer backplane buses. Draft 6.2 proposed standard P896.1 // IEEE, November.—1983.
2. George P. White Bus structure eases multiprocessor integration // Computer design.—1984.—Vol. 23, N 7.—P. 123—135.
3. Спецификация шинной архитектуры Мультибас-II Multibus II System Architecture Specification, Handbuch der Firma Intel—Chapter I—VI.—1983.—Intel.
4. Hans-Georg Althoff. Multiprozessorbus—System für Europa karten // Elektronik.—1980.—N 18.—P. 57—62.
5. Andrene A. Allison. Status Report on the P896 Backplane Bus // IEEE Micro.—February 1981.—P. 67—82.
6. FASTBUS modular high speed data acquisition system for high energy physics. Документ IEC/TC45/CB186.
7. Шина VME. Руководство по эксплуатации.—М.; ВЦП, 1983, СР-82290—239.
8. Elmquist, Kells A; Fullmer, Howard, Gustareson David B, and Morrow, George, «Standard Specification for S-100-Bus Interface Devices» // IEEE Task 696. I/D2 Computer.—1979, Vol. 12, N 7.—P. 28—82.
9. The German National Committee submit herewith the specification for the AMS BUS—System for consideration in SC47B.—IEC47B (Germany) 5 May, 1982.
10. ОСТ 25 969—82. Системы малых электронных вычислительных машин. Интерфейс И41. Технические требования—60 с.
11. ГОСТ 26.765.51-86. Система электронных модулей. Магистральный параллельный интерфейс (МПИ). Общие требования.
12. David B. Gustavson. Fastbus status from a system designers point of view // IEEE Transactions on Nuclear Science. Vol. NS-28, N 5.—October 1981.—P. 3796—3600.
13. Deiss S. R. Applicability of the Fastbus Standard to distributed control // IEEE Transactions on Nuclear Science.—Vol. NS-28, N 3.—June 1981.—P. 2222—2224.
14. Рекомендуемый стандарт на шину 796 для систем, содержащих микроЭВМ (Рабочая группа IEEE по шине 796), октябрь 1980, ЦНИИатоминформ, 1982.—Вып. 15.—С. 5—58.
15. ЕВРОБАС. Стандарт на модульные системы управления, обработки и сбора данных для промышленных применений. Временная спецификация. Комитет ЕЗОНЕ, декабрь, 1960 (Eurobus, Modular data acquisition processing and control system standart for industrial application. Tentative specification.—ECONE Committee).—М.: ВЦП.—1981, Г-10862.
16. Jencharis L. But the right bus evokes the system's best // Electronics Design.—1980.—Vol. 28, N 10.—P. 125—130.
17. Barril Nicholson. STE-Eurocard bus, P1000 // Microprocessor and microsystems.—1983.—Vol. 7, N 6.—P. 269—271.
18. Carl Warren. Understanding bus basics helps resolve design conflicts // EDN.—May 27.—1981.—P. 159—173.
19. Modulares — Microprozessor — stenersystem Das universell System Für die fertigungstechnik. Проспект MPSt с выставки в Ганновере, 1981.
20. Мячев А. А. Система ввода-вывода ЭВМ.—М.: Энергоатомиздат, 1983.
21. Дьюре Л. Исследование системы Эригон-бас. Durie L. and other. The Erigon Bas Staty, CERN 82—07.—June 1982.—Geneva 1982.—М.: Энергоатомиздат 1983.—Вып. 17.—С. 3—60.
22. Эрглис К. Э. Магистрально-модульные многопроцессорные измерительно-управляющие системы (обзор) // ПТЭ.—1983.—№ 1.—С. 7—20.
23. Рыбаков В. Г. Современные магистрально-модульные системы, Характеристика и тенденции развития (обзор) // ПТЭ.—1983.—№ 4.—С. 7—28.
24. ЕВРОБАС. Стандарт модульной системы сбора данных, обработки и управления для промышленного применения. Проект технического описания. Комитет ЕЗОНЕ, декабрь 1980.—М.: ЦНИИатоминформ, 1981.—Вып. 12.—166 с.
25. John Hill. Eurobus—the UK—designed general—purpose backplane bus // Microprocessors and microsystems.—1982.—Vol. 6, N 9.—P. 483—487.
26. Specification for Eurobus International Standard IS06951-1986E Information processing—Processor System bus interface (Eurobus A).
27. Черных Е. В. Стандартные интерфейсы для программно-модульных многопроцессорных систем (обзор) // ПТЭ.—1982.—№ 4.—С. 5—35.
28. Проект стандарта на малые системы ЕС, предложенного группой по изучению малых систем комитету ЕСОНЕ.—Цюрих, 1981.—М.: Энергоатомиздат, 1982.—Вып. 13.—90 с.
29. Proposal for a Small System Standart E3S, Submitted by Small System Study Group to ESONE AGA, Zürich, 1981, SS/DEC/40.
30. Andreas Grimm. G—64: Kleiner Bus nut grobem Komfort // Electronics.—1985.—H1.—P. 109—112.
31. Cosma Pabovctsidis and Jean-Louis Ebener G-64 But snits mid-range industrial Me-applications // EDN.—1985.—N 9.—P. 225—263.
32. Carl Warren. High-performance buses clear a path for future MCS // EDN.—1981.—Vol. 26.—June 10.—P. 157—187.
33. Draft. Microprocessor system bus 1, 8-bit and 16-bit data. International Electrotechnical Commission. Technical Committee N 47: Semiconductor Devices and Integrated Circuits. Sub-Committee 47B // Microprocessor Systems—Switzerland Geneva. July—1983.
34. Микропроцессорная системная шина 1, 8-разрядные и 16 разрядные данные. Проект стандарта МЭК. М.: ВЦП, 1983, Е-76236.
35. Multibus II System Architecture Specification handbuch dor firma Intel—Chapter I—IV. Intel, 146077—13, 1983.
36. John Deaston. Structure des bus dans Jarchitecture Multibus II. Electronique Industrielle N63/15—01—1984. P. 49—53.
37. Johan Gezer. Multibas II: 32-bit bus für leistungsfähige offene system // Elektronik.—1984.—N 1.—P. 52—55.
38. Джон Бистон. Шина для будущих 32-разрядных многопроцессорных систем // Электроника: Пер. журн. США Electronics—1984.—№ 6.—С. 43—51.
39. Richard W. Boberg Proposed Microcomputer System 796 Bus Standard // Computer.—October.—1980.—P. 89—105.
40. Richard Boberg. Major standardization issues of the proposed IEEE 796 bus—Multibus // Microprocessors and microsystems.—1982.—Vol 6, N 9.—P. 471—474.
41. Auswertelogik für inkremental Wegme Systeme // Elektronik.—1985.—H1.—P. 45—50.
42. Hans jurgen Nishik. 68008 erscrit Z80 // Elektronik.—1984.—№ 14.—P. 120—123.
43. Stephan Speth IEEE—p896—Bus für 16— and 32—bit—Prozessoren // Elektronik.—1982.—N 10.—P. 107—110.
44. MC-86. Famile in Doppelpaforformat. Der Weg zum Erfolg auf dem europäischen 8/16—Bit Microcomputermarkt. Mania Vertriebsgesellschaft mbH Hauptstra be86. 6384 Schmitt 2 (Niederreifenberg).
45. ОСТ 25 795—78 СМ ЭВМ. Интерфейс общая шина.
46. ОСТ 25 904—80 СМ ЭВМ и АСВТ—Интерфейс ИВС. Структура и состав. Логические и электрические условия. Конструктивная реализация.
47. ОСТ 25 25 984-82 ГСП. Комплек технических средств для локаль-

- ных информационно-управляющих систем (КТС ЛИУС-2). Интерфейс ИК1. Технические требования.
48. Диденко К. И. Проектирование агрегатных комплексов технических средств для АСУ ТП — М.: Энергоатомиздат, 1984.— 168 с.
 49. Carl Warren. Compare MC — bus specs to find the bus you need // EDN.— 1981.— Vol. 26.— June 10.— P. 141—153.
 50. Mark Garetz. P696/S100 — a bus which supports a wide range of 8-bit and 16-bit processors // Microprocessors and microsystems.— 1982.— Vol 6, N 9.— P. 466—704.
 51. Richard De Bock. Versabus — a multiprocessor bus standard — and VME bus — its Eurocard counterpart // Microprocessors and microsystems.— 1982.— Vol 6, N 9.— P. 475—481.
 52. Michael Rudgh. VME-bus modulares konzept für Me-karten mit Europaformat // Elektronik.— 1982.— N 10.— P. 90—96.
 53. VME-bus Specification, Motorola, 1982, Dcc.
 54. International Electrotechnical Commission IEC821BUS. Microprocessor system bus for 1 to 4 byte.
 55. Peter Harold. Powerful local buses join VME bus // EDN.— 1985.— N 9.— P. 199—208.
 56. Klaus Petersen. VMS-bus and VMX bus: Erweiterung des VME bus — konzeptes // Elektronik.— 1984.— N 12.— P. 76—81.
 57. ОСТ И 305.903-680. Микропроцессорные средства вычислительной техники. Технические средства. Интерфейс межмодульный. Техническое описание.— 20 с.
 58. Форс. Стандартная микропроцессорная шина, упрощающая задачи разработчиков микрокомпьютеров // Электроника: Пер. журн. США Electronics.— 1978.— Т. 51.— № 15.— С. 33—41.
 59. CMOS — Computer in Europa — Format // Elektronik.— 1982.— N 7.
 60. ФАСТБАС. Модульная действующая система сбора данных для физики высоких энергий и других применений. Временная спецификация. Комитет НМ США, июль 1980.— М.: ВЦП, 1981, Г—10863.— 363 с.
 61. Василадзе С. Г. FASTBUS — стандарт для построения быстродействующей электронной аппаратуры IV поколения // ПТЭ.— 1982.— № 5.— С. 5—27.
 62. FASTBUS, modular high speed data. Acquisition Systems, U.S. NIM Committee, 1983, Sept.
 63. IEEE Micro, N 4, 1985.
 64. Tim Elsmore. Standard bus for 8-bit microprocessor systems // Microprocessors and microsystems.— 1982.— Vol 6, N 9.— P. 455—465.
 65. Иoffee А. Ф. М. Массовые персональные ЭВМ серии АГАТ // Микропроцессорные средства и системы.— 1984.— № 1.— С. 56.
 66. ОСТ 4.ГО.304.203. Устройства ввода-вывода, устройства подготовки данных. Абонентские пункты. Элементы замены типовые микропроцессорные. Микропроцессорная магистраль. Основные параметры. Редакция 1—79, 14 с.
 67. Siemens components. 1983, H12.
 68. Bunning j Z-BUS and Peripheral Support Packages the Distributed Computer System Together // Electronic Design.— 1979.— Vol 27, N 24.— P. 144—149.
 69. Трейстер Р. Персональные компьютеры фирмы ИБМ.— М.: Мир, 1986.

За справками обращаться по адресу: Москва, ул. Вавилова, 24, ИИЭУМ. Тел. 234-18-65.

Статья поступила 6 ноября 1986 г. (Окончание см. в № 1, 1988 г.)

КРАТКИЕ СООБЩЕНИЯ

УДК 681.124.62

А. Я. Гофман

УНИВЕРСАЛЬНОЕ УСТРОЙСТВО СОПРЯЖЕНИЯ СЧИТЫВАТЕЛЯ С ПЕРФОЛЕНТЫ FS-1501 С ИНТЕРФЕЙСОМ ИРПР

Во многих случаях, когда объем вводимой в микроЭВМ информации невелик, все еще целесообразно использовать перфоленочное оборудование. Например, во встраиваемых управляющих микроЭВМ использовать НМД и НГМД часто целесообразно, особенно для эпизодического ввода небольших по объему тестовых или отладочных программ.

Устройство сопряжения (далее УС) позволяет согласовывать по уровням сигналов и управлению считыватель с перфоленты (СП) FS-1501 с интерфейсом для радиального подключения устройств с параллельной передачей информации ИРПР (которым оснащены большинство микроЭВМ). УС состоит из преобразователей уровня, буферного регистра, узла управления ходом СП, узлов формирования сигнала готовности ГИ—И, формирования строга СТР—И, одновибраторов и управляемых инверторов.

При неготовности СП узел формирования сигнала ГИ—И заблокирован. При опускании поворотного плеча СП узел разблокируется и разрешает формирование сигнала ГИ—И при нажатии на кнопку. В ответ на ГИ—И ИРПР выставляет сигнал ЗП—И, по которому производится очистка буферного регистра и в СП выдается сигнал «СТАРТ». При поступлении из СП сигнала СИ информация строится в буферном регистре,

в ИРПР выдается сигнал СТР—И и в СП выдается сигнал «СТОП».

После считывания данных из регистра данных источника ИРПР снимает сигнал ЗП—И. Это вызывает снятие устройством сопряжения сигнала СТР—И. На этом цикл ввода символа с перфоленты окончен и ИРПР может выставить следующий запрос ЗП—И.

Введение с УС узла формирования сигнала готовности вызвано тем, что ИРПР выставляет сигнал ЗП—И сразу по приходу ГИ—И, т. е. при нажатии кнопки перфоленты будет продвигаться на одну строку. Это не создает неудобств и позволяет контролировать правильность запроски СП. Повторным нажатием кнопки можно снять готовность устройства.

Управляемые инверторы позволяют установить активные уровни сигналов в зависимости от конкретной реализации ИРПР.

УС подключается непосредственно к устройству байтового параллельного интерфейса, входящего в состав микроЭВМ «Электроника МС 1201» (в рамках протокола ИРПР). При этом переключатели SA1.6 и SA1.8 микроЭВМ должны быть установлены в положение 0, а SA1.7 — в положение 1. Это позволяет без переделок микроЭВМ и СП работать со стандартным программным обеспечением с адресами регистров состояния (177550) и данных (177552).

Если планируется использовать УС только с одним типом микроЭВМ, управляемые инверторы можно исключить из схемы УС. Тогда, например, для микроЭВМ «Электроника МС 1201» нужно использовать сигналы ГИ—ИН, ЗП—ИВ, СТР—ИН, СБРОС ВУ Н, D0... D7, изменив положение переключателя SA1.6 на 1 (буква Н обозначает активный низкий уровень сигнала, В — высокий).

Телефон для запросов о технической документации: 55-30-08, Донецк
Статья поступила 15 января 1987 г.

Т. А. Куправа

ИНТЕРПРЕТАТОР ЯЗЫКА ФОКАЛ В МИКРОСХЕМЕ ПЗУ

Ряд версий языка Фокал разработан и реализован в виде интерпретаторов для микроЭВМ с системой команд микроЭВМ «Электроника 60». Интерпретаторы, содержащиеся в масочных БИС ПЗУ типа К1801РЕ1, К1801РЕ2 емкостью 8К байт, отличаются компактностью исходных текстов программ, достаточно развитыми диалоговыми, графическими возможностями. Их целесообразно применять в недорогих микроЭВМ с внешними ресурсами, недостаточными для постановки компиляторов языков Фортран, Паскаль и др.

В таблице приведены обобщенные характеристики серийных БИС ПЗУ с интерпретаторами языка Фокал.

Обобщенные характеристики интерпретаторов
языка Фокал

Характеристика	Версия интерпретатора		
	К1801РЕ1-018	К1801РЕ2-084	К1801РЕ1-058
МикроЭВМ	«Электроника БК-0010»	«Электроника БК-0010Щ»	ДВК-1М
Область памяти	120 000...137 776	137 776	140 000...157 776
Способ обращения к ВУ	Посредством драйверов в ПЗУ К1801РЕ1-017		Непосредственно к регистру ВУ
Наличие драйверов ИРПС сети	—	+	+
Дополнительные ВУ, поддерживаемые опратораи	Бытовой магнитофон		Печатающее устройство, перфоленточные УВВ
Данные	Арифметические переменные, одномерные и двумерные массивы; диапазон $10^{-28} \dots 10^{+28}$, точность — до 6 десятичных цифр		
Число операторов	19	17	20
Встроенные функции	13 арифметических функций, генератор случайных чисел ($-1 \dots +1$), управление общей магистралью микроЭВМ, ввод-вывод символьных данных, программируемая, одного параметра		Считывание и обнуление таймера
Графика	Управление параллельным портом ввода-вывода		—
	Управление символьным маркером		—
	Две функции 256×256 точек цв. 256×512 точек ч/б; бытовой телевизор		10 функций 286×400 точек ч/б; дисплей 1Б1Э-00-013.1 с контроллером графического дисплея ДВК

Диагностика ошибок — текстовая, редактирование — построечно-экранное

Они нашли применение в школьных кабинетах вычислительной техники. Интерпретатор в составе школьной микроЭВМ поддерживает обмен информацией по каналу ИРПС с центральной микроЭВМ аудиторной сети, обеспечивает автономную работу учащегося. Интерпретирующий режим отладки и выполнения программ, исключая скучные этапы ожидания результатов трансляции, в сочетании с хорошими диагностическими возможностями, делает интерпретатор удобным инструментальным средством оснащения компьютерных классов.

Программа-интерпретатор выполнена в виде набора модулей с четко определенными функциями и интерфейсами, базируется на однозначном формальном определении языка и составлена с учетом особенностей архитектуры микроЭВМ. Для использования преимуществ изготовления в ПЗУ и повышения «живучести» интерпретирующей системы предусмотрены различные виды тестирования, защиты от сбоев, динамического контроля.

Адрес для справок: 384900, г. Сухуми, ул. Чогуа, д. 40, Абхазский государственный университет им. Горького, каф. алгебры и геометрии, тел. 2-86-94.

Сообщение поступило 25 декабря 1986 г.

УДК 681.32

М. П. Попов, Д. П. Забоев, В. Г. Зайнулин,
С. В. Забоев, В. С. Исаков

ИНТЕРФЕЙС МИКРОЭВМ «ЭЛЕКТРОНИКА ДЗ-28» — АВТОМАТ «ОРГТЕКСТ»

Для эксплуатации микроЭВМ «Электроника ДЗ-28» в комплексе с организационным автоматом «Оргтекст» разработано устройство сопряжения, состоящее из двух функциональных модулей размером 100×150 мм (интерфейсы вывода и ввода), обеспечивающих обмен алфавитно-цифровой информацией между микроЭВМ и автоматом. Для внешних соединений использованы разъемы «ввод-вывод» и «перфоратор» на микроЭВМ и «фотосчитыватель» на автомате «Оргтекст». Такой способ подключения позволил сохранить конструктивные особенности и автономность каждого из агрегатов.

Для управления процессором обмена алфавитно-цифровой информацией разработано программное обеспечение.

За справками следует обращаться по адресу: 167610, ГСП, Сыктывкар, ул. Коммунистическая, д. 28, Институт биологии Коми филиала АН ССР.

Сообщение поступило 7 июля 1987 г.

ИНФОРМАЦИЯ

Академия наук Украинской ССР и Институт кибернетики имени В. М. Глушкова АН УССР провели Международный конкурс по созданию базового интерфейса программных средств (ИНТЕРФЕЙС СЭВ).

Целью конкурса было определение концепций и принципов построения базового интерфейса и мобильной технологической среды, обеспечивающих коллективную разработку качественных программных средств, новые принципы их фондирования и стыковки друг с другом при независимой разработке, а также их многократную применимость взамен дублирующих разработок.

Из 64 работ, представленных на конкурс, лучшими были признаны 15: НРБ—2, ВНР—2, ГДР—1, ПНР—1, СССР—7, ЧССР—3.

УДК 621.328.3.049.776

В. Н. Ачкасов, А. Н. Глазнев, С. А. Цыбин,
Н. Я. Мешеряков

**БЫСТРОДЕЙСТВУЮЩИЕ НЕТАКТИРУЕМЫЕ
СТАТИЧЕСКИЕ ОЗУ С БАЙТОВОЙ
ОРГАНИЗАЦИЕЙ КМ581РУ5**

Микросхемы КМ581РУ5 (группы Б, В, Г) представляют собой нетактируемые (асинхронные) статические ОЗУ емкостью 16 кбит (2048 бит×8), изготовленные по планарной технологии КМОП с самосовмещенным по-

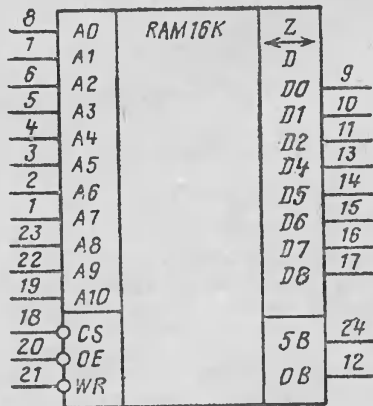


Рис. 1. Условное графическое обозначение ОЗУ КМ581РУ5

Таблица 1
Назначение выводов микросхемы
ОЗУ КМ581РУ5

Вывод	Назначение
1...8	Входы адреса A0...A7
9...11	Входы-выходы данных D0...D2
12	Вывод общий
13...17	Входы-выходы данных D3...D7
18	Вход сигнала выбора кристалла CS
19	Вход адреса A10
20	Вход сигнала разрешения выхода данных OE
21	Вход сигнала записи WR
22, 23	Входы адреса A8, A9
24	Напряжение питания U _{cc}

Таблица истинности

Входы управления			Входы-выходы данных D0...D7	Режим работы
CS	OE	WR		
1	X	X	Третье состояние	Схема не выбрана (хранение)
0	1	1	Третье состояние	
0	0	1	Выходная информация	Схема выбрана (считывание) Выход запрещен Считывание
0	X	0	Входная информация	

Примечание: X — произвольное логическое состояние

Таблица 3
Электрические параметры микросхем серии КМ581

Параметр, единица измерения	Обозначение	Значение параметра					
		КМ581РУ5Б		КМ581РУ5В		КМ581РУ5Г	
		не менее	не более	не менее	не более	не менее	не более
Выходное напряжение низкого уровня при токе нагрузки I _{OL} =4 мА, В	U _{OL}		0,4		0,4		0,4
Выходное напряжение высокого уровня при токе нагрузки I _{OH} =1 мА, В	U _{OH}	2,4		2,4		2,4	
Ток потребления, мА	I _{CC}	80		70		70	
Динамический ток потребления на максимальной рабочей частоте, мА	I _{CCO}	80		70		70	
Ток утечки по входам (CS, WR, OE, A0...A10), мкА	I _{CS1}	15		15		15	
Ток утечки на входах-выходах (D0...D7), мкА	I _{CS2}	2		2		2	
Ток утечки по входам (CS, WR, OE, A0...A10), мкА	I _{LI}	-10	10	-10	10	-10	10
Ток утечки на входах-выходах (D0...D7), мкА	I _{LO}	-10	10	-10	10	-10	10
Емкость входов (CS, WR, OE, A0...A10), пф	C _i		5		5		5
Емкость входов-выходов (D0...D7), пф	C _{I/O}		7		7		7

Таблица 4

Предельно допустимые электрические параметры

Параметр, единица измерения	Обозначение	Значение параметра	
		не менее	не более
Напряжение питания, В	U_{CC}	4,5	5,5
Входное напряжение высокого уровня по выводам D0...D7, В	U_{IH}	2,2	$U_{CC} \pm 0,3В$
Входное напряжение высокого уровня по входам A0...A10, С, OE, В	U_{IH}	2,2	6
Входное напряжение низкого уровня, В	U_{IL}	-0,3	0,8
Отрицательное значение входного импульсного напряжения низкого уровня при длительности импульса не более 50 нс, В	U_{IL}	-1,0	—
Выходной ток высокого уровня, мА	I_{OH}	-1,0	—
Выходной ток низкого уровня, мА	I_{OL}	—	4,0
Емкость нагрузки, пФ	C_L	—	500

Таблица 5

Временные параметры микросхем серии KM581

Параметр	Обозначение	Норма, нс		
		KM581PY5B	KM581PY5E	KM581PY5Г
		не менее	не менее	не менее
Время цикла считывания	$t_{CY} (RD)$	120	150	200
Время цикла записи	$t_{CY} (WR)$	120	150	200
Время выборки адреса	$t_A (A)$	120	150	200
Время выборки разрешения	$t_A (OE)$	80	100	120
Время установления сигнала выбора кристалла относительно сигнала адреса	$t_{SU} (A - CS)$	Не регламентируется		
Время выбора	$t_A (CS)$	120	150	200
Время удержания сигнала записи относительно сигнала выбора кристалла	$t_H (CS - WR)$	70	90	120
Время удержания сигнала записи относительно сигнала адреса	$t_H (A - WR)$	105	120	140
Время установления сигнала записи относительно сигнала адреса	$t_{SU} (A - WR)$	20	20	20
Длительность сигнала записи	$t_W (WR)$	70	90	120
Время восстановления сигнала записи относительно сигнала адреса	$t_{REC} (A - WR)$	5	10	10
Время удержания сигнала выбора кристалла относительно сигнала записи	$t_H (WR - CS)$	70	90	120
Время удержания сигнала записи относительно сигнала входной информации	$t_H (CDI - WR)$	35	40	60
Время сохранения сигнала входной информации после сигнала записи	$t_V (WR - DI)$	5	10	10

Примечание. Длительность фронтов управляющих сигналов не более 100 нс, но не должна превышать 10 % от длительности цикла

ликремниевым затвором и высокоомными поликремниевыми резисторами. Число элементов на кристалле равно 102 000. Микросхемы выполняются в 24-выводном металлокерамическом корпусе типа 2120.24-11. По входным и выходным уровням сигналов совместимы с TTL схемами. Имеют один источник напряжения питания 5 В.

Условное графическое обозначение микросхемы приведено на рис. 1, назначение выводов показано в табл. 1. На выводы микросхемы, наряду с адресными сигналами и сигналами входной информации, подаются три управляющих: выбор кристалла CS, разрешение выхода данных OE и сигнал записи WR, определяющие режимы работы ОЗУ в соответствии с таблицей истинности (табл. 2). Сигнал CS высоким уровнем напряжения переводит микросхему в режим хранения. Сигнал OE определяет состояние выходных буферных каскадов (высокий уровень напряжения OE устанавливает выходы в третье состояние). Сигнал управляет режимом записи-считывания.

Структурная схема ОЗУ приведена на рис. 2, электрические параметры — в табл. 3, значения предельно допустимых электрических режимов эксплуатации — в табл. 4. Микросхема содержит накопитель, состоящий из 16384 запоминающих элементов, дешифраторы строк и столбцов со схемами ввода-вывода, блоки формирователей адресных сигналов и управления режимом, устройства ввода и данных. Накопитель разбит на 8 секций по 2048 запоминающих элементов. В качестве запоминающего элемента выбран 4-транзисторный элемент триггерного типа с высокоомными поликремниевыми резисторами. Синхронную работу всех узлов ОЗУ в режимах записи, считывания и хранения обеспечивает блок управления режимом.

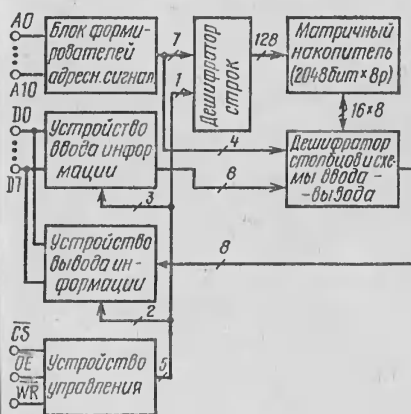


Рис. 2. Структурная схема ОЗУ KM581PY5

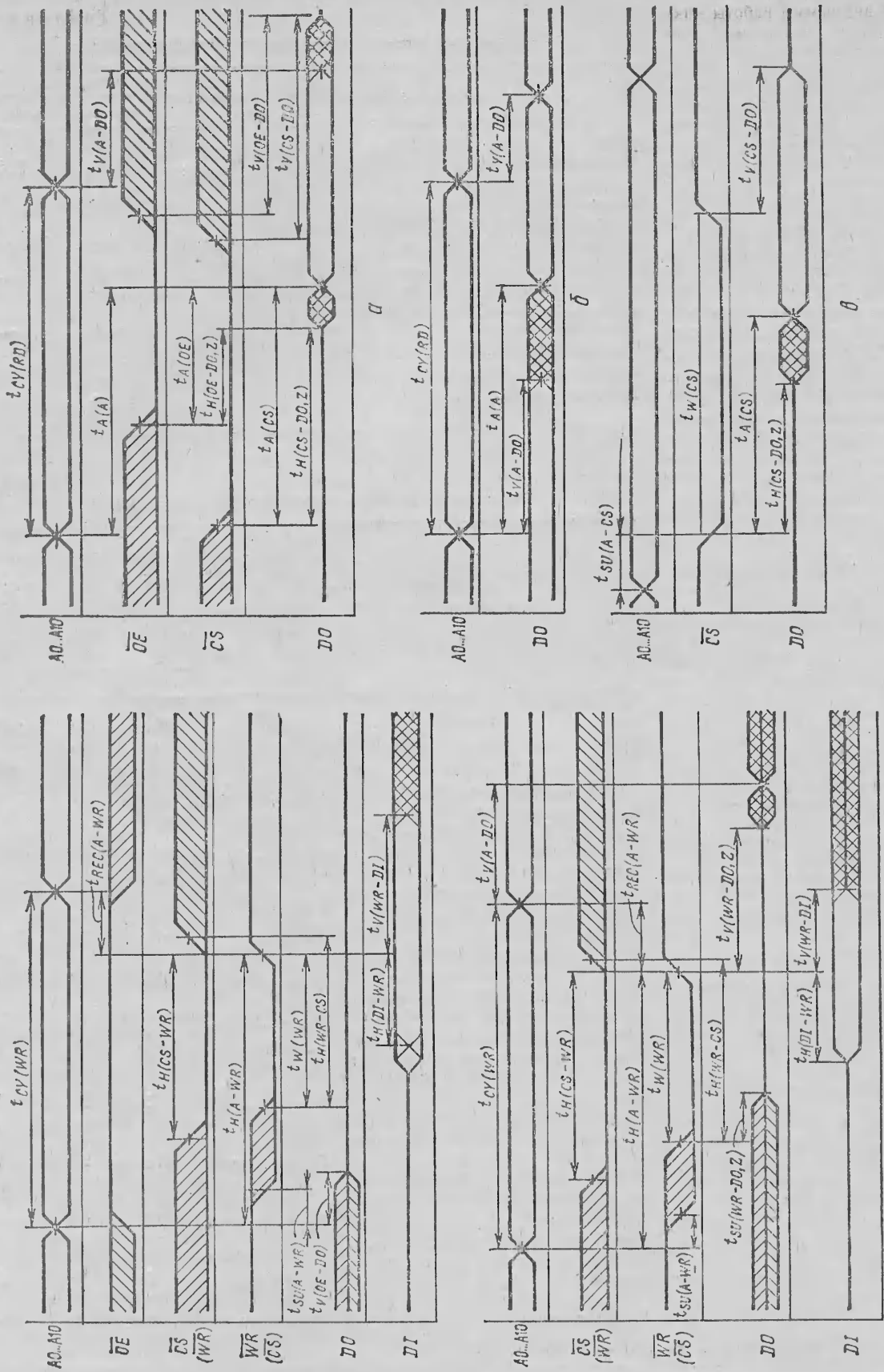


Рис. 3. Режим считывания:
 а — $U_{WR} = U_{IH}$; б — $U_{WR} = U_{IL}$; $U_{CS} = U_{IH}$;
 $U_{OP} = U_{IH}$; в — $U_{WR} = U_{IH}$; $U_{OE} = U_{IL}$

Рис. 4. Режим записи

Таблица 6
Справочные временные параметры микросхем серии КМ581

Временные диаграммы работы микросхемы в режиме считывания представлены на рис. 3, в режиме записи — на рис. 4, а соответствующие им величины временных интервалов — в табл. 5. Значения справочных временных параметров микросхем приведены в табл. 6.

При выполнении операции считывания, когда сигнал WR находится в состоянии «Лог.1», а сигналы CS и OE — в состоянии «Лог.0», данные из восьми запоминающих элементов (из одного в каждой секции) одновременно передаются через схемы ввода-вывода в устройство вывода информации и далее на выходы $D0...D7$ микросхемы. В режиме считывания по сигналу адреса (см. рис. 3, б) минимальное значение цикла считывания $t_{CY}(RD)$ определяется временем выборки адреса $t_{A(A)}$ и равно ему. В режиме считывания по сигналу выбора кристалла CS (см. рис. 3, в) минимальное значение цикла считывания $t_{CY}(RD)$ определяется суммой времени выбора $t_{A(CS)}$ и времени установления сигнала выбора кристалла относительно сигнала адреса $t_{SU}(A-CS)$.

При выполнении операции записи, когда сигналы WR и CS находятся в состоянии «Лог.0», выходные буферные каскады устанавливаются в третье состояние (состояние высокого импеданса). Выходные данные через устройство ввода информации и схемы ввода-вывода записываются в соответствующие восемь запоминающих элементов (одинаковые в каждой секции накопителя). В режиме записи входная информация должна быть подана на информационные входы-выходы $D0...D7$ после переключения выходных буферных каскадов в третье состояние.

Сигнал OE управляет непосредственно состоянием выходных каскадов

Параметр	Обозначение	Норма, нс					
		КМ581РУ5Б		КМ581РУ6В		КМ581РУ5Г	
		не менее	не более	не менее	не более	не менее	не более
Время удержания выхода в третьем состоянии относительно сигнала выбора кристалла	$t_H(CS - DO, Z)$	10	—	15	—	15	—
Время удержания выхода в третьем состоянии относительно сигнала разрешения выхода	$t_H(OE - DO, Z)$	10	—	15	—	15	—
Время сохранения сигнала выходной информации после сигнала выбора кристалла	$t_V(CS - DO)$	0	40	0	50	0	60
Время сохранения сигнала выходной информации после сигнала разрешения выхода	$t_V(-DO)$	0	40	0	50	0	60
Время сохранения сигнала выходной информации после изменения сигнала адреса	$t_V(A - DO)$	10	—	15	—	15	—
Время установления выхода в третье состояние относительно сигнала записи	$t_{SU}(WR - DO, Z)$	0	50	0	60	0	60
Время сохранения выхода в третьем состоянии после сигнала записи	$t_V(WR - DO, Z)$	5	—	10	—	10	—

и позволяет упростить временную диаграмму сигналов на информационных входах-выходах $D0...D7$ в режиме записи (см. рис. 4).

В режиме хранения, когда сигнал CS равен «Лог.1», устройства ввода и вывода информации и выходные буферные каскады находятся в выключенном состоянии, независимо от со-

стояния сигналов на других выводах микросхемы дешифратора строк.

Высокое быстродействие при умеренной потребляемой мощности и асинхронный режим работы микросхем позволяют прогнозировать их широкое применение в различных средствах вычислительной техники и системах ЧПУ.

КРАТКОЕ СООБЩЕНИЕ

УДК 681.3.06

Г. Х. Геворкян, В. Н. Семенов

АННОТАЦИЯ К ПАКЕТУ ПРОГРАММ ДЛЯ РАСЧЕТА ЭЛЕКТРИЧЕСКИХ СХЕМ

Назначение пакета — расчет схем произвольной топологии с линейными элементами методами контурных токов (КТ) и узловых потенциалов (УП) на постоянном и переменном токах.

Язык — подмножество языка Бейсик, имеющееся на всех отечественных ПЭВМ, обуславливающее независимость пакета от технических и базовых программных средств.

Состав: 5 программ для метода КТ (МКТО...МКТ4),

5 программ для метода УП (МУПО...МУП4) и 6 вспомогательных подпрограмм.

Режим работы — диалоговый. Возможности: от ручного составления и ввода матриц инцидентий до полностью автоматизированного расчета, когда требуется только ввести параметры ветвей и номера узлов. Расчет стационарного режима на постоянном и переменном токах.

Минимальные требования к составу ПЭВМ: процессор с памятью 16К байт и дисплей.

Применяется для расчетов: профессиональных — полный анализ цепей произвольной сложности, получение АФЧ характеристик, определение резонансов и т. п.; учебных — освоение методики расчета цепей на постоянном и переменном токах, в том числе и с применением матричных методов, основанных на методах КТ и УП.

Адрес для обращения за документацией: 143500, Московская область, г. Истра, отделение Всесоюзного электротехнического института им. В. И. Ленина, тел. 4-67-70.

Сообщение поступило 30 марта 1987 г.

Председатель семинаров — Г. Р. Громов
 Научный руководитель семинаров — А. П. Ершов

15 сентября 1987 г. состоялся семинар «Аппаратная структура и программное обеспечение типового микропроцессорного комплекса и устройств связи с объектами автоматизации».

Инженер-консультант Головного консультационно-технического центра МЭП СССР С. В. Ермуратская (468-81-75, Москва) рассказала о созданном в ГКЦ МЭП учебном видеофильме «Микропроцессоры: вводный курс». В курс вошли 10 серий по 25—40 мин.: «История развития вычислительной техники» и «Аппаратные средства микропроцессорных систем» (лектор — А. К. Малиновский); «Методы программирования микропроцессоров (ассемблер)» (лектор С. А. Горюшина); «Методы программирования микропроцессоров (машинный код)» и «Разработка микропроцессорных систем» (лектор — А. В. Александров); «Разработка и выполнение программ в реальном масштабе времени» (лектор — М. Г. Весноватов); «Средства и методы отладки микропроцессорных систем», «Архитектура и периферийные устройства микропроцессоров», «Применение микропроцессоров в народном хозяйстве» и «Тенденции развития микропроцессорной техники» (лектор — А. В. Александров)**.

Несколько серий видеофильма (в том числе и по теме семинара) были показаны участникам в фойе до и после семинара.

Инженер-консультант ГКЦ МЭП СССР А. О. Брызгалин (468-81-75, Москва) показал два промышленных контроллера [один — на базе МПК БИС КР580, другой — на основе БИС КМ1816ВЕ35], применимых для решения наиболее простых задач автоматизации, когда использовать мини- и даже персональные ЭВМ неэффективно. Он рассказал об их функциональных и технических характеристиках (статьи об этих контроллерах см. в одном из очередных номеров «МП»).

В. Н. Барышников (939-22-83, 939-10-87, Москва) рассказал о модульной системе автоматизации «Вариант-50» и показал ее в действии.

Назначение системы «Вариант-50» — автоматизация относительно простых научных экспериментов, приборов и установок, когда использовать унифицированную аппаратуру, например, типа «КАМАК» нецелесообразно из-за высокой стоимости сложных аппаратных средств. Предлагаемая система построена в виде кассеты из функционально законченных модулей, выходящих на магистраль межмодульного обмена информацией. С конкретными ЭВМ (например, «Электроника-60», «Электроника ДЗ-28») кассета связывается через модули контроллеров, организующих обмен информацией между ЭВМ и магистралью. Вместо контроллера в блок можно установить модуль микроЭВМ — при этом система превращается в автономный комплекс для решения определенных задач. ПЭВМ «Ириша» встраивается в эту кассету без контроллера.

Вставляемый в кассету (320×276×178 мм) каждый модуль представляет собой печатную плату (230×150×23 мм). Максимальное число модулей в кассете — 12.

Универсальный модуль связи с цифровыми приборами помогает подключить серийно выпускаемые цифровые измерительные приборы к системе, а универсальный приемопередающий регистр — серийно выпускаемые устройства (накопители на МЛ А311—4, накопители на МД ГМД-70, печатающее устройство ДЗМ-180 и т. п.). Универсальный модуль управления

Модуль индикации алфавитно-цифровой информации выводит на экран стандартного ТВ-монитора до 1024 знаков латинского и русского алфавита, цифр и специальных символов (32 строки по 32 символа), а модуль индикации графической информации — графическую информацию (512×256 точек).

Модуль управления графопостроителем содержит два 12-разрядных ЦАП, схемы управления пером и индикацией, встроенное ПЗУ для хранения программ работы с модулем и систему обратной опции графики.

Емкость модуля энергонезависимого ОЗУ — 4К байт, а модуля дополнительного ОЗУ — 64К байт.

Системный модуль содержит схемы канала прямого доступа, таймера, последовательного интерфейса, интерфейса для подключения клавиатуры, системного ПЗУ, блока звуковой индикации.

А. П. Казанцев (923-96-68, доб. 2-93; Москва) показал и охарактеризовал автоматизированное рабочее место исследователя на базе болгарской персональной ЭВМ «Правец», совместимой с «Apple II», о разнообразных устройствах связи с аналитическими инструментами биохимических исследований, программном интерфейсе пользователя «Интеракт», и разработанных (нетиповых) интерфейсных платах (для подключения двоично-десятичных прецизионных приборов и двухкоординатного самописца).

Затем он рассказал о созданной С. И. Бородиным (923-96-68, доб. 2-58) системе приборов лабораторной автоматизации (СПЛАВ). Это «конструктор» для создания сложных исполнительных устройств при автоматизации рутинных процессов в исследовательской лаборатории физико-химического профиля. У различных типов модулей (насосов, дозаторов, манипуляторов и т. д.) конструкция и логическая спецификация. СПЛАВ — это «руки» для ЭВМ.

Для эффективного использования персонального компьютера с системой СПЛАВ разработан контроллер в виде платы ПЭВМ «Правец» и создан программный интерфейс пользователя (система «Интеракт»), представляющий «меню» программ. Основные подсистемы — интерпретатор технологического языка и монитор дисковых операций. Диалог с пользователем система поддерживает текстовыми и музыкальными сообщениями. Для управления базой данных использовалась реляционная СУБД типа ПФС (персональная файловая система).

П. В. Полянский (107005, Москва, 2-я Бауманская ул., д. 5; МВТУ им. Н. Э. Баумана, кафедра АМ-7) сделал два сообщения.

Минимальное устройство связи с объектом (УСО) для ПЭВМ «Электроника БК 0010» предназначено для ввода восьми аналоговых сигналов в ПЭВМ, вывода одного аналогового и аппаратного поддержания функции таймера. Входные и выходные сигналы для УСО — напряжения в пределах от 0 до 10 В. ПЭВМ «Электроника БК 0010» совместно с этим УСО (55×55×145 мм) применены для управления высокотемпературной пайкой крупногабаритных изделий.

Система сбора данных (ССД) при исследовании объектов с распределенными параметрами выдает на исследуемый объект тестирующее воздействие, регистрирует отклик объекта и передает полученный массив данных в ПЭВМ «Электроника БК 0010». Система позволяет генерировать два линейно возрастающих тока с амплитудой от 0 до 1,5 А с точностью не хуже 0,5%. Цикл измерения длится 1 мс; за это время проводится 1024 измерения. Разрядность данных — 6 двоичных

* См. с. 1 цветной вкладки
 ** См. с. 2, 3 цветной вкладки

разрядов. Система сбора данных совместно с ПЭВМ «Электроника БК 0010» применяются для исследования распределенности тока электронного пучка при электронно-лучевой сварке.

Ю. П. Страшун (455-51-91, Москва) рассказал об УСО для микроЭВМ семейства СМ1800 (см. «МП», 1987, № 2) и о контроллерах, позволяющих применять эти УСО для персональной ЭВМ СМ1810.

В заключение на многочисленные вопросы о надежности отечественных персональных ЭВМ ответил ведущий семинара — Г. Р. Громов (см. об этом также в «МП», 1985, № 4).

Функции семинара все более расширяются: например, за последнее время много семинаров сыграли роль общественного отдела кадров. Ищущие работу (или наоборот, специалистов) по микропроцессорным специальностям присылали записки и после объявления со сцены, как правило, находили.

Начальник КТЦ МЭП СССР А. В. Александров (468-81-75, Москва) сделал объявление о сборе заказов от пользователей (предприятий) на разработку: заказных БИС, выполняющих определенные функции; контроллеров и микроЭВМ в заданных габаритах с заданными показателями; программ (либо систем программ) для заданных систем управления. Разработанные аппаратно-программные системы КТЦ предлагает на договорных началах реализовать, отладить и, сдать в готовом для эксплуатации виде («под ключ»), обучив пользователей правильной работе с системой.

Работа семинара, как обычно, завершилась «микросеминарами»: окруженные слушателями, докладчики ответили на многочисленные вопросы, еще раз продемонстрировали свои системы в действии.

Обзор подготовил ученый секретарь семинаров
С. С. Матвеев.

Уважаемая редакция

С микропроцессорной техникой я имею дело давно. Когда приходилось слышать или читать о темпах роста числа компьютеров за рубежом, то с чувством сожаления приходилось сознавать, что у нас до сих пор нет в продаже ПК. И вот появились первые ласточки — «БК 0010», «Микрошз». Однако купить их пока практически невозможно. После появления публикаций «Радио 86РК» я стал обладателем ПК, хотя не обошлось без проблем с микросхемами. Но обо всем этом забываешь, увесившись за компьютер.

Написать в редакцию меня побудили статьи о компьютерных играх. Это действительно увлекательное занятие как для детей, так и для взрослых...

Я все-таки надеюсь, что в ближайшее время компьютер станет массовым, поэтому необходимо развивать соответствующую инфраструктуру. Может быть имеет смысл уже сейчас создать в крупных городах, например, в Москве, Ленинграде, Новосибирске своеобразные банки игровых и учебных программ, объединив их в единую сеть. Пользуясь комбинацией телефон—компьютер—телевизор можно будет обращаться в эти банки программ и выбирать необходимые в зависимости от возраста, возможностей, опыта работы, языка программирования и т. д. Этим можно будет достигнуть не только соединения ПК в единую сеть, но и сократить время распространения программ. Появится возможность вести статистику популярности программ, а самое главное, сидящий за компьютером будет чувствовать себя частицей мира компьютеров.

Когда это будет, где и в каких масштабах? Об этом и хотелось бы прочитать на страницах вашего журнала.

Телегин Г. Н.
г. Семипалатинск

ПОЗДРАВЛЯЕМ



Редакция журнала, авторский актив и тысячи наших читателей поздравляют председателя Государственного комитета СССР по вычислительной технике и информатике, Героя Социалистического Труда, дважды лауреата Государственной премии СССР

ГОРШКОВА НИКОЛАЯ ВАСИЛЬЕВИЧА

с шестидесятилетием и желают ему доброго здоровья и творческих успехов.

УВАЖАЕМЫЕ ЧИТАТЕЛИ!

Редакция получила много писем со справедливыми нареканиями на нерегулярность выхода журнала. Причина задержки с выходом первых номеров журнала была более чем прозаическая, хотя и непростительная. К моменту выхода первого номера в свет журнал передавался из одного ведомства в другое — из ГКНТ СССР в ГКВТИ СССР — прежний издатель, к сожалению, прервал с журналом хозяйственные отношения до того, как новый издатель полностью взял журнал на свое попечение. Несколько месяцев, увы, ушло у заинтересованных ведомств на уточнение взаимных переходных обязательств, после чего типография вынуждена была во избежание эффекта падающего домино поставить первый номер журнала в хвост длинной очереди других журналов. Поэтому и последующие номера выходили с опозданием.

Мы просим читателей принять наши извинения по поводу «травматической» задержки с выходом первых номеров журнала в 1987 году. Надеемся, что она не отвратит вас от нашего общего дела.

Редакция

УДК 681.323

Шмат В. К. Специализированные БИС управления для СБИС ЗУ ЦМД//Микропроцессорные средства и системы.— 1987.— № 6.— С. 3.

Приведено описание, алгоритмы работы и временные диаграммы специализированных БИС управления К1806ВП1-103 и К1806ВП1-157 для СБИС ЗУ ЦМД типа К1602РЦ2.

UDC 681.323

Shmat V. K. Specialized controller LSIs for CMD VLSI memory chips.// Microprocessor devices and systems.— 1987.— N 6.— P. 3.

General description, operation algorithms and timing diagrams for K1806VP1-103 and K1806VP1-157 integrated circuits are given. The chips are designed to control magnetic bubble memory VLSI ICs type K1602PЦ2.

УДК 681.142

Шмат В. К. Контроллер на основе СБИС ЗУ ЦМД для измерительно-вычислительных комплексов//Микропроцессорные средства и системы.— 1987.— № 6.— С. 9.

Рассмотрен контроллер ВЗУ на СБИС ЗУ ЦМД, выполненный на основе специализированной БИС управления К1806ВП1-103. ВЗУ может быть использовано в качестве системного носителя ОС РАФОС, ОС ДВК. Приводится описание программы драйвера.

UDC 681.142

Shmat V. K. VLSI magnetic bubble memory controller for measurement and control computer systems// Microprocessor devices and systems.— 1987.— N 6.— P. 9.

External magnetic bubble memory controller using special controller chip K1806VP1-103 is described. The external memory may be used as disk emulator loaded with RAFOS operating system and serve as system device in DBK computers. Disk emulation handler for RT-11 compatible OS is also listed.

УДК 681.3.06

Петропавловский Р. Р., Ходак А. Е. Драйвер для вывода графической и алфавитно-цифровой информации//Микропроцессорные средства и системы.— 1987.— № 6.— С. 15.

Предполагается драйвер, выводящий графическую и алфавитно-цифровую информацию для ОС РАФОС микроЭВМ семейства «Электроника 60», который расширяет функциональные возможности печатающего устройства УВВПЧ-30 — 004.

UDC 681.3.06

Petrovavlovsky R. R., Hodak A. E. Graphic and alpha-numeric output handler.// Microprocessor devices and systems.— 1987.— N 6.— P. 15.

Printer handler for УВВПЧ-30-004 dot-matrix printer is proposed. The handler works in RT-11-compatible operating system on "Elektronika-60" microcomputer and enables both graphic and character output to the printer.

УДК 681.3.06

Гостев А. В. Ускорение считывания с диска в операционной системе СР/М-80//Микропроцессорные средства и системы.— 1987.— № 6.— С. 25.

Рассмотрена программа, позволяющая вдвое повысить скорость считывания с дисков одинарной плотности в операционной системе СР/М-80 за счет использования промежуточного буфера. Приведен текст программы.

Разработка программы выполнена на микроЭВМ СМ 1800.

UDC 681.3.

Gostev A. V. Increasing disk I/O data transfer rate in CP/M-80// Microprocessor devices and systems.— 1987.— N 6.— P. 25.

Disk I/O accelerator program is proposed, which doubles data transfer rate for single-density disks in CP/M-80 operating system due to the use of auxiliary buffer. The source text of the program is listed. The program was developed on CM-1800 microcomputer.

УДК 681.3.06

Добрыневский С. Ф. Драйвер графопостроителя ЭМ-7042АМ для ОС ДВК Микропроцессорные средства и системы.— 1987.— № 6.— С. 27.

Рассмотрен драйвер для управления графопостроителем ЭМ-7042АМ системными средствами ОС ДВК. При использовании драйвера графопостроитель становится доступным для стандартных процедур текстового ввода-вывода на уровне ОС и при работе с языками высокого уровня.

UDC 681.3.06

Dobrinevsky S. F. ЭМ-7042АМ plotter handler for OS/DVK.// Microprocessor devices and systems.— 1987.— N 6.— P. 27.

Digital plotter handler for RT-11 — compatible operating system is proposed. When installed into OS, the handler executes standard text output system requests and may be used for high-level language programming using conventional I/O procedures.

УДК 681.32

Емелин В. П., Новоселов В. В., Пузанков Д. В. Комфорт для микропрограммиста//Микропроцессорные средства и системы.— 1987.— № 6.— С. 30.

Охарактеризованы инструментальные средства (оборудование и программное обеспечение) для автоматизации разработки микропрограмм контроллеров на базе секционированных БИС.

UDC 681.32

Emelin V. P., Novoselov V. V., Puzanov D. V. Comfort for microprogrammer// Microprocessor devices and systems.— 1987.— N 6.— P. 30.

Hardware and software design tools for automated microprogram development for bit-slice microprocessor controllers are characterized.

Барышников В. Н., Бондарь И. Н., Воронов М. А., Романов В. Ю. Программы стартового ПЗУ ПЭВМ «Ириша» // Микропроцессорные средства и системы.— 1987.— № 6.— С. 56.

Описаны программы начального старта ПЭВМ «Ириша», обеспечивающие инициализацию отдельных узлов ПЭВМ после включения питания, выбор устройств для загрузки программ пользователя и переход к их исполнению. Приведена распечатка кодов программы для размещения в ПЗУ.

Барышников В. Н., Бондарь И. Н., Воронов М. А., Кулаков В. Б., Романов В. Ю. Монитор-отладчик программ для ПЭВМ «Ириша» // Микропроцессорные средства и системы.— 1987.— № 6.— С. 61.

Описана программа монитора-отладчика, позволяющая создавать, тестировать и запускать программы на уровне мнемокода команд процессора КР580ВМ80А. Приведены исходные тексты для настройки монитора на ПЭВМ «Ириша» и распечатка кодов программы отладчика для размещения в ПЗУ этой ПЭВМ.

Кипаренко В. В. Стандартизация интерфейсов микропроцессорных систем // Микропроцессорные средства и системы.— 1987.— № 6.— С. 68.

В обзоре рассмотрены особенности стандартизации интерфейсов распределенных систем управления. Приведены их основные характеристики, отмечены особенности.

Baryshnikov V. N., Bondar I. N., Voronov M. A., Romanov V. Yu. Start-up ROM routines for "IRISHA" computer. // Microprocessor devices and systems.— 1987.— N 6.— P. 56.

The operation of main start-up routines in "IRISHA" microcomputer is explained. The start-up ROM includes routines for power-on system hardware initialization, selection of the device to be bootstrapped, loading of user programs and their start. Hex dump of start-up ROM contents is also given.

Baryshnikov V. N., Bondar I. N., Voronov M. A., Kulakov V. B., Romanov V. Yu. Monitor-debugger for "IRISHA" PC. // Microprocessor devices and systems.— 1987.— N 6.— P. 61.

The monitor-debugger program designed for KR580ВМ80А program creation and test in assembly-language mnemonics is described. The source listings necessary for program installation as well as ROM hex dump for "IRISHA" computer are included.

Kiparenko V. V. Microprocessor interface standardization. // Microprocessor devices and systems.— 1987.— N 6.— P. 68.

The survey reveals some peculiarities of microprocessor interface standardization process in distributed control systems. Main technical specifications of the interface as well as description of some specific features are also included.

УКАЗАТЕЛЬ СТАТЕЙ, ОПУБЛИКОВАННЫХ
В ЖУРНАЛЕ ЗА 1987 ГОД

2 Беляков Б. Н., Зюляркин Д. Д., Подвальный А. М., Чернышев В. А.— Производительные силы и производственные отношения в сфере производства программного продукта
1 Ракитов А.— Информатизация общества и стратегия ускорения

МИКРОПРОЦЕССОРНАЯ ТЕХНИКА

1 Бобков В. А., Чернуха Б. Н., Свиридович В. С., Ключников В. П.— Расширенный микропроцессорный комплект БИС серии К588
3 Бокарев А. В., Гаморин М. Ю., Кабанов А. И.— БИС адаптера магистралей СМ ЭВМ и микроЭВМ «Электроника 60»
3 Бокарев А. В., Гаморин М. Ю., Кабанов А. И.— Адаптер магистралей МПИ-ОПШ
2 Весноватов М. Г., Карацуба Г. В., Павлов В. В., Старшова В. А.— Перспективные однокристалльные ЭВМ
Габдуллин Р. Р., Горемыкин В. В., Горяшко А. П., Косов Л. С., Лукович Г. А., Миронов В. Г.— Комплект БИС для встроенного диагностирования микропроцессорных систем
4 Жуков Н. Н., Корзенков А. С., Офицеров В. Н.— Печатающее устройство «Электроника МС 6307»
Кушарев В. Н., Свиридович В. С., Чернуха Б. Н., Ключников В. П., Бобков В. А., Становский В. В.— Контроллер аналого-цифрового преобразователя К588ВГ4
1 Кушарев В. Н., Свиридович В. С., Чернуха Б. Н., Бобков В. А., Ключников В. П., Нижникова Н. В.— Таймер К588ВН1
1 Морозов С. А., Барановский Д. М., Минкин Л. К., Семинкин А. П., Черкай А. Д.— Однокристалльные ЭВМ серии КБ1013

Мячев А. А.— Микроконтроллеры экономичных подсистем автоматизации на базе еврокаркасов и приборных магистралей 4
2 Свиридович В. С., Черноусова Т. Г., Чернуха Б. Н., Бобков В. А.— Контроллер прерываний К588ВН1 5
1 Шмат В. К.— Специализированные БИС управления для СБИС ЗУ ЦМД 6
Шмат В. К.— Контроллер на основе СБИС ЗУ ЦМД для измерительно-вычислительных комплексов 6

ПЕРСОНАЛЬНЫЕ КОМПЬЮТЕРЫ

Алексенко А. Г., Глазков М. А., Галицын А. А.— Высоконадежная ПЭВМ на базе БИС К1810ВМ86 1
Вигдорчик Г. В., Бохменцев М. Я., Климович В. П., Леонов П. П.— Персональная ЭВМ «Квант» 1
Вигдорчик Г. В., Бохменцев М. Я., Климович В. П., Леонов П. П., Семик В. П.— Персональная ЭВМ ПК-11 1
Воробьев А. Д., Киселер Э. Г., Пац В. Б.— ПЭВМ «Истра»: архитектура, технические характеристики 1
Громов Г. Р.— Игровая компонента персональной ЭВМ: стимулятор творчества, педагогический прием, жанр киноискусства 3
Кочетков Г. Б.— Нужен ли компьютер дома? 3
Лемко Л. М., Гладков В. В., Ермаков С. В., Жуков В. Н.— Персональный микрокомпьютер «Электроника МК 85» 4
Лонатин В. И., Старовойтов Ю. Н.— Программирование персонального компьютера «Электроника МК 85» с использованием языка Бейсик 4
5 Пажитнов А. Л.— Логическая структура компьютерной игры 3

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Арпаксыд В. М., Володарский И. Б., Дорфман А. Е.— Мобильная операционная система для ПЭВМ «Искра 226» 5
 Берестихевский С. И., Колосова Т. В., Мартыненко О. Н.— Пакет программ по прикладной статистике для персональных ЭВМ 4
 Большаков И. А.— Программную документацию — высококачественно! 3
 Бочков С. О., Смелянский Р. Л.— Символьный отладчик для языка Си 5
 Борзов Г. В., Ляпунов М. М.— Программные трюки на МАКРО-11 5
 Борковский А. Б.— Текстовая база данных 5
 Брябрин В. М., Блинов Д. М.— Интегрированная система для решения прикладных задач 1
 Варсанюфьев Д. В., Дымченко А. Г., Кушнirenко А. Г., Лебедев Г. В.— Непосредственный драйвер текста НДТ-83 и системы на его базе 1
 Воржев А. В., Зверков Б. С., Кикоть А. И., Яковлева Е. В.— Интегрированная система разработки кросс-системы на базе ДВК-2М для микропроцессоров 4
 Гнездилова Г. Г.— Интегрированная операционная среда персональной ЭВМ 1
 Горожин А. Д., Вознюк Л. А., Чвыров Д. А.— Операционная система СР/М на базе КНМЛ 4
 Гостев А. В.— Ускорение считывания с диска в ОС СР/М-80 6
 Григорьев А. Г.— Диалоговый дизассемблер для загрузочных модулей операционной системы РАФОС 4
 Григорьев А. Г.— Адаптированная операционная система АДОС для СМ ЭВМ 4
 Добриневский С. Ф.— Драйвер графопостроителя ЭМ-7042АМ для ОС ДВК 6
 Доломанов В. Г., Канофьев А. В.— Структурные особенности интерпретирующей системы Бейспк/РАФОС 4
 Емелин В. П., Новоселов В. В., Пузанков Д. В.— Комфорт для микропрограммиста 6
 Завилов В. Н., Константинов М. Ю., Померанец М. В.— Программирование на языке Паскаль для микроЭВМ «Электроника БК-0010» 1
 Игнатова И. Г., Преснухин Д. Л.— Выбор режимов адресации при программировании в системе команд микроЭВМ типа «Электроника 60» 6
 Кавалерчик Б. Я.— Надежность программного обеспечения и условия эксплуатации 3
 Казменко С. В.— Автоматизация проектирования систем на основе объединения концепций макетирования и «программирования без программиста» 4
 Каргин А. А.— Проблемно-ориентированное программное обеспечение микропроцессорных систем в робототехнике 2
 Коминаров И. З., Молодцова Л. В., Самощенко А. В., Теплинский С. В., Школа А. А.— Программное обеспечение комплекса функционального диагностирования микроЭВМ «Электроника С5-21М» 2
 Котляров В. П.— Технология разработки программного обеспечения встроенных микроЭВМ и поддерживающие ее инструментальные комплексы 5
 Ляпин А. А., Деснер И. Г.— Кроссассемблер для микропроцессора К1810ВМ86 1
 Мешков Н. А.— Студенческий научно-производственный отряд математиков-программистов «Прометей» 6
 Петропавловский Р. Р., Ходак А. Е.— Драйвер для вывода графической и алфавитно-цифровой информации 6
 Погодин Ю. М.— МИАСС — система микропрограммирования на языке АМДАСМ 5

Подольский Л. И., Лясковский А. П.— Система программирования Quasic-2 для микроЭВМ 2
 Попов С. Н.— Программа «Имитатор ДОС» для 8-разрядной ПЭВМ 6
 Рауд А. К., Смелянский Р. Л.— Отечественные кросс системы 5
 Руденко Ю. М., Чернова Т. Ф.— Библиотека стандартных программ для БИС микроЭВМ серии К1816 6
 Селицкий С. С., Сыркин М. Ю.— Процедура перемещения частей загрузочного модуля для микропроцессора КР580ИК80 4
 Сизов К. А.— Программа обработки текстов с переменной шириной знакоформирующей матрицы 6
 Снежко Е. М., Тихомиров А. Е.— Компактный мобильный кроссассемблер для микропроцессора КР580ВМ80 3
 Хаев В. Э.— Пакет программ для хранения планово-экономической информации на микроЭВМ «Искра 226» 4
 Хацкевич Л. Д., Койлис Л. Л.— СУБД для персональной ЭВМ «Электроника 85» 6
 Щелкунов Н. Н., Дзанов А. П.— Техника программирования 16-разрядных микроконтроллеров 2

Как учить программированию

 Круг Г. К., Кабанов В. А., Черных А. В.— Инструментальные диалоговые обучающие системы на микроЭВМ 3
 Подвальный С. Л., Кравец О. Я., Михин Ю. А.— Школьный вычислительный комплекс 3

Тестирование и отладка

 Альперович Е. И., Белый В. Г., Большинский С. И.— Двухмашинный отладочный комплекс для разработки устройств на базе БИС серии КР580 6
 Бабышников В. Н., Бондарь И. Н., Воронов М. А., Кулаков В. Б.— Программы стартового ПЗУ ПЭВМ «Ириша» 6
 Барышников В. Н., Бондарь И. Н., Воронов М. А., Романов В. Ю.— Монитор-отладчик программ для ПЭВМ «Ириша» 6
 Богущ С. С., Борисов В. С., Горемыкин В. В.— Обнаружение и исправление ошибок в ПЗУ 1
 Бородин С. М., Зайцев В. Л., Новиков Ю. В.— Быстродействующий логический анализатор для аппаратно-программных комплексов разработки микропроцессорных систем 1
 Бородин С. М., Новиков Ю. В.— Модуль логического анализатора для контрольно-измерительных систем на базе микроЭВМ 1
 Бровка В. И.— Комплекс для отладки изделий на основе однокристальной ЭВМ К1816ВБ51 6
 Восвудский В. П., Каштанов В. В., Семенова О. С., Фокин А. А.— Система диагностирования в периферийном процессоре «Электроника МС 1603» 6
 Гайдучок Р. М., Шумский Г. А.— Портативная система отладки — тестер «Садко» 1
 Горовой В. Р., Васильев Н. П.— Комплекс средств отладки микропроцессорных систем «Электроника НЦ-803» 1
 Ефремов В. Я.— Сигнатурный анализатор 6
 Каширин В. М., Понтак М. С., Рубинштейн Д. И., Смолкин В. С.— Комплекс отладки систем на базе однокристальных ЭВМ серии КМ1816 6
 Коршунов И. Г., Ярменко В. Г.— Простой 64-входовый логический анализатор на микроЭВМ «Электроника Д3-28» 1
 Пилипович В. А., Есман А. К., Ерилов А. А., Савченко А. А.— 16-разрядный микроконтроллер со встроенными средствами поиска неисправностей с помощью сигнатурного анализа 1
 Припоров С. В.— Экранный отладчик для микроЭВМ СМ1800 6

Соколовский А. С., Яковлев Н. И., Смолин А. Т., Сытник М. Г.—Комбинированный цифровой прибор для диагностирования неисправностей МП систем
1
Степанов А. Н., Жуков В. А.—Комплекс аппаратно-программных средств отладки микропрограмм для разрядно-модульных микропроцессоров
6
Тетерин Ю. Н.—Организация встроенного тестирования устройства подготовки данных на основе ЕС 9075
1
Ткаченко А. М., Барго В. Л., Тютюнников Н. В.—Методы тестирования микроЭВМ
1
Тюлькин С. П.—Программа тестирования ОЗУ
1

Магистральи МП-систем

Канцеров В. А., Першин А. С.—VME — магистраль нового поколения
5
Кипаренко В. В.—Стандартизация интерфейсов микропроцессорных систем
6
Коломейцев В. А., Степченко Ю. А., Филин А. В.—Интерфейс с последовательным арбитражем: реализация и пути усовершенствования
5
Колпаков И. Ф.—Шина VME и ее применения
5

Машинная графика

Алексеев М. М., Брезовский М. А., Свирич Б. Н., Яблонский А. К., Яким В. В.—Интерактивная система геометрического моделирования для СВ ЭВМ и векторного процессора
5
Алешин А. Н., Крюков С. Н.—Программное обеспечение символично-графического цветного телевизионного дисплея
4
Бабкин П. А., Солопенко О. П., Тарасов Б. В., Федорин В. Г.—Средства цветной машинной графики для микроЭВМ «Электроника 60»
4
Большаков И. А.—«Черепашья» графика на ДВК
5
Брлбрин В. М., Сираджов Б. Т.—Пакет демонстрационной графики АЛЬФА-ГРАФ для ПЭВМ типа ЕС-1841
5
Глумов М. Г., Вернер С. А.—Драйверы графических устройств СМП КУЛОН-1 в операционной системе NTS
4
Демин А. П., Харитонов Г. И.—Устройство отображения графической информации на алфавитно-цифровом дисплее
4
Кулаичев А. П.—Графический ассистент для синтеза каталогизации и экспонирования цветных изображений
5
Микропроцессорная графическая станция ГТ-80
4
Овчинников Л. Г., Сороченко Н. И.—Устройство индикации
4
Попов С. Н.—Графический редактор для ПЭВМ «Микроша»
5
Семенов П. А., Процак А. М., Егоров В. П.—Цветная графика в микроЭВМ «Электроника 60» и «Электроника ИЦ-80»
4

Инженер оформляет заявку на изобретение

Блининков В. И., Минашин В. П., Бедин Б. Д., Сарьян В. К.—Система безбумажной технологии делопроизводства по заявкам на изобретения
4
Сарьян В. К., Корытов В. В.—Увеличение времени жизни диалоговой системы по оформлению первичных документов по заявкам
4
Сарьян В. К., Смолич Г. Г.—Взаимодействие больших и малых баз данных в автоматизированной системе оформления заявок
4

Лабунов В. А., Макушок Ю. Е., Пархутик В. П., Щершульский В. И.—Диалоговая система сбора и статистического анализа данных для персональных компьютеров
2
Сычев Н. Ф., Штеренгарц Е. М.—Виртуальный интерфейс микроЭВМ «Электроника 60»
3
Ходаковский Е. А.—Сопряжение микропроцессора с «медленными» устройствами ввода-вывода информации
3
Аппаратно-программные средства для автоматизации производственных процессов
1

Адамов Е. О., Иванов В. Г., Ляпунов М. М.—Аппаратные средства КАПРИ
2
Алексеева Р. П., Жильцов А. В., Коробова Л. Е., Михайлов О. А., Песков М. И., Сулицкий Ю. Н.—ДВК-2М в производстве фотопластов печатных плат
2
Алексенко А. Г., Галицын А. А., Баландин К. Н., Наратов С. В.—Система автоматизированного проектирования матричных БИС на базе ДВК-2М
2
Антонюк Б. Д.—МАР-интегрированная система автоматизации производства
2
Горбачев В. С.—Микропроцессорные системы управления роботами
2
Ильин Е. П., Ляпунов М. М.—Аппаратура дистанционного контроля и управления с оперативными УЧПУ
2
Кузнецов М. Н., Громов Ю. К., Кумачев Ю. М., Некрасов Л. Т., Плешаков В. А.—Распределенная система управления технологическим оборудованием и гибкими автоматизирующими линиями
2
Обноуленский П. А., Рудометов Е. А., Фокин А. Л., Харазов В. Г., Капустина Н. В.—МикроЭВМ «Электроника БК-0010» в системе управления производством кварцевого стекла
2
Флейшер Е. Г.—Организация межпроцессорного обмена в УЧПУ с подчиненными контроллерами
2
Чернявский А. Д., Шаповалов В. И.—Комплексирование микроЭВМ «Электроника ИЦ80-20 и НМЛ
2
Целуйко В. З., Климович В. Г., Ратников В. Н., Пеньков И. В.—Микропроцессорные системы управления серии «Гранит 02»
2

Медицинское приложение

Бородкин С. М., Лукьянов В. И., Зайцев В. А., Тетерин Е. А.—Микрокомпьютерные средства контроля состояния мозга
3
Каневский А. М., Розман Б. Я.—Двухплатный микроконтроллер на базе МПК БИС КР580 для медицинских применений
3
Семенов П. А., Федоров С. Н., Миронова Э. М., Егорова Э. В.—Микрокомпьютерная система на базе МПК БИС КР580 для автоматической обработки биосигналов глаза
3
Чернин С. Л., Береговская Е. М.—Ультразвуковой эхоофтальмометр на базе однокристалльной микроЭВМ КР1816ВЕ48
3
Шварев Ю. И.—Использование ЭВМ «Искра 226.6» в электроэнцефалографическом эксперименте
3

Микропроцессоры — новый инструмент музыкального творчества

Барышников Ю. Н., Белкин Б. Г., Гордон М. Г.—Микропрограммируемый процессор — инструмент звукорежиссера
3
Кальян В. П.—Роль музыкального языка в музыкальном синтезе
3
Кузнецов А. И., Михайлова Н. Н.—Системотехника компьютерных музыкальных синтезаторов
3

Лукьянов Д. А., Михайлова Н. Н.—МИДИ — сетевой интерфейс музыкальных систем	3	Томсинский И. Я.—Как разработать электронную схему?	5
Родионов А. Б.—Персональный компьютер в музыкальном творчестве	3	Чабан С. Д., Фильцагин Ю. А.—Модуль десятиканального аналогового вывода на базе микроЭВМ «Электроника 60»	4
Танган А. С.—Музыкальная информатика: средства и перспективы	3	Шалуло А. М., Афанасьева Н. Н.—Отладочный комплекс контроллера «Электроника К1-20»	4
Системы технического зрения		Щелкунов Н. Н., Дианов А. П.—Одноплатный 16-разрядный микроконтроллер	1
Гусев И. Т., Зайцев К. С., Бережинский И. В., Баталин А. А., Кац Д. А.—Система технического зрения для контроля режущего инструмента на станках с ЧПУ	2	Справочная информация	
Матвсенко В. И., Староверов Ю. Г.—Полутоновая система технического зрения	2	Водянкин А. Г., Моисеенко В. И.—Учебная локальная сеть микроЭВМ	4
Руцков М. В.—Видеопроцессор бинарных сигналов	2	Дианов А. П., Щелкунов Н. Н.—Организация динамической памяти микросистем	4
Хоботов А. Г., Плеханов В. М., Крылов В. В.—Система технического зрения на базе серийной микроЭВМ	2	Монахов В. Т.—Программы любителей для «Электроники БК-0010»	4
Устройства связи микроЭВМ с объектами		Мышкин И. Л., Щербаков А. С.—Одноплатная микроЭВМ «ПРОЕКТ-80»	4
Аюпов Р. М., Чабан С. Д.—Восьмиканальный модуль последовательного обмена для микроЭВМ «Электроника 60»	2	Краткие сообщения	
Гаврилов А. С., Тюленева Л. А.—Микроконтроллер на основе ОМП К1801ВМ1 и БИС серии К583	2	Ачкасов В. Н., Глазнев А. Н., Цыбин С. А.—Быстродействующие неактивируемые статические ОЗУ с байтовой организацией КМ581РУ5	6
Страшун Ю. П.—Средства связи микроЭВМ СМ1800 с объектом	2	Кулешова В. И.—Серийные микропроцессорные комплекты БИС	4
Мамедов Ф. С., Васильев В. Г.—Сопряжение быстродействующего интегрального АЦП К1107ПВ1 с микроконтроллерами	2	Микропроцессорный комплект серии КР580	5
Обмен опытом		Однократно программируемые ПЗУ серии КР556	1, 2, 3
Баранов А. С., Островский Е. М.—Интерфейс многопроцессорной системы	2	Справочная информация	
Грибов И. В., Шумаков А. В.—Опыт эксплуатации диалоговых вычислительных комплексов	1	Краткие сообщения	
УЧЕБНЫЙ ЦЕНТР		Власов П. А., Матвеев В. В., Ягунов М. А.—Микропролог с графическими возможностями	2
Антоняк И. И., Зайшлый Я. С., Фриш А. Е.—Импульсный стабилизатор специализированных напряжений для микропроцессора КР580ВМ80	4	Геворкян Г. Х., Семенов В. Н.—Аннотация к пакету программ для расчета электрических схем	6
Валиков В. В.—Синхрогенератор для формирования телевизионного сигнала	1	Гофман А. Я.—Универсальное устройство сопряжения считывателя с перфоленты Г-1501 с интерфейсом ИРПП	6
Гнатив Р. М., Скобылко А. Я.—Цифровой генератор синусоидальных сигналов с управлением по частоте на спецпроцессоре	4	Динес В. Л., Мартюхин В. А.—Кадровый редактор для микропроцессорной системы КТС ЛИУС-2 с дисплеем РИН-609	2
Дианов А. П., Щелкунов Н. Н.—Система проектирования микропроцессорных устройств	5	Казанцев А. П.—Интерфейс внешних функций интерпретатора Фокал-БК-0010	4
Ерухимов Б. Л., Черненко В. Н.—Трехуровневый комплекс для автоматизации научных исследований на радиотелескопе РАТАН 600	5	Казаринов В. Е. и др.—Автоматизированный эталонный поромер	2
Коломисц Л. В., Горемыкин В. К.—Защита интегральных схем по цепям питания	4	Калужный В. М., Корпилович А. А., Уваров Е. И., Федан П. Н.—Программный контроллер сопряжения микроЭВМ «Искра 226» и системы КАМАК	6
Королев В. Н., Жарков А. С., Зубченко А. П., Штанков А. Ю.—Микропроцессорная система МС-80	5	Холоимов А. Ю.—Микропроцессорная система управления телевизионными установками	2
Кузьминов А. Ю., Мацея А. В.—Сопряжение анализатора спектра СК4-72/2 с микроЭВМ СМ1800	1	Куправа Т. А.—Интерпретатор языка Фокал в микросхеме ПЗУ	6
Малежин О. Б., Крыликов Н. О., Преснухин Д. Л.—Интерфейс параллельного ввода-вывода микроЭВМ «Электроника 60» на основе БИС серии К1801, К588	5	Лапиров А. В., Рудометов Е. А., Харазов В. Г.—АЦП на БИС К1113ПВ1 для персональной ЭВМ «Электроника БК-0010»	4
Пипченко А. Н., Гончарук С. П., Пономаренко В. В., Савельев А. Е.—Распределенная система управления судовой электростанцией на базе КТС ЛИУС-2	5	Найденев А. В., Туманов А. А., Комаров И. Е., Романенков В. А.—МикроЭВМ «Искра 226» в системе проектирования контроллеров на базе БИС КР580	1
Поздеев В. С., Тихонов О. Н.—Отображение информации на дисплее ИЖВ	4	Петух А. М., Романюк А. Н., Ободник Д. Т., Подольский О. А.—Программно-аппаратная реализация линейного интерполятора	4
Солоненко В. И., Касперович А. Н., Ефремов А. И.—Электронный диск — внешнее ЗУ ЭВМ	3	Забоев С. В., Исаков В. С.—Интерфейс микроЭВМ «Электроника ДЗ-28» — автомат «Оргтекст»	6
Страбыкин Д. А.—Экспериментальные исследования микропроцессорных устройств с помощью диалоговых систем управления	4	Сорокин Ю. Ю., Максимяк С. П., Субач В. В.—Простой адаптер для подключения кассетного магнитофона к линии последовательного обмена	6
Тилинин Д. А., Глазачев Н. К., Айсанов Р. Б.—ПЭВМ «Океан 240»: конструкция и методика отладки	3	Сорекин Ю. Ю., Максимяк С. П., Субач В. В.—Автоматический останов привода в накопителе «Электроника ГМД-7012»	6

НА IV ВСЕСОЮЗНОЙ КОНФЕРЕНЦИИ ПО ПРОБЛЕМАМ МАШИННОЙ ГРАФИКИ (г. Протвино, сентябрь, 1987)

Цветная растровая графика открывает совершенно новые (порой неожиданные) сферы для применения ЭВМ. Показательны в этом отношении доклады об отечественных цветных дисплейных системах ГАММА. Дисплей ГАММА-4 выводит на экран цветное изображение на растре 1024×768 точек, содержащее до 256 из возможных 16 млн. оттенков (рис. 1). Алгоритмы заполнения многоуголь-

Рис. 1



ников и координатный шар, как устройство ввода, позволяют использовать дисплейную систему для создания рисунков. Рис. 2 был подготовлен непосредственно на экране дисплея. Система может использоваться и для раскраски рисунка, введенного в машину с помощью оцифровщика или телевизионной камеры. Этот способ демонстрируется на рис. 3, а некоторая его обработка — на рис. 4.

Рис. 2

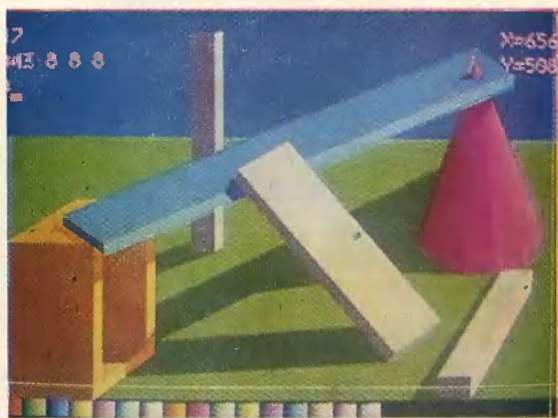


Рис. 3



Рис. 4



1р. 10к.

Индекс 70588



Москва • 1987 |

МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ

ВНИМАНИЕ

Если Вы не успели подписаться на наш журнал, **НАПОМИНАЕМ**, что подписка на «МП» **ПОКВАРТАЛЬНАЯ**